



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
- ☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Java
- ☐ Lisp
- ☐ C
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ on ne peut pas
- ☐ avec le compilateur
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ en l'exécutant sur l'ordinateur

Question 4 En informatique, est-ce qu'un *type* ?

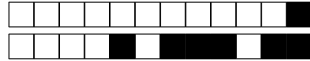
- ☐ une variable
- ☐ une constante
- ☐ une personne
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques

Question 5 ♣ En C, un entier peut être de type :

- ☐ long
- ☐ double
- ☐ int
- ☐ très long
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ 100
- ☐ -35
- ☐ -1
- ☐ -34



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ pair
- ☐ positif
- ☐ impair
- ☐ négatif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 126 et 767
- ☐ -3 et 768
- ☐ 125 et 765
- ☐ aucune des deux

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont toujours exacts
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « formel donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x \neq 0$
- ☐ $x = 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de structurer le programme
- ☐ de rendre le programme plus complexe
- ☐ de paramétrer les actions
- ☐ de rendre le programme moins lisible
- ☐ de localiser les objets
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ variables
- ☐ effectifs
- ☐ formels
- ☐ effaçables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie uniquement à la première itération
- ☐ vraie uniquement à la dernière itération
- ☐ vraie à chaque itération
- ☐ jamais vraie

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ la probabilité que a soit égal à b
- ☐ le minimum de a et b
- ☐ le produit de a par b
- ☐ le pgcd (plus grand commun diviseur) de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ -1 et N+1
- ☐ 1 et N
- ☐ 0 et N
- ☐ 0 et N-1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 1 seul énoncé itératif **for**
- ☐ 3 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ jamais vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ toujours vérifié
- ☐ parfois vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016**Durée 1h30**

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ C
☐ Lisp
☐ Java
☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ en l'exécutant sur l'ordinateur
☐ on ne peut pas
☐ avec le compilateur
☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents

Question 4 En informatique, est-ce qu'un *type* ?

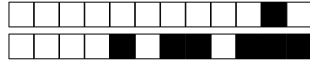
- ☐ une variable
☐ une constante
☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
☐ une personne

Question 5 ♣ En C, un entier peut être de type :

- ☐ long
☐ très long
☐ int
☐ double
☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -35
☐ -1
☐ 100
☐ -34



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ pair
- ☐ négatif
- ☐ positif
- ☐ impair

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 125 et 765
- ☐ aucune des deux
- ☐ 126 et 767
- ☐ -3 et 768

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont toujours exacts
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif donnée » et « résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$
- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de paramétrer les actions
- ☐ de structurer le programme
- ☐ de rendre le programme plus complexe
- ☐ de localiser les objets
- ☐ de rendre le programme moins lisible
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ variables
- ☐ formels
- ☐ effaçables
- ☐ effectifs

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie uniquement à la dernière itération
- ☐ vraie uniquement à la première itération
- ☐ jamais vraie
- ☐ vraie à chaque itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le produit de a par b
- ☐ le minimum de a et b
- ☐ la probabilité que a soit égal à b
- ☐ le pgcd (plus grand commun diviseur) de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N
- ☐ 1 et N
- ☐ 0 et N-1
- ☐ -1 et N+1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 1 seul énoncé itératif **for**
- ☐ 3 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ est vérifié si l'ordinateur est en forme
- ☐ jamais vérifié
- ☐ parfois vérifié
- ☐ toujours vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ elle renvoie 1 si `x` $\in$ `t` et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
- ☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Lisp
- ☐ C
- ☐ Java
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ on ne peut pas
- ☐ avec le compilateur
- ☐ en l'exécutant sur l'ordinateur
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents

Question 4 En informatique, est-ce qu'un *type* ?

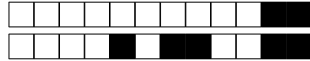
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une variable
- ☐ une personne
- ☐ une constante

Question 5 ♣ En C, un entier peut être de type :

- ☐ très long
- ☐ int
- ☐ long
- ☐ double
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ 100
- ☐ -1
- ☐ -34
- ☐ -35



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ négatif
- ☐ positif
- ☐ impair
- ☐ pair

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 125 et 765
- ☐ 126 et 767
- ☐ -3 et 768
- ☐ aucune des deux

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont toujours exacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « formel donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de structurer le programme
- ☐ de paramétrer les actions
- ☐ de rendre le programme moins lisible
- ☐ de rendre le programme plus complexe
- ☐ de localiser les objets
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effaçables
- ☐ formels
- ☐ variables
- ☐ effectifs

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie uniquement à la dernière itération
- ☐ vraie à chaque itération
- ☐ jamais vraie
- ☐ vraie uniquement à la première itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le produit de a par b
- ☐ le minimum de a et b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ la probabilité que a soit égal à b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ -1 et N+1
- ☐ 1 et N
- ☐ 0 et N
- ☐ 0 et N-1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 2 énoncés itératifs **for**
- ☐ 3 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ est vérifié si l'ordinateur est en forme
- ☐ parfois vérifié
- ☐ jamais vérifié
- ☐ toujours vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016**Durée 1h30**

Nom :

Prénom :

ID :

*Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.**Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.***Question 1**

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ C
☐ Lisp
☐ Java
☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ en l'exécutant sur l'ordinateur
☐ avec le compilateur
☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
☐ on ne peut pas

Question 4 En informatique, est-ce qu'un *type* ?

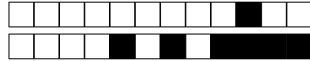
- ☐ une personne
☐ une constante
☐ une variable
☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques

Question 5 ♣ En C, un entier peut être de type :

- ☐ int
☐ très long
☐ long
☐ double
☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -34
☐ 100
☐ -1
☐ -35



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ impair
- ☐ positif
- ☐ négatif
- ☐ pair

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ aucune des deux
- ☐ -3 et 768
- ☐ 125 et 765
- ☐ 126 et 767

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ les calculs avec les réels sont toujours exacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$
- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de structurer le programme
- ☐ de rendre le programme moins lisible
- ☐ de paramétrer les actions
- ☐ de localiser les objets
- ☐ de rendre le programme plus complexe
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effectifs
- ☐ effaçables
- ☐ variables
- ☐ formels

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie à chaque itération
- ☐ vraie uniquement à la dernière itération
- ☐ vraie uniquement à la première itération
- ☐ jamais vraie

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le produit de a par b
- ☐ la probabilité que a soit égal à b
- ☐ le minimum de a et b
- ☐ le pgcd (plus grand commun diviseur) de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 1 et N
- ☐ 0 et N-1
- ☐ -1 et N+1
- ☐ 0 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 1 seul énoncé itératif **for**
- ☐ 3 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ jamais vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ parfois vérifié
- ☐ toujours vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ elle renvoie 1 si `x` $\in$ `t` et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
- ☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ C
- ☐ Java
- ☐ Lisp
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ en l'exécutant sur l'ordinateur
- ☐ on ne peut pas
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ avec le compilateur

Question 4 En informatique, est-ce qu'un *type* ?

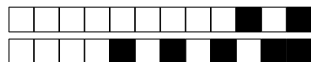
- ☐ une variable
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une constante
- ☐ une personne

Question 5 ♣ En C, un entier peut être de type :

- ☐ très long
- ☐ double
- ☐ long
- ☐ int
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ 100
- ☐ -34
- ☐ -35
- ☐ -1



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ pair
- ☐ impair
- ☐ négatif
- ☐ positif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ aucune des deux
- ☐ 126 et 767
- ☐ -3 et 768
- ☐ 125 et 765

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont toujours exacts
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de structurer le programme
- ☐ de localiser les objets
- ☐ de rendre le programme plus complexe
- ☐ de rendre le programme moins lisible
- ☐ de paramétrer les actions
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effaçables
- ☐ effectifs
- ☐ formels
- ☐ variables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie uniquement à la dernière itération
- ☐ vraie uniquement à la première itération
- ☐ jamais vraie
- ☐ vraie à chaque itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le minimum de a et b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le produit de a par b
- ☐ la probabilité que a soit égal à b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ -1 et N+1
- ☐ 1 et N
- ☐ 0 et N-1
- ☐ 0 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés **if-then-else**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

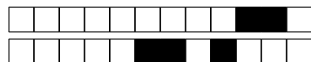
- ☐ toujours vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ parfois vérifié
- ☐ jamais vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016**Durée 1h30**

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ C
☐ Java
☐ Lisp
☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
☐ avec le compilateur
☐ on ne peut pas
☐ en l'exécutant sur l'ordinateur

Question 4 En informatique, est-ce qu'un *type* ?

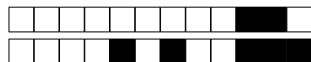
- ☐ une personne
☐ une variable
☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
☐ une constante

Question 5 ♣ En C, un entier peut être de type :

- ☐ double
☐ très long
☐ long
☐ int
☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -35
☐ -34
☐ 100
☐ -1



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ négatif
- ☐ positif
- ☐ pair
- ☐ impair

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 125 et 765
- ☐ -3 et 768
- ☐ 126 et 767
- ☐ aucune des deux

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ les calculs avec les réels sont toujours exacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « formel donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$
- ☐ $x = 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x \neq 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de rendre le programme moins lisible
- ☐ de paramétrer les actions
- ☐ de rendre le programme plus complexe
- ☐ de structurer le programme
- ☐ de localiser les objets
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effaçables
- ☐ effectifs
- ☐ variables
- ☐ formels

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie à chaque itération
- ☐ vraie uniquement à la première itération
- ☐ jamais vraie
- ☐ vraie uniquement à la dernière itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ la probabilité que a soit égal à b
- ☐ le produit de a par b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le minimum de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N
- ☐ 0 et N-1
- ☐ -1 et N+1
- ☐ 1 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 3 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 2 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ est vérifié si l'ordinateur est en forme
- ☐ toujours vérifié
- ☐ jamais vérifié
- ☐ parfois vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016**Durée 1h30**

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ C
☐ Java
☐ Lisp
☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ avec le compilateur
☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
☐ en l'exécutant sur l'ordinateur
☐ on ne peut pas

Question 4 En informatique, est-ce qu'un *type* ?

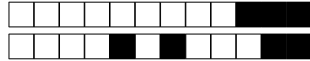
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
☐ une personne
☐ une constante
☐ une variable

Question 5 ♣ En C, un entier peut être de type :

- ☐ double
☐ long
☐ très long
☐ int
☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -34
☐ 100
☐ -1
☐ -35



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ positif
- ☐ négatif
- ☐ pair
- ☐ impair

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 126 et 767
- ☐ aucune des deux
- ☐ 125 et 765
- ☐ -3 et 768

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ les calculs avec les réels sont toujours exacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et $(x \neq 1 \text{ ou } y \neq 0)$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de rendre le programme plus complexe
- ☐ de localiser les objets
- ☐ de structurer le programme
- ☐ de paramétrer les actions
- ☐ de rendre le programme moins lisible
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ variables
- ☐ formels
- ☐ effaçables
- ☐ effectifs

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie à chaque itération
- ☐ vraie uniquement à la première itération
- ☐ jamais vraie
- ☐ vraie uniquement à la dernière itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le minimum de a et b
- ☐ le produit de a par b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ la probabilité que a soit égal à b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N
- ☐ -1 et N+1
- ☐ 0 et N-1
- ☐ 1 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés itératifs **for**
- ☐ 3 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés **if-then-else**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (i.e. `t[i]`) est :

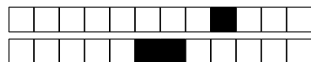
- ☐ parfois vérifié
- ☐ toujours vérifié
- ☐ jamais vérifié
- ☐ est vérifié si l'ordinateur est en forme

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016**Durée 1h30**

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Lisp
☐ Java
☐ C
☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
☐ avec le compilateur
☐ on ne peut pas
☐ en l'exécutant sur l'ordinateur

Question 4 En informatique, est-ce qu'un *type* ?

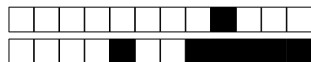
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
☐ une constante
☐ une variable
☐ une personne

Question 5 ♣ En C, un entier peut être de type :

- ☐ très long
☐ double
☐ long
☐ int
☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -35
☐ -1
☐ 100
☐ -34



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ impair
- ☐ négatif
- ☐ pair
- ☐ positif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ -3 et 768
- ☐ 125 et 765
- ☐ aucune des deux
- ☐ 126 et 767

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont toujours exacts
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

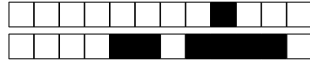
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « formel donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x = 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de rendre le programme plus complexe
- ☐ de localiser les objets
- ☐ de structurer le programme
- ☐ de paramétrer les actions
- ☐ de rendre le programme moins lisible
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ formels
- ☐ variables
- ☐ effectifs
- ☐ effaçables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie uniquement à la dernière itération
- ☐ vraie à chaque itération
- ☐ vraie uniquement à la première itération
- ☐ jamais vraie

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ la probabilité que a soit égal à b
- ☐ le produit de a par b
- ☐ le minimum de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N-1
- ☐ 0 et N
- ☐ 1 et N
- ☐ -1 et N+1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

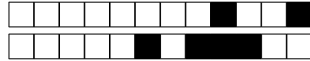
- ☐ toujours vérifié
- ☐ jamais vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ parfois vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ elle ne fait rien et ne renvoie rien
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016**Durée 1h30**

Nom :

Prénom :

ID :

*Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.**Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.***Question 1**

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ C
☐ Java
☐ Lisp
☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ en l'exécutant sur l'ordinateur
☐ on ne peut pas
☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
☐ avec le compilateur

Question 4 En informatique, est-ce qu'un *type* ?

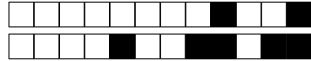
- ☐ une variable
☐ une constante
☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
☐ une personne

Question 5 ♣ En C, un entier peut être de type :

- ☐ double
☐ très long
☐ long
☐ int
☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -34
☐ 100
☐ -1
☐ -35



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ impair
- ☐ positif
- ☐ négatif
- ☐ pair

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ -3 et 768
- ☐ aucune des deux
- ☐ 126 et 767
- ☐ 125 et 765

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont toujours exacts
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

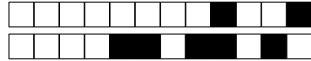
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de paramétrer les actions
- ☐ de rendre le programme plus complexe
- ☐ de localiser les objets
- ☐ de rendre le programme moins lisible
- ☐ de structurer le programme
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effaçables
- ☐ variables
- ☐ formels
- ☐ effectifs

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie uniquement à la première itération
- ☐ jamais vraie
- ☐ vraie uniquement à la dernière itération
- ☐ vraie à chaque itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le minimum de a et b
- ☐ le produit de a par b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ la probabilité que a soit égal à b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ -1 et N+1
- ☐ 0 et N-1
- ☐ 1 et N
- ☐ 0 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 1 seul énoncé itératif **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

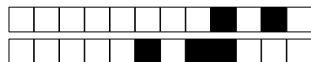
- ☐ est vérifié si l'ordinateur est en forme
- ☐ parfois vérifié
- ☐ toujours vérifié
- ☐ jamais vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
- ☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Lisp
- ☐ Java
- ☐ C
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ on ne peut pas
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ avec le compilateur
- ☐ en l'exécutant sur l'ordinateur

Question 4 En informatique, est-ce qu'un *type* ?

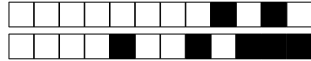
- ☐ une variable
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une constante
- ☐ une personne

Question 5 ♣ En C, un entier peut être de type :

- ☐ int
- ☐ double
- ☐ long
- ☐ très long
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -34
- ☐ -35
- ☐ 100
- ☐ -1



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ négatif
- ☐ impair
- ☐ positif
- ☐ pair

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 126 et 767
- ☐ aucune des deux
- ☐ 125 et 765
- ☐ -3 et 768

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont toujours exacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

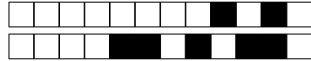
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ $x = 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de paramétrer les actions
- ☐ de rendre le programme moins lisible
- ☐ de rendre le programme plus complexe
- ☐ de localiser les objets
- ☐ de structurer le programme
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours vraies avant l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effectifs
- ☐ formels
- ☐ effaçables
- ☐ variables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ jamais vraie
- ☐ vraie uniquement à la dernière itération
- ☐ vraie uniquement à la première itération
- ☐ vraie à chaque itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le minimum de a et b
- ☐ la probabilité que a soit égal à b
- ☐ le produit de a par b
- ☐ le pgcd (plus grand commun diviseur) de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N-1
- ☐ 1 et N
- ☐ -1 et N+1
- ☐ 0 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés **if-then-else**
- ☐ 3 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

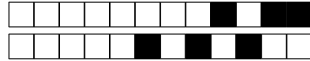
- ☐ toujours vérifié
- ☐ parfois vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ jamais vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
- ☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Java
- ☐ C
- ☐ Lisp
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ on ne peut pas
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ avec le compilateur
- ☐ en l'exécutant sur l'ordinateur

Question 4 En informatique, est-ce qu'un *type* ?

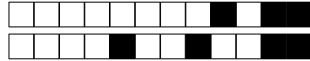
- ☐ une variable
- ☐ une personne
- ☐ une constante
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques

Question 5 ♣ En C, un entier peut être de type :

- ☐ int
- ☐ long
- ☐ très long
- ☐ double
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -35
- ☐ -34
- ☐ 100
- ☐ -1



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ pair
- ☐ négatif
- ☐ positif
- ☐ impair

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ -3 et 768
- ☐ aucune des deux
- ☐ 126 et 767
- ☐ 125 et 765

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont toujours exacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

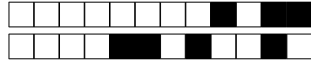
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de rendre le programme moins lisible
- ☐ de paramétrer les actions
- ☐ de localiser les objets
- ☐ de structurer le programme
- ☐ de rendre le programme plus complexe
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé E sont des affirmations qui doivent :

- ☐ être toujours fausses avant et après l'exécution de E
- ☐ de temps en temps vraies avant et après l'exécution de E
- ☐ être toujours vraies, respectivement, avant et après l'exécution de E
- ☐ être toujours vraies avant l'exécution de E

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ formels
- ☐ effectifs
- ☐ effaçables
- ☐ variables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie uniquement à la première itération
- ☐ jamais vraie
- ☐ vraie à chaque itération
- ☐ vraie uniquement à la dernière itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ la probabilité que a soit égal à b
- ☐ le produit de a par b
- ☐ le minimum de a et b
- ☐ le pgcd (plus grand commun diviseur) de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N
- ☐ -1 et N+1
- ☐ 0 et N-1
- ☐ 1 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés itératifs **for**
- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 1 seul énoncé itératif **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

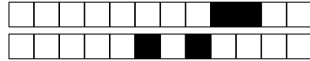
- ☐ parfois vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ toujours vérifié
- ☐ jamais vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016**Durée 1h30**

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Java
☐ Lisp
☐ C
☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
☐ avec le compilateur
☐ en l'exécutant sur l'ordinateur
☐ on ne peut pas

Question 4 En informatique, est-ce qu'un *type* ?

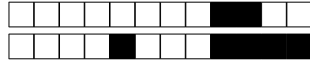
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
☐ une constante
☐ une personne
☐ une variable

Question 5 ♣ En C, un entier peut être de type :

- ☐ double
☐ long
☐ int
☐ très long
☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -34
☐ -35
☐ 100
☐ -1



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ pair
- ☐ positif
- ☐ impair
- ☐ négatif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ aucune des deux
- ☐ 125 et 765
- ☐ 126 et 767
- ☐ -3 et 768

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ les calculs avec les réels sont toujours exacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de localiser les objets
- ☐ de structurer le programme
- ☐ de rendre le programme plus complexe
- ☐ de rendre le programme moins lisible
- ☐ de paramétrer les actions
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effaçables
- ☐ effectifs
- ☐ formels
- ☐ variables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

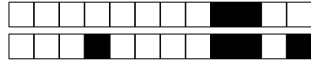
- ☐ vraie uniquement à la dernière itération
- ☐ vraie à chaque itération
- ☐ jamais vraie
- ☐ vraie uniquement à la première itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le minimum de a et b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le produit de a par b
- ☐ la probabilité que a soit égal à b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 1 et N
- ☐ 0 et N-1
- ☐ -1 et N+1
- ☐ 0 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 1 seul énoncé itératif **for**
- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 2 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ toujours vérifié
- ☐ jamais vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ parfois vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ elle ne fait rien et ne renvoie rien
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016**Durée 1h30**

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Java
☐ Lisp
☐ C
☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ on ne peut pas
☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
☐ en l'exécutant sur l'ordinateur
☐ avec le compilateur

Question 4 En informatique, est-ce qu'un *type* ?

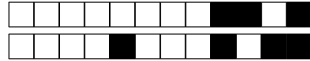
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
☐ une constante
☐ une variable
☐ une personne

Question 5 ♣ En C, un entier peut être de type :

- ☐ double
☐ long
☐ int
☐ très long
☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ 100
☐ -34
☐ -1
☐ -35



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ pair
- ☐ négatif
- ☐ impair
- ☐ positif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 126 et 767
- ☐ 125 et 765
- ☐ -3 et 768
- ☐ aucune des deux

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ les calculs avec les réels sont toujours exacts
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « formel donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ $x = 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de paramétrer les actions
- ☐ de localiser les objets
- ☐ de rendre le programme plus complexe
- ☐ de rendre le programme moins lisible
- ☐ de structurer le programme
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effectifs
- ☐ variables
- ☐ effaçables
- ☐ formels

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie à chaque itération
- ☐ vraie uniquement à la première itération
- ☐ jamais vraie
- ☐ vraie uniquement à la dernière itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le minimum de a et b
- ☐ la probabilité que a soit égal à b
- ☐ le produit de a par b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N
- ☐ 1 et N
- ☐ -1 et N+1
- ☐ 0 et N-1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 3 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (i.e. `t[i]`) est :

- ☐ parfois vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ jamais vérifié
- ☐ toujours vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
- ☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Java
- ☐ Lisp
- ☐ C
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ en l'exécutant sur l'ordinateur
- ☐ avec le compilateur
- ☐ on ne peut pas

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une personne
- ☐ une variable
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une constante

Question 5 ♣ En C, un entier peut être de type :

- ☐ très long
- ☐ double
- ☐ int
- ☐ long
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ 100
- ☐ -1
- ☐ -35
- ☐ -34



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ pair
- ☐ impair
- ☐ négatif
- ☐ positif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 126 et 767
- ☐ 125 et 765
- ☐ -3 et 768
- ☐ aucune des deux

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ les calculs avec les réels sont toujours exacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

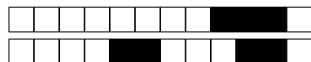
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « formel donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de localiser les objets
- ☐ de paramétrer les actions
- ☐ de structurer le programme
- ☐ de rendre le programme moins lisible
- ☐ de rendre le programme plus complexe
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effaçables
- ☐ effectifs
- ☐ formels
- ☐ variables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

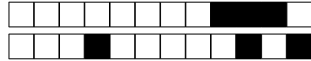
- ☐ jamais vraie
- ☐ vraie uniquement à la dernière itération
- ☐ vraie uniquement à la première itération
- ☐ vraie à chaque itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ la probabilité que a soit égal à b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le produit de a par b
- ☐ le minimum de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N-1
- ☐ 1 et N
- ☐ 0 et N
- ☐ -1 et N+1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ jamais vérifié
- ☐ parfois vérifié
- ☐ toujours vérifié
- ☐ est vérifié si l'ordinateur est en forme

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
- ☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ C
- ☐ Java
- ☐ Lisp
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ avec le compilateur
- ☐ on ne peut pas
- ☐ en l'exécutant sur l'ordinateur

Question 4 En informatique, est-ce qu'un *type* ?

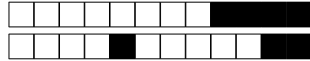
- ☐ une personne
- ☐ une constante
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une variable

Question 5 ♣ En C, un entier peut être de type :

- ☐ int
- ☐ long
- ☐ très long
- ☐ double
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ 100
- ☐ -34
- ☐ -35
- ☐ -1



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ impair
- ☐ positif
- ☐ négatif
- ☐ pair

Question 8 À quelle valeur décimale correspond les valeurs 0176 et 0x2FF ?

- ☐ 126 et 767
- ☐ 125 et 765
- ☐ aucune des deux
- ☐ -3 et 768

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont toujours exacts
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x \neq 0$
- ☐ $x \neq 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de paramétrer les actions
- ☐ de rendre le programme moins lisible
- ☐ de rendre le programme plus complexe
- ☐ de localiser les objets
- ☐ de structurer le programme
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ formels
- ☐ effectifs
- ☐ effaçables
- ☐ variables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

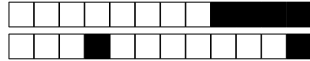
- ☐ jamais vraie
- ☐ vraie uniquement à la dernière itération
- ☐ vraie à chaque itération
- ☐ vraie uniquement à la première itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le minimum de a et b
- ☐ la probabilité que a soit égal à b
- ☐ le produit de a par b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 1 et N
- ☐ 0 et N
- ☐ -1 et N+1
- ☐ 0 et N-1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 1 seul énoncé itératif **for**
- ☐ 3 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ toujours vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ parfois vérifié
- ☐ jamais vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
- ☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Lisp
- ☐ Java
- ☐ C
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ en l'exécutant sur l'ordinateur
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ avec le compilateur
- ☐ on ne peut pas

Question 4 En informatique, est-ce qu'un *type* ?

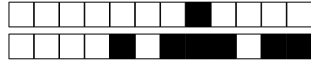
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une personne
- ☐ une variable
- ☐ une constante

Question 5 ♣ En C, un entier peut être de type :

- ☐ très long
- ☐ long
- ☐ double
- ☐ int
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -1
- ☐ 100
- ☐ -35
- ☐ -34



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ pair
- ☐ négatif
- ☐ impair
- ☐ positif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ -3 et 768
- ☐ aucune des deux
- ☐ 126 et 767
- ☐ 125 et 765

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont toujours exacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|a - b| < \epsilon$
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x = 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de paramétrer les actions
- ☐ de rendre le programme plus complexe
- ☐ de structurer le programme
- ☐ de localiser les objets
- ☐ de rendre le programme moins lisible
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effectifs
- ☐ formels
- ☐ effaçables
- ☐ variables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie à chaque itération
- ☐ vraie uniquement à la dernière itération
- ☐ vraie uniquement à la première itération
- ☐ jamais vraie

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le produit de a par b
- ☐ la probabilité que a soit égal à b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le minimum de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N
- ☐ 0 et N-1
- ☐ -1 et N+1
- ☐ 1 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 1 seul énoncé itératif **for**
- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**

Question 19

En C, à l'exécution, l'accès à un élément d'indice *i* (*i.e.* `t[i]`) est :

- ☐ toujours vérifié
- ☐ parfois vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ jamais vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
- ☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Java
- ☐ C
- ☐ Lisp
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ en l'exécutant sur l'ordinateur
- ☐ on ne peut pas
- ☐ avec le compilateur

Question 4 En informatique, est-ce qu'un *type* ?

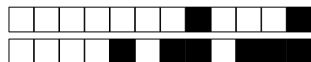
- ☐ une constante
- ☐ une personne
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une variable

Question 5 ♣ En C, un entier peut être de type :

- ☐ double
- ☐ très long
- ☐ long
- ☐ int
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -34
- ☐ -35
- ☐ 100
- ☐ -1



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ négatif
- ☐ impair
- ☐ positif
- ☐ pair

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 126 et 767
- ☐ 125 et 765
- ☐ -3 et 768
- ☐ aucune des deux

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ les calculs avec les réels sont toujours exacts
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée » et « résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x = 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de structurer le programme
- ☐ de paramétrer les actions
- ☐ de localiser les objets
- ☐ de rendre le programme plus complexe
- ☐ de rendre le programme moins lisible
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effectifs
- ☐ effaçables
- ☐ formels
- ☐ variables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie à chaque itération
- ☐ vraie uniquement à la dernière itération
- ☐ vraie uniquement à la première itération
- ☐ jamais vraie

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ la probabilité que a soit égal à b
- ☐ le minimum de a et b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le produit de a par b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N
- ☐ 1 et N
- ☐ -1 et N+1
- ☐ 0 et N-1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés **if-then-else**
- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés itératifs **for**
- ☐ 3 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ jamais vérifié
- ☐ toujours vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ parfois vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
- ☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ C
- ☐ Java
- ☐ Lisp
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ en l'exécutant sur l'ordinateur
- ☐ on ne peut pas
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ avec le compilateur

Question 4 En informatique, est-ce qu'un *type* ?

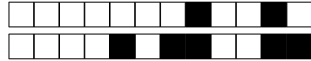
- ☐ une constante
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une personne
- ☐ une variable

Question 5 ♣ En C, un entier peut être de type :

- ☐ très long
- ☐ int
- ☐ double
- ☐ long
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ 100
- ☐ -35
- ☐ -1
- ☐ -34



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ impair
- ☐ pair
- ☐ négatif
- ☐ positif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ aucune des deux
- ☐ 126 et 767
- ☐ 125 et 765
- ☐ -3 et 768

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont toujours exacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « formel donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$
- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de rendre le programme plus complexe
- ☐ de paramétrer les actions
- ☐ de rendre le programme moins lisible
- ☐ de localiser les objets
- ☐ de structurer le programme
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effaçables
- ☐ variables
- ☐ formels
- ☐ effectifs

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie à chaque itération
- ☐ jamais vraie
- ☐ vraie uniquement à la première itération
- ☐ vraie uniquement à la dernière itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ la probabilité que a soit égal à b
- ☐ le produit de a par b
- ☐ le minimum de a et b
- ☐ le pgcd (plus grand commun diviseur) de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 1 et N
- ☐ -1 et N+1
- ☐ 0 et N-1
- ☐ 0 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 1 seul énoncé itératif **for**
- ☐ 3 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ parfois vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ toujours vérifié
- ☐ jamais vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
- ☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Lisp
- ☐ Java
- ☐ C
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ on ne peut pas
- ☐ avec le compilateur
- ☐ en l'exécutant sur l'ordinateur
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents

Question 4 En informatique, est-ce qu'un *type* ?

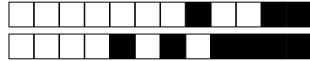
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une constante
- ☐ une personne
- ☐ une variable

Question 5 ♣ En C, un entier peut être de type :

- ☐ int
- ☐ long
- ☐ très long
- ☐ double
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ 100
- ☐ -34
- ☐ -35
- ☐ -1



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ pair
- ☐ positif
- ☐ impair
- ☐ négatif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ -3 et 768
- ☐ aucune des deux
- ☐ 125 et 765
- ☐ 126 et 767

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont toujours exacts
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « formel donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de rendre le programme moins lisible
- ☐ de localiser les objets
- ☐ de paramétrer les actions
- ☐ de structurer le programme
- ☐ de rendre le programme plus complexe
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ formels
- ☐ effectifs
- ☐ variables
- ☐ effaçables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

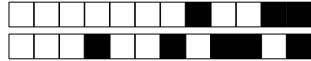
- ☐ vraie uniquement à la première itération
- ☐ vraie uniquement à la dernière itération
- ☐ vraie à chaque itération
- ☐ jamais vraie

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le minimum de a et b
- ☐ le produit de a par b
- ☐ la probabilité que a soit égal à b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ -1 et N+1
- ☐ 0 et N
- ☐ 1 et N
- ☐ 0 et N-1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés **if-then-else**
- ☐ 1 seul énoncé itératif **for**
- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

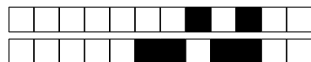
- ☐ jamais vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ toujours vérifié
- ☐ parfois vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
- ☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Java
- ☐ Lisp
- ☐ C
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ avec le compilateur
- ☐ en l'exécutant sur l'ordinateur
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ on ne peut pas

Question 4 En informatique, est-ce qu'un *type* ?

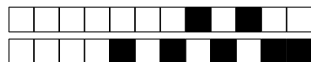
- ☐ une personne
- ☐ une constante
- ☐ une variable
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques

Question 5 ♣ En C, un entier peut être de type :

- ☐ long
- ☐ très long
- ☐ double
- ☐ int
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -34
- ☐ -1
- ☐ -35
- ☐ 100



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ impair
- ☐ positif
- ☐ négatif
- ☐ pair

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 126 et 767
- ☐ aucune des deux
- ☐ -3 et 768
- ☐ 125 et 765

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont toujours exacts
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée » et « résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de structurer le programme
- ☐ de localiser les objets
- ☐ de rendre le programme plus complexe
- ☐ de rendre le programme moins lisible
- ☐ de paramétrer les actions
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ variables
- ☐ effectifs
- ☐ formels
- ☐ effaçables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

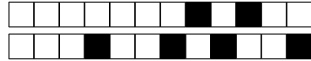
- ☐ jamais vraie
- ☐ vraie uniquement à la dernière itération
- ☐ vraie uniquement à la première itération
- ☐ vraie à chaque itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le minimum de a et b
- ☐ la probabilité que a soit égal à b
- ☐ le produit de a par b
- ☐ le pgcd (plus grand commun diviseur) de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 1 et N
- ☐ 0 et N
- ☐ 0 et N-1
- ☐ -1 et N+1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés **if-then-else**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (i.e. `t[i]`) est :

- ☐ est vérifié si l'ordinateur est en forme
- ☐ jamais vérifié
- ☐ toujours vérifié
- ☐ parfois vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016**Durée 1h30**

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Lisp
☐ Java
☐ C
☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ on ne peut pas
☐ avec le compilateur
☐ en l'exécutant sur l'ordinateur
☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents

Question 4 En informatique, est-ce qu'un *type* ?

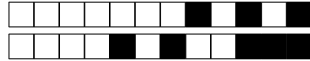
- ☐ une personne
☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
☐ une constante
☐ une variable

Question 5 ♣ En C, un entier peut être de type :

- ☐ double
☐ int
☐ long
☐ très long
☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -35
☐ -1
☐ -34
☐ 100



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ pair
- ☐ impair
- ☐ négatif
- ☐ positif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ -3 et 768
- ☐ 126 et 767
- ☐ 125 et 765
- ☐ aucune des deux

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont toujours exacts
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x \neq 0$
- ☐ $x \neq 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de paramétrer les actions
- ☐ de localiser les objets
- ☐ de structurer le programme
- ☐ de rendre le programme moins lisible
- ☐ de rendre le programme plus complexe
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ variables
- ☐ effectifs
- ☐ effaçables
- ☐ formels

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

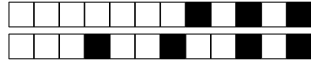
- ☐ vraie uniquement à la première itération
- ☐ vraie uniquement à la dernière itération
- ☐ jamais vraie
- ☐ vraie à chaque itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le produit de a par b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le minimum de a et b
- ☐ la probabilité que a soit égal à b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 1 et N
- ☐ 0 et N-1
- ☐ 0 et N
- ☐ -1 et N+1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés itératifs **for**
- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ jamais vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ parfois vérifié
- ☐ toujours vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
- ☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ C
- ☐ Java
- ☐ Lisp
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ on ne peut pas
- ☐ avec le compilateur
- ☐ en l'exécutant sur l'ordinateur

Question 4 En informatique, est-ce qu'un *type* ?

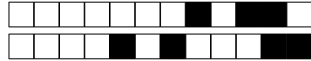
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une constante
- ☐ une variable
- ☐ une personne

Question 5 ♣ En C, un entier peut être de type :

- ☐ double
- ☐ très long
- ☐ int
- ☐ long
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -35
- ☐ 100
- ☐ -1
- ☐ -34



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ positif
- ☐ impair
- ☐ pair
- ☐ négatif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 126 et 767
- ☐ aucune des deux
- ☐ -3 et 768
- ☐ 125 et 765

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont toujours exacts
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$
- ☐ $x = 0$ et $(x \neq 1 \text{ ou } y \neq 0)$
- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de rendre le programme plus complexe
- ☐ de paramétrer les actions
- ☐ de localiser les objets
- ☐ de rendre le programme moins lisible
- ☐ de structurer le programme
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé E sont des affirmations qui doivent :

- ☐ être toujours vraies avant l'exécution de E
- ☐ de temps en temps vraies avant et après l'exécution de E
- ☐ être toujours vraies, respectivement, avant et après l'exécution de E
- ☐ être toujours fausses avant et après l'exécution de E

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effaçables
- ☐ effectifs
- ☐ variables
- ☐ formels

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

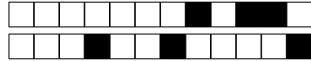
- ☐ vraie à chaque itération
- ☐ vraie uniquement à la première itération
- ☐ vraie uniquement à la dernière itération
- ☐ jamais vraie

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le minimum de a et b
- ☐ la probabilité que a soit égal à b
- ☐ le produit de a par b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ -1 et N+1
- ☐ 0 et N
- ☐ 1 et N
- ☐ 0 et N-1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés **if-then-else**
- ☐ 3 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ parfois vérifié
- ☐ jamais vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ toujours vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
- ☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ C
- ☐ Java
- ☐ Lisp
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ en l'exécutant sur l'ordinateur
- ☐ on ne peut pas
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ avec le compilateur

Question 4 En informatique, est-ce qu'un *type* ?

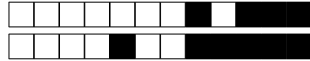
- ☐ une variable
- ☐ une constante
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une personne

Question 5 ♣ En C, un entier peut être de type :

- ☐ int
- ☐ double
- ☐ long
- ☐ très long
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -1
- ☐ 100
- ☐ -35
- ☐ -34



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ pair
- ☐ positif
- ☐ impair
- ☐ négatif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ aucune des deux
- ☐ -3 et 768
- ☐ 126 et 767
- ☐ 125 et 765

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont toujours exacts
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

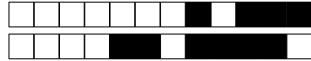
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « formel donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x \neq 0$
- ☐ $x = 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de paramétrer les actions
- ☐ de rendre le programme moins lisible
- ☐ de structurer le programme
- ☐ de localiser les objets
- ☐ de rendre le programme plus complexe
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effectifs
- ☐ variables
- ☐ effaçables
- ☐ formels

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

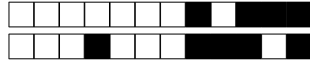
- ☐ vraie à chaque itération
- ☐ vraie uniquement à la dernière itération
- ☐ jamais vraie
- ☐ vraie uniquement à la première itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le produit de a par b
- ☐ la probabilité que a soit égal à b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le minimum de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N
- ☐ 1 et N
- ☐ -1 et N+1
- ☐ 0 et N-1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 3 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

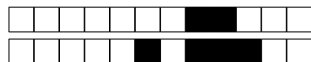
- ☐ est vérifié si l'ordinateur est en forme
- ☐ parfois vérifié
- ☐ jamais vérifié
- ☐ toujours vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
- ☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Lisp
- ☐ Java
- ☐ C
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ on ne peut pas
- ☐ avec le compilateur
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ en l'exécutant sur l'ordinateur

Question 4 En informatique, est-ce qu'un *type* ?

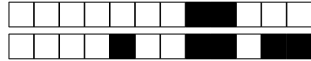
- ☐ une personne
- ☐ une constante
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une variable

Question 5 ♣ En C, un entier peut être de type :

- ☐ double
- ☐ très long
- ☐ int
- ☐ long
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -34
- ☐ -1
- ☐ -35
- ☐ 100



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ pair
- ☐ positif
- ☐ négatif
- ☐ impair

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 125 et 765
- ☐ aucune des deux
- ☐ -3 et 768
- ☐ 126 et 767

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ les calculs avec les réels sont toujours exacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

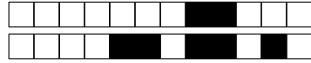
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « formel donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x \neq 0$
- ☐ $x \neq 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de rendre le programme moins lisible
- ☐ de localiser les objets
- ☐ de rendre le programme plus complexe
- ☐ de structurer le programme
- ☐ de paramétrer les actions
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effaçables
- ☐ variables
- ☐ effectifs
- ☐ formels

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie uniquement à la dernière itération
- ☐ vraie à chaque itération
- ☐ jamais vraie
- ☐ vraie uniquement à la première itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le minimum de a et b
- ☐ le produit de a par b
- ☐ la probabilité que a soit égal à b
- ☐ le pgcd (plus grand commun diviseur) de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N-1
- ☐ 0 et N
- ☐ -1 et N+1
- ☐ 1 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés **if-then-else**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

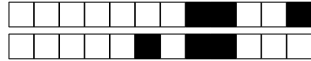
- ☐ toujours vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ parfois vérifié
- ☐ jamais vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ elle ne fait rien et ne renvoie rien
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016**Durée 1h30**

Nom :

Prénom :

ID :

*Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.**Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.***Question 1**

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Java
☐ Lisp
☐ C
☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ en l'exécutant sur l'ordinateur
☐ on ne peut pas
☐ avec le compilateur
☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents

Question 4 En informatique, est-ce qu'un *type* ?

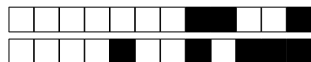
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
☐ une variable
☐ une personne
☐ une constante

Question 5 ♣ En C, un entier peut être de type :

- ☐ double
☐ très long
☐ int
☐ long
☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -34
☐ -1
☐ 100
☐ -35



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ pair
- ☐ positif
- ☐ négatif
- ☐ impair

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ aucune des deux
- ☐ -3 et 768
- ☐ 126 et 767
- ☐ 125 et 765

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont toujours exacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

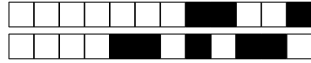
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et $(x \neq 1 \text{ ou } y \neq 0)$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de localiser les objets
- ☐ de paramétrer les actions
- ☐ de rendre le programme plus complexe
- ☐ de structurer le programme
- ☐ de rendre le programme moins lisible
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ formels
- ☐ effectifs
- ☐ effaçables
- ☐ variables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie uniquement à la dernière itération
- ☐ jamais vraie
- ☐ vraie à chaque itération
- ☐ vraie uniquement à la première itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le produit de a par b
- ☐ le minimum de a et b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ la probabilité que a soit égal à b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ -1 et N+1
- ☐ 1 et N
- ☐ 0 et N-1
- ☐ 0 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés **if-then-else**
- ☐ 2 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**
- ☐ 3 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

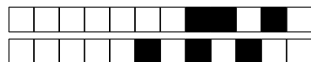
- ☐ est vérifié si l'ordinateur est en forme
- ☐ toujours vérifié
- ☐ jamais vérifié
- ☐ parfois vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
- ☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Java
- ☐ C
- ☐ Lisp
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ en l'exécutant sur l'ordinateur
- ☐ on ne peut pas
- ☐ avec le compilateur
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents

Question 4 En informatique, est-ce qu'un *type* ?

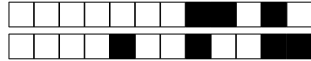
- ☐ une constante
- ☐ une variable
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une personne

Question 5 ♣ En C, un entier peut être de type :

- ☐ double
- ☐ très long
- ☐ long
- ☐ int
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -35
- ☐ -1
- ☐ -34
- ☐ 100



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ pair
- ☐ négatif
- ☐ positif
- ☐ impair

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ -3 et 768
- ☐ 125 et 765
- ☐ 126 et 767
- ☐ aucune des deux

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont toujours exacts
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

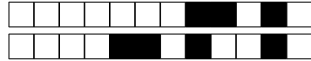
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif donnée » et « résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de localiser les objets
- ☐ de rendre le programme plus complexe
- ☐ de structurer le programme
- ☐ de rendre le programme moins lisible
- ☐ de paramétrer les actions
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ variables
- ☐ effaçables
- ☐ formels
- ☐ effectifs

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie à chaque itération
- ☐ jamais vraie
- ☐ vraie uniquement à la dernière itération
- ☐ vraie uniquement à la première itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le produit de a par b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ la probabilité que a soit égal à b
- ☐ le minimum de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N-1
- ☐ 0 et N
- ☐ -1 et N+1
- ☐ 1 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

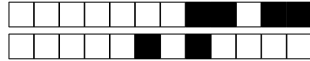
- ☐ toujours vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ parfois vérifié
- ☐ jamais vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
- ☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Java
- ☐ Lisp
- ☐ C
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ en l'exécutant sur l'ordinateur
- ☐ avec le compilateur
- ☐ on ne peut pas

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une personne
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une constante
- ☐ une variable

Question 5 ♣ En C, un entier peut être de type :

- ☐ double
- ☐ long
- ☐ très long
- ☐ int
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -1
- ☐ -34
- ☐ -35
- ☐ 100



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ impair
- ☐ positif
- ☐ négatif
- ☐ pair

Question 8 À quelle valeur décimale correspond les valeurs 0176 et 0x2FF ?

- ☐ 125 et 765
- ☐ aucune des deux
- ☐ -3 et 768
- ☐ 126 et 767

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont toujours exacts
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

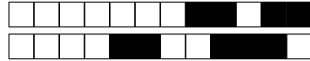
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « formel donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de rendre le programme moins lisible
- ☐ de localiser les objets
- ☐ de paramétrer les actions
- ☐ de structurer le programme
- ☐ de rendre le programme plus complexe
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effectifs
- ☐ variables
- ☐ effaçables
- ☐ formels

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

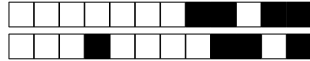
- ☐ vraie uniquement à la première itération
- ☐ jamais vraie
- ☐ vraie à chaque itération
- ☐ vraie uniquement à la dernière itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ la probabilité que a soit égal à b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le produit de a par b
- ☐ le minimum de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ -1 et N+1
- ☐ 1 et N
- ☐ 0 et N-1
- ☐ 0 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 3 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 2 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

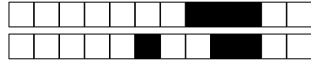
- ☐ jamais vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ toujours vérifié
- ☐ parfois vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016**Durée 1h30**

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Java
☐ C
☐ Lisp
☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ avec le compilateur
☐ on ne peut pas
☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
☐ en l'exécutant sur l'ordinateur

Question 4 En informatique, est-ce qu'un *type* ?

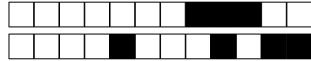
- ☐ une variable
☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
☐ une personne
☐ une constante

Question 5 ♣ En C, un entier peut être de type :

- ☐ int
☐ long
☐ très long
☐ double
☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -34
☐ 100
☐ -35
☐ -1



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ pair
- ☐ impair
- ☐ positif
- ☐ négatif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ -3 et 768
- ☐ 125 et 765
- ☐ 126 et 767
- ☐ aucune des deux

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ les calculs avec les réels sont toujours exacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

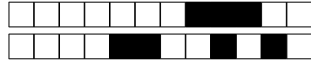
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée » et « résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de rendre le programme moins lisible
- ☐ de rendre le programme plus complexe
- ☐ de paramétrer les actions
- ☐ de structurer le programme
- ☐ de localiser les objets
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effaçables
- ☐ variables
- ☐ formels
- ☐ effectifs

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie uniquement à la première itération
- ☐ vraie à chaque itération
- ☐ jamais vraie
- ☐ vraie uniquement à la dernière itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ la probabilité que a soit égal à b
- ☐ le minimum de a et b
- ☐ le produit de a par b
- ☐ le pgcd (plus grand commun diviseur) de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 1 et N
- ☐ -1 et N+1
- ☐ 0 et N-1
- ☐ 0 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 1 seul énoncé itératif **for**
- ☐ 3 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

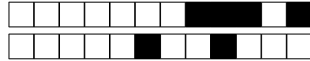
- ☐ jamais vérifié
- ☐ toujours vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ parfois vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016**Durée 1h30**

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Java
☐ C
☐ Lisp
☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ on ne peut pas
☐ en l'exécutant sur l'ordinateur
☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
☐ avec le compilateur

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une variable
☐ une constante
☐ une personne
☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques

Question 5 ♣ En C, un entier peut être de type :

- ☐ double
☐ très long
☐ int
☐ long
☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ 100
☐ -35
☐ -1
☐ -34



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ négatif
- ☐ pair
- ☐ impair
- ☐ positif

Question 8 À quelle valeur décimale correspond les valeurs 0176 et 0x2FF ?

- ☐ 126 et 767
- ☐ aucune des deux
- ☐ 125 et 765
- ☐ -3 et 768

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont toujours exacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

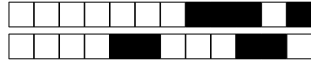
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « formel donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ $x \neq 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de paramétrer les actions
- ☐ de localiser les objets
- ☐ de rendre le programme plus complexe
- ☐ de structurer le programme
- ☐ de rendre le programme moins lisible
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effectifs
- ☐ formels
- ☐ variables
- ☐ effaçables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie uniquement à la première itération
- ☐ vraie uniquement à la dernière itération
- ☐ vraie à chaque itération
- ☐ jamais vraie

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le minimum de a et b
- ☐ la probabilité que a soit égal à b
- ☐ le produit de a par b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N-1
- ☐ 1 et N
- ☐ 0 et N
- ☐ -1 et N+1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés **if-then-else**
- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

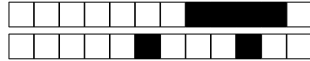
- ☐ parfois vérifié
- ☐ jamais vérifié
- ☐ toujours vérifié
- ☐ est vérifié si l'ordinateur est en forme

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
- ☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ C
- ☐ Java
- ☐ Lisp
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ on ne peut pas
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ en l'exécutant sur l'ordinateur
- ☐ avec le compilateur

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une constante
- ☐ une personne
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une variable

Question 5 ♣ En C, un entier peut être de type :

- ☐ int
- ☐ double
- ☐ très long
- ☐ long
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -35
- ☐ -1
- ☐ -34
- ☐ 100



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ positif
- ☐ pair
- ☐ impair
- ☐ négatif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 126 et 767
- ☐ 125 et 765
- ☐ aucune des deux
- ☐ -3 et 768

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont toujours exacts
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

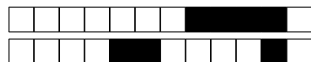
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « formel donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de paramétrer les actions
- ☐ de rendre le programme moins lisible
- ☐ de rendre le programme plus complexe
- ☐ de structurer le programme
- ☐ de localiser les objets
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours vraies avant l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effaçables
- ☐ variables
- ☐ effectifs
- ☐ formels

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ jamais vraie
- ☐ vraie uniquement à la première itération
- ☐ vraie à chaque itération
- ☐ vraie uniquement à la dernière itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le produit de a par b
- ☐ la probabilité que a soit égal à b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le minimum de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 1 et N
- ☐ 0 et N
- ☐ 0 et N-1
- ☐ -1 et N+1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés **if-then-else**
- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés itératifs **for**
- ☐ 3 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ jamais vérifié
- ☐ parfois vérifié
- ☐ toujours vérifié
- ☐ est vérifié si l'ordinateur est en forme

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
- ☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Lisp
- ☐ C
- ☐ Java
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ en l'exécutant sur l'ordinateur
- ☐ on ne peut pas
- ☐ avec le compilateur

Question 4 En informatique, est-ce qu'un *type* ?

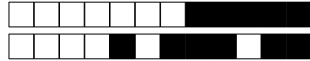
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une variable
- ☐ une personne
- ☐ une constante

Question 5 ♣ En C, un entier peut être de type :

- ☐ int
- ☐ double
- ☐ long
- ☐ très long
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ 100
- ☐ -34
- ☐ -1
- ☐ -35



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ négatif
- ☐ pair
- ☐ positif
- ☐ impair

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ -3 et 768
- ☐ 125 et 765
- ☐ aucune des deux
- ☐ 126 et 767

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont toujours exacts
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « formel donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et $(x \neq 1 \text{ ou } y \neq 0)$
- ☐ $x \neq 0$
- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de localiser les objets
- ☐ de paramétrer les actions
- ☐ de structurer le programme
- ☐ de rendre le programme moins lisible
- ☐ de rendre le programme plus complexe
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ formels
- ☐ effaçables
- ☐ variables
- ☐ effectifs

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie uniquement à la première itération
- ☐ vraie uniquement à la dernière itération
- ☐ jamais vraie
- ☐ vraie à chaque itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ la probabilité que a soit égal à b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le minimum de a et b
- ☐ le produit de a par b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 1 et N
- ☐ 0 et N
- ☐ 0 et N-1
- ☐ -1 et N+1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 1 seul énoncé itératif **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ parfois vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ toujours vérifié
- ☐ jamais vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
- ☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ C
- ☐ Lisp
- ☐ Java
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ on ne peut pas
- ☐ en l'exécutant sur l'ordinateur
- ☐ avec le compilateur

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une constante
- ☐ une variable
- ☐ une personne

Question 5 ♣ En C, un entier peut être de type :

- ☐ très long
- ☐ long
- ☐ int
- ☐ double
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -1
- ☐ -34
- ☐ -35
- ☐ 100



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ impair
- ☐ positif
- ☐ pair
- ☐ négatif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ aucune des deux
- ☐ 126 et 767
- ☐ -3 et 768
- ☐ 125 et 765

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont toujours exacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « formel donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$
- ☐ $x = 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x \neq 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de paramétrer les actions
- ☐ de rendre le programme moins lisible
- ☐ de localiser les objets
- ☐ de structurer le programme
- ☐ de rendre le programme plus complexe
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ variables
- ☐ formels
- ☐ effectifs
- ☐ effaçables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ jamais vraie
- ☐ vraie uniquement à la dernière itération
- ☐ vraie à chaque itération
- ☐ vraie uniquement à la première itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le minimum de a et b
- ☐ la probabilité que a soit égal à b
- ☐ le produit de a par b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ -1 et N+1
- ☐ 1 et N
- ☐ 0 et N-1
- ☐ 0 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés itératifs **for**
- ☐ 3 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés **if-then-else**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ toujours vérifié
- ☐ parfois vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ jamais vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016**Durée 1h30**

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ C
☐ Lisp
☐ Java
☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ en l'exécutant sur l'ordinateur
☐ on ne peut pas
☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
☐ avec le compilateur

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une variable
☐ une personne
☐ une constante
☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques

Question 5 ♣ En C, un entier peut être de type :

- ☐ int
☐ double
☐ long
☐ très long
☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -1
☐ 100
☐ -34
☐ -35



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ impair
- ☐ négatif
- ☐ pair
- ☐ positif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 125 et 765
- ☐ 126 et 767
- ☐ -3 et 768
- ☐ aucune des deux

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont toujours exacts
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « formel donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et $(x \neq 1 \text{ ou } y \neq 0)$
- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de rendre le programme moins lisible
- ☐ de localiser les objets
- ☐ de structurer le programme
- ☐ de paramétrer les actions
- ☐ de rendre le programme plus complexe
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ formels
- ☐ variables
- ☐ effaçables
- ☐ effectifs

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ jamais vraie
- ☐ vraie uniquement à la dernière itération
- ☐ vraie uniquement à la première itération
- ☐ vraie à chaque itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ la probabilité que a soit égal à b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le produit de a par b
- ☐ le minimum de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N-1
- ☐ -1 et N+1
- ☐ 1 et N
- ☐ 0 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 3 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ est vérifié si l'ordinateur est en forme
- ☐ jamais vérifié
- ☐ parfois vérifié
- ☐ toujours vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ elle renvoie 1 si `x` $\in$ `t` et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
- ☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ C
- ☐ Lisp
- ☐ Java
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ en l'exécutant sur l'ordinateur
- ☐ avec le compilateur
- ☐ on ne peut pas

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une variable
- ☐ une personne
- ☐ une constante

Question 5 ♣ En C, un entier peut être de type :

- ☐ long
- ☐ int
- ☐ double
- ☐ très long
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -34
- ☐ -35
- ☐ 100
- ☐ -1



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ positif
- ☐ impair
- ☐ négatif
- ☐ pair

Question 8 À quelle valeur décimale correspond les valeurs 0176 et 0x2FF ?

- ☐ 125 et 765
- ☐ 126 et 767
- ☐ aucune des deux
- ☐ -3 et 768

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont toujours exacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ $x = 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de paramétrer les actions
- ☐ de rendre le programme plus complexe
- ☐ de rendre le programme moins lisible
- ☐ de structurer le programme
- ☐ de localiser les objets
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effaçables
- ☐ variables
- ☐ formels
- ☐ effectifs

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie uniquement à la dernière itération
- ☐ jamais vraie
- ☐ vraie uniquement à la première itération
- ☐ vraie à chaque itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le minimum de a et b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le produit de a par b
- ☐ la probabilité que a soit égal à b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N
- ☐ -1 et N+1
- ☐ 1 et N
- ☐ 0 et N-1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 1 seul énoncé itératif **for**
- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ toujours vérifié
- ☐ jamais vérifié
- ☐ parfois vérifié
- ☐ est vérifié si l'ordinateur est en forme

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
- ☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Java
- ☐ Lisp
- ☐ C
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ avec le compilateur
- ☐ en l'exécutant sur l'ordinateur
- ☐ on ne peut pas
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents

Question 4 En informatique, est-ce qu'un *type* ?

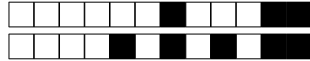
- ☐ une variable
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une personne
- ☐ une constante

Question 5 ♣ En C, un entier peut être de type :

- ☐ double
- ☐ int
- ☐ long
- ☐ très long
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -34
- ☐ -35
- ☐ -1
- ☐ 100



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ positif
- ☐ pair
- ☐ impair
- ☐ négatif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 125 et 765
- ☐ -3 et 768
- ☐ aucune des deux
- ☐ 126 et 767

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont toujours exacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif donnée » et « résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x = 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de paramétrer les actions
- ☐ de structurer le programme
- ☐ de rendre le programme moins lisible
- ☐ de rendre le programme plus complexe
- ☐ de localiser les objets
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effaçables
- ☐ effectifs
- ☐ variables
- ☐ formels

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie uniquement à la première itération
- ☐ vraie uniquement à la dernière itération
- ☐ vraie à chaque itération
- ☐ jamais vraie

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le produit de a par b
- ☐ la probabilité que a soit égal à b
- ☐ le minimum de a et b
- ☐ le pgcd (plus grand commun diviseur) de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N
- ☐ 0 et N-1
- ☐ -1 et N+1
- ☐ 1 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés itératifs **for**
- ☐ 3 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés **if-then-else**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ parfois vérifié
- ☐ jamais vérifié
- ☐ toujours vérifié
- ☐ est vérifié si l'ordinateur est en forme

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
- ☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Lisp
- ☐ C
- ☐ Java
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ avec le compilateur
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ on ne peut pas
- ☐ en l'exécutant sur l'ordinateur

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une personne
- ☐ une constante
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une variable

Question 5 ♣ En C, un entier peut être de type :

- ☐ double
- ☐ int
- ☐ long
- ☐ très long
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -35
- ☐ -1
- ☐ 100
- ☐ -34



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ positif
- ☐ pair
- ☐ impair
- ☐ négatif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 125 et 765
- ☐ -3 et 768
- ☐ 126 et 767
- ☐ aucune des deux

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont toujours exacts
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée » et « résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de rendre le programme moins lisible
- ☐ de rendre le programme plus complexe
- ☐ de paramétrer les actions
- ☐ de localiser les objets
- ☐ de structurer le programme
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ formels
- ☐ effectifs
- ☐ effaçables
- ☐ variables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ jamais vraie
- ☐ vraie à chaque itération
- ☐ vraie uniquement à la dernière itération
- ☐ vraie uniquement à la première itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le produit de a par b
- ☐ le minimum de a et b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ la probabilité que a soit égal à b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ -1 et N+1
- ☐ 0 et N
- ☐ 1 et N
- ☐ 0 et N-1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 3 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 2 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ parfois vérifié
- ☐ jamais vérifié
- ☐ toujours vérifié
- ☐ est vérifié si l'ordinateur est en forme

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016**Durée 1h30**

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ C
☐ Java
☐ Lisp
☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ on ne peut pas
☐ en l'exécutant sur l'ordinateur
☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
☐ avec le compilateur

Question 4 En informatique, est-ce qu'un *type* ?

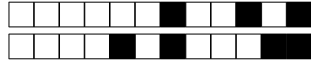
- ☐ une constante
☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
☐ une personne
☐ une variable

Question 5 ♣ En C, un entier peut être de type :

- ☐ int
☐ double
☐ très long
☐ long
☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -34
☐ -35
☐ 100
☐ -1



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ pair
- ☐ positif
- ☐ impair
- ☐ négatif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ -3 et 768
- ☐ 126 et 767
- ☐ aucune des deux
- ☐ 125 et 765

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont toujours exacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|a - b| < \epsilon$
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée » et « résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$
- ☐ $x = 0$ et $(x \neq 1 \text{ ou } y \neq 0)$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de localiser les objets
- ☐ de rendre le programme moins lisible
- ☐ de paramétrer les actions
- ☐ de structurer le programme
- ☐ de rendre le programme plus complexe
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ variables
- ☐ formels
- ☐ effaçables
- ☐ effectifs

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie à chaque itération
- ☐ jamais vraie
- ☐ vraie uniquement à la dernière itération
- ☐ vraie uniquement à la première itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le minimum de a et b
- ☐ le produit de a par b
- ☐ la probabilité que a soit égal à b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ -1 et N+1
- ☐ 0 et N-1
- ☐ 1 et N
- ☐ 0 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés **if-then-else**
- ☐ 3 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ est vérifié si l'ordinateur est en forme
- ☐ toujours vérifié
- ☐ jamais vérifié
- ☐ parfois vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ elle renvoie 1 si `x` $\in$ `t` et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016**Durée 1h30**

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ C
☐ Lisp
☐ Java
☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ on ne peut pas
☐ avec le compilateur
☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
☐ en l'exécutant sur l'ordinateur

Question 4 En informatique, est-ce qu'un *type* ?

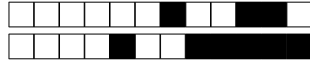
- ☐ une constante
☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
☐ une personne
☐ une variable

Question 5 ♣ En C, un entier peut être de type :

- ☐ long
☐ très long
☐ double
☐ int
☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ 100
☐ -1
☐ -35
☐ -34



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ impair
- ☐ pair
- ☐ positif
- ☐ négatif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 125 et 765
- ☐ -3 et 768
- ☐ aucune des deux
- ☐ 126 et 767

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont toujours exacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée » et « résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et $(x \neq 1 \text{ ou } y \neq 0)$
- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ $x \neq 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de localiser les objets
- ☐ de structurer le programme
- ☐ de paramétrer les actions
- ☐ de rendre le programme plus complexe
- ☐ de rendre le programme moins lisible
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ variables
- ☐ effectifs
- ☐ formels
- ☐ effaçables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie à chaque itération
- ☐ vraie uniquement à la première itération
- ☐ vraie uniquement à la dernière itération
- ☐ jamais vraie

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ la probabilité que a soit égal à b
- ☐ le minimum de a et b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le produit de a par b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N
- ☐ -1 et N+1
- ☐ 1 et N
- ☐ 0 et N-1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés itératifs **for**
- ☐ 3 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés **if-then-else**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ toujours vérifié
- ☐ parfois vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ jamais vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016**Durée 1h30**

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ C
☐ Java
☐ Lisp
☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ en l'exécutant sur l'ordinateur
☐ avec le compilateur
☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
☐ on ne peut pas

Question 4 En informatique, est-ce qu'un *type* ?

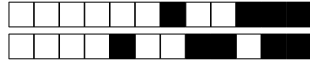
- ☐ une constante
☐ une variable
☐ une personne
☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques

Question 5 ♣ En C, un entier peut être de type :

- ☐ très long
☐ double
☐ int
☐ long
☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -35
☐ -34
☐ 100
☐ -1



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ pair
- ☐ impair
- ☐ positif
- ☐ négatif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 125 et 765
- ☐ -3 et 768
- ☐ aucune des deux
- ☐ 126 et 767

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont toujours exacts
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « formel donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$
- ☐ $x = 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x \neq 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de paramétrer les actions
- ☐ de structurer le programme
- ☐ de localiser les objets
- ☐ de rendre le programme plus complexe
- ☐ de rendre le programme moins lisible
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ variables
- ☐ effectifs
- ☐ formels
- ☐ effaçables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

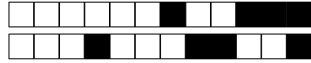
- ☐ vraie à chaque itération
- ☐ vraie uniquement à la première itération
- ☐ jamais vraie
- ☐ vraie uniquement à la dernière itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le minimum de a et b
- ☐ la probabilité que a soit égal à b
- ☐ le produit de a par b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N-1
- ☐ 1 et N
- ☐ -1 et N+1
- ☐ 0 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 3 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ jamais vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ parfois vérifié
- ☐ toujours vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
- ☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ C
- ☐ Lisp
- ☐ Java
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ avec le compilateur
- ☐ on ne peut pas
- ☐ en l'exécutant sur l'ordinateur
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une personne
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une constante
- ☐ une variable

Question 5 ♣ En C, un entier peut être de type :

- ☐ double
- ☐ int
- ☐ long
- ☐ très long
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -35
- ☐ -34
- ☐ 100
- ☐ -1



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ négatif
- ☐ impair
- ☐ positif
- ☐ pair

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 126 et 767
- ☐ aucune des deux
- ☐ -3 et 768
- ☐ 125 et 765

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ les calculs avec les réels sont toujours exacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée » et « résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de structurer le programme
- ☐ de rendre le programme moins lisible
- ☐ de localiser les objets
- ☐ de paramétrer les actions
- ☐ de rendre le programme plus complexe
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effaçables
- ☐ variables
- ☐ formels
- ☐ effectifs

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ jamais vraie
- ☐ vraie uniquement à la première itération
- ☐ vraie uniquement à la dernière itération
- ☐ vraie à chaque itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le minimum de a et b
- ☐ la probabilité que a soit égal à b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le produit de a par b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N-1
- ☐ 0 et N
- ☐ 1 et N
- ☐ -1 et N+1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés **if-then-else**
- ☐ 2 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**
- ☐ 3 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ jamais vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ toujours vérifié
- ☐ parfois vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ elle renvoie 1 si `x` $\in$ `t` et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
- ☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Java
- ☐ Lisp
- ☐ C
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ en l'exécutant sur l'ordinateur
- ☐ avec le compilateur
- ☐ on ne peut pas

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une personne
- ☐ une variable
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une constante

Question 5 ♣ En C, un entier peut être de type :

- ☐ long
- ☐ int
- ☐ très long
- ☐ double
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ 100
- ☐ -1
- ☐ -34
- ☐ -35



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ négatif
- ☐ pair
- ☐ positif
- ☐ impair

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ -3 et 768
- ☐ 126 et 767
- ☐ 125 et 765
- ☐ aucune des deux

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont toujours exacts
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif donnée » et « résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de localiser les objets
- ☐ de structurer le programme
- ☐ de rendre le programme plus complexe
- ☐ de paramétrer les actions
- ☐ de rendre le programme moins lisible
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effaçables
- ☐ effectifs
- ☐ variables
- ☐ formels

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

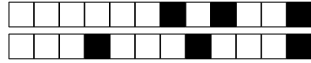
- ☐ vraie à chaque itération
- ☐ vraie uniquement à la dernière itération
- ☐ vraie uniquement à la première itération
- ☐ jamais vraie

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le minimum de a et b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le produit de a par b
- ☐ la probabilité que a soit égal à b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N-1
- ☐ 0 et N
- ☐ -1 et N+1
- ☐ 1 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 3 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ est vérifié si l'ordinateur est en forme
- ☐ parfois vérifié
- ☐ toujours vérifié
- ☐ jamais vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ elle renvoie 1 si `x` $\in$ `t` et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
- ☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ C
- ☐ Java
- ☐ Lisp
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ on ne peut pas
- ☐ en l'exécutant sur l'ordinateur
- ☐ avec le compilateur

Question 4 En informatique, est-ce qu'un *type* ?

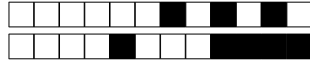
- ☐ une variable
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une constante
- ☐ une personne

Question 5 ♣ En C, un entier peut être de type :

- ☐ int
- ☐ très long
- ☐ double
- ☐ long
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -35
- ☐ -1
- ☐ -34
- ☐ 100



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ positif
- ☐ pair
- ☐ impair
- ☐ négatif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 125 et 765
- ☐ aucune des deux
- ☐ 126 et 767
- ☐ -3 et 768

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont toujours exacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de rendre le programme moins lisible
- ☐ de structurer le programme
- ☐ de rendre le programme plus complexe
- ☐ de localiser les objets
- ☐ de paramétrer les actions
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ variables
- ☐ effectifs
- ☐ formels
- ☐ effaçables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ jamais vraie
- ☐ vraie uniquement à la dernière itération
- ☐ vraie uniquement à la première itération
- ☐ vraie à chaque itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ la probabilité que a soit égal à b
- ☐ le minimum de a et b
- ☐ le produit de a par b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N
- ☐ 0 et N-1
- ☐ -1 et N+1
- ☐ 1 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 1 seul énoncé itératif **for**
- ☐ 3 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ toujours vérifié
- ☐ jamais vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ parfois vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016**Durée 1h30**

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ C
☐ Lisp
☐ Java
☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ on ne peut pas
☐ en l'exécutant sur l'ordinateur
☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
☐ avec le compilateur

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une constante
☐ une personne
☐ une variable
☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques

Question 5 ♣ En C, un entier peut être de type :

- ☐ int
☐ long
☐ double
☐ très long
☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ 100
☐ -1
☐ -34
☐ -35



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ pair
- ☐ impair
- ☐ positif
- ☐ négatif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 126 et 767
- ☐ -3 et 768
- ☐ 125 et 765
- ☐ aucune des deux

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont toujours exacts
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$
- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de rendre le programme moins lisible
- ☐ de localiser les objets
- ☐ de rendre le programme plus complexe
- ☐ de structurer le programme
- ☐ de paramétrer les actions
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé E sont des affirmations qui doivent :

- ☐ être toujours vraies avant l'exécution de E
- ☐ être toujours vraies, respectivement, avant et après l'exécution de E
- ☐ être toujours fausses avant et après l'exécution de E
- ☐ de temps en temps vraies avant et après l'exécution de E

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effectifs
- ☐ effaçables
- ☐ formels
- ☐ variables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie uniquement à la première itération
- ☐ jamais vraie
- ☐ vraie à chaque itération
- ☐ vraie uniquement à la dernière itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le produit de a par b
- ☐ la probabilité que a soit égal à b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le minimum de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N-1
- ☐ 1 et N
- ☐ 0 et N
- ☐ -1 et N+1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 3 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (i.e. `t[i]`) est :

- ☐ est vérifié si l'ordinateur est en forme
- ☐ jamais vérifié
- ☐ parfois vérifié
- ☐ toujours vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ elle ne fait rien et ne renvoie rien
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016**Durée 1h30**

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ C
☐ Java
☐ Lisp
☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
☐ en l'exécutant sur l'ordinateur
☐ on ne peut pas
☐ avec le compilateur

Question 4 En informatique, est-ce qu'un *type* ?

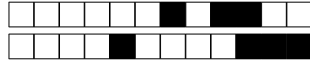
- ☐ une personne
☐ une variable
☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
☐ une constante

Question 5 ♣ En C, un entier peut être de type :

- ☐ très long
☐ int
☐ double
☐ long
☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -35
☐ -34
☐ 100
☐ -1



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ positif
- ☐ négatif
- ☐ impair
- ☐ pair

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ aucune des deux
- ☐ -3 et 768
- ☐ 126 et 767
- ☐ 125 et 765

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont toujours exacts
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « formel donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et $(x \neq 1 \text{ ou } y \neq 0)$
- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de rendre le programme moins lisible
- ☐ de structurer le programme
- ☐ de rendre le programme plus complexe
- ☐ de paramétrer les actions
- ☐ de localiser les objets
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé E sont des affirmations qui doivent :

- ☐ être toujours vraies avant l'exécution de E
- ☐ être toujours vraies, respectivement, avant et après l'exécution de E
- ☐ de temps en temps vraies avant et après l'exécution de E
- ☐ être toujours fausses avant et après l'exécution de E

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effectifs
- ☐ effaçables
- ☐ variables
- ☐ formels

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie uniquement à la première itération
- ☐ vraie à chaque itération
- ☐ jamais vraie
- ☐ vraie uniquement à la dernière itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ la probabilité que a soit égal à b
- ☐ le minimum de a et b
- ☐ le produit de a par b
- ☐ le pgcd (plus grand commun diviseur) de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ -1 et N+1
- ☐ 1 et N
- ☐ 0 et N
- ☐ 0 et N-1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés **if-then-else**
- ☐ 2 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**
- ☐ 3 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (i.e. `t[i]`) est :

- ☐ toujours vérifié
- ☐ jamais vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ parfois vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
- ☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ C
- ☐ Java
- ☐ Lisp
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ en l'exécutant sur l'ordinateur
- ☐ on ne peut pas
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ avec le compilateur

Question 4 En informatique, est-ce qu'un *type* ?

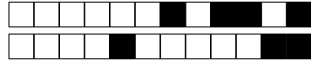
- ☐ une personne
- ☐ une variable
- ☐ une constante
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques

Question 5 ♣ En C, un entier peut être de type :

- ☐ long
- ☐ int
- ☐ double
- ☐ très long
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ 100
- ☐ -35
- ☐ -34
- ☐ -1



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ négatif
- ☐ impair
- ☐ positif
- ☐ pair

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ aucune des deux
- ☐ 125 et 765
- ☐ 126 et 767
- ☐ -3 et 768

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont toujours exacts
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif donnée » et « résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de rendre le programme plus complexe
- ☐ de localiser les objets
- ☐ de rendre le programme moins lisible
- ☐ de paramétrer les actions
- ☐ de structurer le programme
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ formels
- ☐ variables
- ☐ effectifs
- ☐ effaçables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie uniquement à la dernière itération
- ☐ vraie uniquement à la première itération
- ☐ jamais vraie
- ☐ vraie à chaque itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le produit de a par b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ la probabilité que a soit égal à b
- ☐ le minimum de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ -1 et N+1
- ☐ 1 et N
- ☐ 0 et N-1
- ☐ 0 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 3 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 2 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ toujours vérifié
- ☐ jamais vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ parfois vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
- ☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Java
- ☐ C
- ☐ Lisp
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ avec le compilateur
- ☐ en l'exécutant sur l'ordinateur
- ☐ on ne peut pas

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une variable
- ☐ une constante
- ☐ une personne
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques

Question 5 ♣ En C, un entier peut être de type :

- ☐ double
- ☐ très long
- ☐ long
- ☐ int
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -34
- ☐ -1
- ☐ 100
- ☐ -35



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ pair
- ☐ positif
- ☐ impair
- ☐ négatif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 125 et 765
- ☐ 126 et 767
- ☐ -3 et 768
- ☐ aucune des deux

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont toujours exacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « formel donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x = 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de paramétrer les actions
- ☐ de structurer le programme
- ☐ de localiser les objets
- ☐ de rendre le programme moins lisible
- ☐ de rendre le programme plus complexe
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ variables
- ☐ formels
- ☐ effectifs
- ☐ effaçables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie uniquement à la dernière itération
- ☐ vraie à chaque itération
- ☐ vraie uniquement à la première itération
- ☐ jamais vraie

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le minimum de a et b
- ☐ la probabilité que a soit égal à b
- ☐ le produit de a par b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 1 et N
- ☐ 0 et N
- ☐ 0 et N-1
- ☐ -1 et N+1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés **if-then-else**
- ☐ 1 seul énoncé itératif **for**
- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ parfois vérifié
- ☐ jamais vérifié
- ☐ toujours vérifié
- ☐ est vérifié si l'ordinateur est en forme

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016**Durée 1h30**

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Lisp
☐ Java
☐ C
☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ en l'exécutant sur l'ordinateur
☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
☐ avec le compilateur
☐ on ne peut pas

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
☐ une variable
☐ une constante
☐ une personne

Question 5 ♣ En C, un entier peut être de type :

- ☐ double
☐ très long
☐ long
☐ int
☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -1
☐ -34
☐ 100
☐ -35



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ positif
- ☐ pair
- ☐ négatif
- ☐ impair

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ aucune des deux
- ☐ 125 et 765
- ☐ 126 et 767
- ☐ -3 et 768

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont toujours exacts
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et $(x \neq 1 \text{ ou } y \neq 0)$
- ☐ $x \neq 0$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de paramétrer les actions
- ☐ de localiser les objets
- ☐ de rendre le programme plus complexe
- ☐ de rendre le programme moins lisible
- ☐ de structurer le programme
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours vraies avant l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ formels
- ☐ effaçables
- ☐ effectifs
- ☐ variables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie uniquement à la dernière itération
- ☐ vraie à chaque itération
- ☐ vraie uniquement à la première itération
- ☐ jamais vraie

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le produit de a par b
- ☐ le minimum de a et b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ la probabilité que a soit égal à b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ -1 et N+1
- ☐ 0 et N-1
- ☐ 1 et N
- ☐ 0 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**
- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ parfois vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ toujours vérifié
- ☐ jamais vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016**Durée 1h30**

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ C
☐ Lisp
☐ Java
☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ en l'exécutant sur l'ordinateur
☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
☐ on ne peut pas
☐ avec le compilateur

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une constante
☐ une variable
☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
☐ une personne

Question 5 ♣ En C, un entier peut être de type :

- ☐ très long
☐ int
☐ double
☐ long
☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ 100
☐ -1
☐ -34
☐ -35



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ positif
- ☐ impair
- ☐ pair
- ☐ négatif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 126 et 767
- ☐ aucune des deux
- ☐ 125 et 765
- ☐ -3 et 768

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont toujours exacts
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x \neq 0$
- ☐ $x = 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de paramétrer les actions
- ☐ de structurer le programme
- ☐ de localiser les objets
- ☐ de rendre le programme moins lisible
- ☐ de rendre le programme plus complexe
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effaçables
- ☐ effectifs
- ☐ variables
- ☐ formels

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ jamais vraie
- ☐ vraie uniquement à la première itération
- ☐ vraie uniquement à la dernière itération
- ☐ vraie à chaque itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le produit de a par b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le minimum de a et b
- ☐ la probabilité que a soit égal à b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 1 et N
- ☐ 0 et N-1
- ☐ -1 et N+1
- ☐ 0 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés itératifs **for**
- ☐ 3 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés **if-then-else**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (i.e. `t[i]`) est :

- ☐ toujours vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ parfois vérifié
- ☐ jamais vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016**Durée 1h30**

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Lisp
☐ C
☐ Java
☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ avec le compilateur
☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
☐ on ne peut pas
☐ en l'exécutant sur l'ordinateur

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une constante
☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
☐ une personne
☐ une variable

Question 5 ♣ En C, un entier peut être de type :

- ☐ très long
☐ int
☐ long
☐ double
☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -34
☐ 100
☐ -1
☐ -35



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ pair
- ☐ négatif
- ☐ impair
- ☐ positif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ aucune des deux
- ☐ -3 et 768
- ☐ 126 et 767
- ☐ 125 et 765

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ les calculs avec les réels sont toujours exacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif donnée » et « résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de rendre le programme plus complexe
- ☐ de structurer le programme
- ☐ de paramétrer les actions
- ☐ de rendre le programme moins lisible
- ☐ de localiser les objets
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé E sont des affirmations qui doivent :

- ☐ être toujours vraies avant l'exécution de E
- ☐ de temps en temps vraies avant et après l'exécution de E
- ☐ être toujours vraies, respectivement, avant et après l'exécution de E
- ☐ être toujours fausses avant et après l'exécution de E

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ variables
- ☐ effectifs
- ☐ formels
- ☐ effaçables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie à chaque itération
- ☐ vraie uniquement à la première itération
- ☐ vraie uniquement à la dernière itération
- ☐ jamais vraie

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le minimum de a et b
- ☐ le produit de a par b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ la probabilité que a soit égal à b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 1 et N
- ☐ 0 et N
- ☐ 0 et N-1
- ☐ -1 et N+1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés **if-then-else**
- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ toujours vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ parfois vérifié
- ☐ jamais vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ elle ne fait rien et ne renvoie rien
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
- ☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ C
- ☐ Java
- ☐ Lisp
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ on ne peut pas
- ☐ en l'exécutant sur l'ordinateur
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ avec le compilateur

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une constante
- ☐ une variable
- ☐ une personne
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques

Question 5 ♣ En C, un entier peut être de type :

- ☐ très long
- ☐ double
- ☐ int
- ☐ long
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -1
- ☐ -35
- ☐ -34
- ☐ 100



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ impair
- ☐ pair
- ☐ négatif
- ☐ positif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ aucune des deux
- ☐ 126 et 767
- ☐ -3 et 768
- ☐ 125 et 765

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ les calculs avec les réels sont toujours exacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et $(x \neq 1 \text{ ou } y \neq 0)$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ $x = 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de structurer le programme
- ☐ de localiser les objets
- ☐ de rendre le programme plus complexe
- ☐ de rendre le programme moins lisible
- ☐ de paramétrer les actions
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ variables
- ☐ effaçables
- ☐ effectifs
- ☐ formels

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie uniquement à la dernière itération
- ☐ jamais vraie
- ☐ vraie à chaque itération
- ☐ vraie uniquement à la première itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ la probabilité que a soit égal à b
- ☐ le produit de a par b
- ☐ le minimum de a et b
- ☐ le pgcd (plus grand commun diviseur) de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 1 et N
- ☐ 0 et N-1
- ☐ -1 et N+1
- ☐ 0 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 1 seul énoncé itératif **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ toujours vérifié
- ☐ jamais vérifié
- ☐ parfois vérifié
- ☐ est vérifié si l'ordinateur est en forme

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016**Durée 1h30**

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Java
☐ C
☐ Lisp
☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ on ne peut pas
☐ en l'exécutant sur l'ordinateur
☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
☐ avec le compilateur

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une variable
☐ une constante
☐ une personne
☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques

Question 5 ♣ En C, un entier peut être de type :

- ☐ long
☐ très long
☐ double
☐ int
☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -34
☐ -1
☐ -35
☐ 100



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ impair
- ☐ positif
- ☐ pair
- ☐ négatif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 125 et 765
- ☐ -3 et 768
- ☐ aucune des deux
- ☐ 126 et 767

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont toujours exacts
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ $x = 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de structurer le programme
- ☐ de paramétrer les actions
- ☐ de localiser les objets
- ☐ de rendre le programme plus complexe
- ☐ de rendre le programme moins lisible
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effaçables
- ☐ formels
- ☐ effectifs
- ☐ variables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie uniquement à la dernière itération
- ☐ jamais vraie
- ☐ vraie à chaque itération
- ☐ vraie uniquement à la première itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le minimum de a et b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ la probabilité que a soit égal à b
- ☐ le produit de a par b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N
- ☐ 0 et N-1
- ☐ -1 et N+1
- ☐ 1 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 1 seul énoncé itératif **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ toujours vérifié
- ☐ parfois vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ jamais vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016**Durée 1h30**

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ C
☐ Java
☐ Lisp
☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ on ne peut pas
☐ avec le compilateur
☐ en l'exécutant sur l'ordinateur
☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une variable
☐ une constante
☐ une personne
☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques

Question 5 ♣ En C, un entier peut être de type :

- ☐ très long
☐ long
☐ int
☐ double
☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -1
☐ -35
☐ -34
☐ 100



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ négatif
- ☐ impair
- ☐ pair
- ☐ positif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ aucune des deux
- ☐ -3 et 768
- ☐ 126 et 767
- ☐ 125 et 765

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont toujours exacts
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée » et « résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$
- ☐ $x = 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x \neq 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de structurer le programme
- ☐ de localiser les objets
- ☐ de rendre le programme moins lisible
- ☐ de rendre le programme plus complexe
- ☐ de paramétrer les actions
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effaçables
- ☐ variables
- ☐ formels
- ☐ effectifs

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie à chaque itération
- ☐ vraie uniquement à la dernière itération
- ☐ jamais vraie
- ☐ vraie uniquement à la première itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le produit de a par b
- ☐ le minimum de a et b
- ☐ la probabilité que a soit égal à b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N-1
- ☐ -1 et N+1
- ☐ 0 et N
- ☐ 1 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 3 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ est vérifié si l'ordinateur est en forme
- ☐ toujours vérifié
- ☐ parfois vérifié
- ☐ jamais vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ Aucune de ces réponses n'est correcte.

**Examen du 28/10/2016****Durée 1h30**

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ C
☐ Java
☐ Lisp
☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ avec le compilateur
☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
☐ en l'exécutant sur l'ordinateur
☐ on ne peut pas

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
☐ une personne
☐ une constante
☐ une variable

Question 5 ♣ En C, un entier peut être de type :

- ☐ double
☐ int
☐ très long
☐ long
☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -34
☐ -1
☐ -35
☐ 100



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ négatif
- ☐ pair
- ☐ impair
- ☐ positif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ aucune des deux
- ☐ 126 et 767
- ☐ 125 et 765
- ☐ -3 et 768

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont toujours exacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de structurer le programme
- ☐ de localiser les objets
- ☐ de paramétrer les actions
- ☐ de rendre le programme plus complexe
- ☐ de rendre le programme moins lisible
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effaçables
- ☐ formels
- ☐ variables
- ☐ effectifs

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ jamais vraie
- ☐ vraie uniquement à la première itération
- ☐ vraie uniquement à la dernière itération
- ☐ vraie à chaque itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le produit de a par b
- ☐ la probabilité que a soit égal à b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le minimum de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N-1
- ☐ 0 et N
- ☐ -1 et N+1
- ☐ 1 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés itératifs **for**
- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ parfois vérifié
- ☐ jamais vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ toujours vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si `x` $\in$ `t` et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016**Durée 1h30**

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Java
☐ Lisp
☐ C
☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ avec le compilateur
☐ on ne peut pas
☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
☐ en l'exécutant sur l'ordinateur

Question 4 En informatique, est-ce qu'un *type* ?

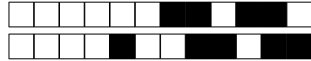
- ☐ une variable
☐ une personne
☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
☐ une constante

Question 5 ♣ En C, un entier peut être de type :

- ☐ long
☐ int
☐ double
☐ très long
☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -34
☐ -1
☐ -35
☐ 100



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ impair
- ☐ positif
- ☐ pair
- ☐ négatif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 126 et 767
- ☐ aucune des deux
- ☐ -3 et 768
- ☐ 125 et 765

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont toujours exacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ $x \neq 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de structurer le programme
- ☐ de localiser les objets
- ☐ de rendre le programme plus complexe
- ☐ de rendre le programme moins lisible
- ☐ de paramétrer les actions
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effaçables
- ☐ variables
- ☐ formels
- ☐ effectifs

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie uniquement à la dernière itération
- ☐ vraie à chaque itération
- ☐ jamais vraie
- ☐ vraie uniquement à la première itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ la probabilité que a soit égal à b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le produit de a par b
- ☐ le minimum de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N
- ☐ 1 et N
- ☐ -1 et N+1
- ☐ 0 et N-1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 2 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (i.e. `t[i]`) est :

- ☐ jamais vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ parfois vérifié
- ☐ toujours vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016**Durée 1h30**

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Lisp
☐ C
☐ Java
☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ on ne peut pas
☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
☐ en l'exécutant sur l'ordinateur
☐ avec le compilateur

Question 4 En informatique, est-ce qu'un *type* ?

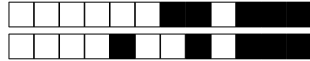
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
☐ une variable
☐ une constante
☐ une personne

Question 5 ♣ En C, un entier peut être de type :

- ☐ int
☐ long
☐ double
☐ très long
☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ 100
☐ -1
☐ -34
☐ -35



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ positif
- ☐ négatif
- ☐ impair
- ☐ pair

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 125 et 765
- ☐ aucune des deux
- ☐ -3 et 768
- ☐ 126 et 767

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont toujours exacts
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée » et « résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et $(x \neq 1 \text{ ou } y \neq 0)$
- ☐ $x \neq 0$
- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de localiser les objets
- ☐ de paramétrer les actions
- ☐ de rendre le programme plus complexe
- ☐ de structurer le programme
- ☐ de rendre le programme moins lisible
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé E sont des affirmations qui doivent :

- ☐ être toujours fausses avant et après l'exécution de E
- ☐ de temps en temps vraies avant et après l'exécution de E
- ☐ être toujours vraies, respectivement, avant et après l'exécution de E
- ☐ être toujours vraies avant l'exécution de E

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ formels
- ☐ variables
- ☐ effaçables
- ☐ effectifs

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie à chaque itération
- ☐ jamais vraie
- ☐ vraie uniquement à la dernière itération
- ☐ vraie uniquement à la première itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le produit de a par b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le minimum de a et b
- ☐ la probabilité que a soit égal à b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N-1
- ☐ 1 et N
- ☐ 0 et N
- ☐ -1 et N+1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 3 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ parfois vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ jamais vérifié
- ☐ toujours vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ elle ne fait rien et ne renvoie rien
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016**Durée 1h30**

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Java
☐ Lisp
☐ C
☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
☐ on ne peut pas
☐ avec le compilateur
☐ en l'exécutant sur l'ordinateur

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
☐ une variable
☐ une personne
☐ une constante

Question 5 ♣ En C, un entier peut être de type :

- ☐ int
☐ long
☐ double
☐ très long
☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -1
☐ 100
☐ -35
☐ -34



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ négatif
- ☐ pair
- ☐ impair
- ☐ positif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 125 et 765
- ☐ aucune des deux
- ☐ -3 et 768
- ☐ 126 et 767

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont toujours exacts
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x \neq 0$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de structurer le programme
- ☐ de rendre le programme moins lisible
- ☐ de rendre le programme plus complexe
- ☐ de localiser les objets
- ☐ de paramétrer les actions
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ formels
- ☐ variables
- ☐ effectifs
- ☐ effaçables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ jamais vraie
- ☐ vraie uniquement à la première itération
- ☐ vraie uniquement à la dernière itération
- ☐ vraie à chaque itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le produit de a par b
- ☐ le minimum de a et b
- ☐ la probabilité que a soit égal à b
- ☐ le pgcd (plus grand commun diviseur) de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ -1 et N+1
- ☐ 1 et N
- ☐ 0 et N
- ☐ 0 et N-1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés **if-then-else**
- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés itératifs **for**
- ☐ 3 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ est vérifié si l'ordinateur est en forme
- ☐ jamais vérifié
- ☐ parfois vérifié
- ☐ toujours vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016**Durée 1h30**

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Java
☐ Lisp
☐ C
☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
☐ avec le compilateur
☐ en l'exécutant sur l'ordinateur
☐ on ne peut pas

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une constante
☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
☐ une variable
☐ une personne

Question 5 ♣ En C, un entier peut être de type :

- ☐ double
☐ long
☐ très long
☐ int
☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -34
☐ 100
☐ -1
☐ -35



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ négatif
- ☐ pair
- ☐ impair
- ☐ positif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ -3 et 768
- ☐ 126 et 767
- ☐ aucune des deux
- ☐ 125 et 765

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ les calculs avec les réels sont toujours exacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée » et « résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x \neq 0$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de structurer le programme
- ☐ de rendre le programme plus complexe
- ☐ de rendre le programme moins lisible
- ☐ de paramétrer les actions
- ☐ de localiser les objets
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effectifs
- ☐ effaçables
- ☐ formels
- ☐ variables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie à chaque itération
- ☐ jamais vraie
- ☐ vraie uniquement à la première itération
- ☐ vraie uniquement à la dernière itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le minimum de a et b
- ☐ le produit de a par b
- ☐ la probabilité que a soit égal à b
- ☐ le pgcd (plus grand commun diviseur) de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N-1
- ☐ 1 et N
- ☐ -1 et N+1
- ☐ 0 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 3 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ toujours vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ jamais vérifié
- ☐ parfois vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016**Durée 1h30**

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Lisp
☐ C
☐ Java
☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ avec le compilateur
☐ on ne peut pas
☐ en l'exécutant sur l'ordinateur
☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une variable
☐ une personne
☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
☐ une constante

Question 5 ♣ En C, un entier peut être de type :

- ☐ int
☐ long
☐ double
☐ très long
☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ 100
☐ -35
☐ -34
☐ -1



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ positif
- ☐ pair
- ☐ impair
- ☐ négatif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 125 et 765
- ☐ aucune des deux
- ☐ -3 et 768
- ☐ 126 et 767

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ les calculs avec les réels sont toujours exacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et $(x \neq 1 \text{ ou } y \neq 0)$
- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de rendre le programme plus complexe
- ☐ de paramétrer les actions
- ☐ de rendre le programme moins lisible
- ☐ de structurer le programme
- ☐ de localiser les objets
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ variables
- ☐ effaçables
- ☐ effectifs
- ☐ formels

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie uniquement à la première itération
- ☐ vraie uniquement à la dernière itération
- ☐ vraie à chaque itération
- ☐ jamais vraie

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ la probabilité que a soit égal à b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le minimum de a et b
- ☐ le produit de a par b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N-1
- ☐ -1 et N+1
- ☐ 0 et N
- ☐ 1 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 3 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ parfois vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ jamais vérifié
- ☐ toujours vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016**Durée 1h30**

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Java
☐ C
☐ Lisp
☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ avec le compilateur
☐ en l'exécutant sur l'ordinateur
☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
☐ on ne peut pas

Question 4 En informatique, est-ce qu'un *type* ?

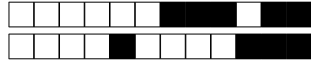
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
☐ une personne
☐ une constante
☐ une variable

Question 5 ♣ En C, un entier peut être de type :

- ☐ int
☐ double
☐ long
☐ très long
☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -35
☐ -34
☐ 100
☐ -1



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ négatif
- ☐ impair
- ☐ pair
- ☐ positif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 125 et 765
- ☐ aucune des deux
- ☐ -3 et 768
- ☐ 126 et 767

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont toujours exacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x \neq 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de rendre le programme moins lisible
- ☐ de paramétrer les actions
- ☐ de structurer le programme
- ☐ de localiser les objets
- ☐ de rendre le programme plus complexe
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effectifs
- ☐ effaçables
- ☐ formels
- ☐ variables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ jamais vraie
- ☐ vraie à chaque itération
- ☐ vraie uniquement à la première itération
- ☐ vraie uniquement à la dernière itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le produit de a par b
- ☐ la probabilité que a soit égal à b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le minimum de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ -1 et N+1
- ☐ 0 et N
- ☐ 1 et N
- ☐ 0 et N-1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés **if-then-else**
- ☐ 2 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**
- ☐ 3 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ parfois vérifié
- ☐ toujours vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ jamais vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
- ☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ C
- ☐ Java
- ☐ Lisp
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ en l'exécutant sur l'ordinateur
- ☐ avec le compilateur
- ☐ on ne peut pas
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents

Question 4 En informatique, est-ce qu'un *type* ?

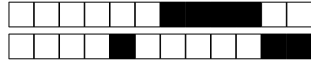
- ☐ une variable
- ☐ une constante
- ☐ une personne
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques

Question 5 ♣ En C, un entier peut être de type :

- ☐ très long
- ☐ double
- ☐ int
- ☐ long
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ 100
- ☐ -35
- ☐ -34
- ☐ -1



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ négatif
- ☐ positif
- ☐ pair
- ☐ impair

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 125 et 765
- ☐ -3 et 768
- ☐ 126 et 767
- ☐ aucune des deux

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont toujours exacts
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|a - b| < \epsilon$
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée » et « résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x = 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de paramétrer les actions
- ☐ de rendre le programme moins lisible
- ☐ de structurer le programme
- ☐ de rendre le programme plus complexe
- ☐ de localiser les objets
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours vraies avant l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effectifs
- ☐ formels
- ☐ effaçables
- ☐ variables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie uniquement à la dernière itération
- ☐ vraie à chaque itération
- ☐ jamais vraie
- ☐ vraie uniquement à la première itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ la probabilité que a soit égal à b
- ☐ le minimum de a et b
- ☐ le produit de a par b
- ☐ le pgcd (plus grand commun diviseur) de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ -1 et N+1
- ☐ 0 et N
- ☐ 1 et N
- ☐ 0 et N-1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés **if-then-else**
- ☐ 2 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**
- ☐ 3 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ parfois vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ jamais vérifié
- ☐ toujours vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
- ☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ C
- ☐ Java
- ☐ Lisp
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ on ne peut pas
- ☐ en l'exécutant sur l'ordinateur
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ avec le compilateur

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une variable
- ☐ une personne
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une constante

Question 5 ♣ En C, un entier peut être de type :

- ☐ double
- ☐ long
- ☐ très long
- ☐ int
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -34
- ☐ -1
- ☐ 100
- ☐ -35



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ positif
- ☐ pair
- ☐ impair
- ☐ négatif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 125 et 765
- ☐ -3 et 768
- ☐ 126 et 767
- ☐ aucune des deux

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ les calculs avec les réels sont toujours exacts
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$
- ☐ $x = 0$ et $(x \neq 1$ ou $y \neq 0)$
- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de rendre le programme plus complexe
- ☐ de structurer le programme
- ☐ de rendre le programme moins lisible
- ☐ de paramétrer les actions
- ☐ de localiser les objets
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ formels
- ☐ effectifs
- ☐ effaçables
- ☐ variables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie à chaque itération
- ☐ vraie uniquement à la dernière itération
- ☐ jamais vraie
- ☐ vraie uniquement à la première itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ la probabilité que a soit égal à b
- ☐ le minimum de a et b
- ☐ le produit de a par b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N-1
- ☐ 0 et N
- ☐ -1 et N+1
- ☐ 1 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés **if-then-else**
- ☐ 3 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ jamais vérifié
- ☐ toujours vérifié
- ☐ parfois vérifié
- ☐ est vérifié si l'ordinateur est en forme

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
- ☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Java
- ☐ C
- ☐ Lisp
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ avec le compilateur
- ☐ en l'exécutant sur l'ordinateur
- ☐ on ne peut pas
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une variable
- ☐ une constante
- ☐ une personne

Question 5 ♣ En C, un entier peut être de type :

- ☐ int
- ☐ double
- ☐ long
- ☐ très long
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -1
- ☐ -35
- ☐ 100
- ☐ -34



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ positif
- ☐ négatif
- ☐ pair
- ☐ impair

Question 8 À quelle valeur décimale correspond les valeurs 0176 et 0x2FF ?

- ☐ 125 et 765
- ☐ 126 et 767
- ☐ aucune des deux
- ☐ -3 et 768

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ les calculs avec les réels sont toujours exacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x \neq 0$
- ☐ $x = 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de rendre le programme plus complexe
- ☐ de localiser les objets
- ☐ de structurer le programme
- ☐ de paramétrer les actions
- ☐ de rendre le programme moins lisible
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé E sont des affirmations qui doivent :

- ☐ être toujours vraies avant l'exécution de E
- ☐ de temps en temps vraies avant et après l'exécution de E
- ☐ être toujours vraies, respectivement, avant et après l'exécution de E
- ☐ être toujours fausses avant et après l'exécution de E

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ variables
- ☐ effectifs
- ☐ effaçables
- ☐ formels

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie à chaque itération
- ☐ jamais vraie
- ☐ vraie uniquement à la première itération
- ☐ vraie uniquement à la dernière itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le produit de a par b
- ☐ la probabilité que a soit égal à b
- ☐ le minimum de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N
- ☐ 0 et N-1
- ☐ -1 et N+1
- ☐ 1 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 1 seul énoncé itératif **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ toujours vérifié
- ☐ jamais vérifié
- ☐ parfois vérifié
- ☐ est vérifié si l'ordinateur est en forme

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016**Durée 1h30**

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Lisp
☐ Java
☐ C
☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ avec le compilateur
☐ on ne peut pas
☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
☐ en l'exécutant sur l'ordinateur

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
☐ une constante
☐ une variable
☐ une personne

Question 5 ♣ En C, un entier peut être de type :

- ☐ int
☐ très long
☐ double
☐ long
☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -1
☐ -34
☐ 100
☐ -35



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ pair
- ☐ positif
- ☐ négatif
- ☐ impair

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ aucune des deux
- ☐ 126 et 767
- ☐ -3 et 768
- ☐ 125 et 765

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont toujours exacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et $(x \neq 1 \text{ ou } y \neq 0)$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ $x = 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de rendre le programme moins lisible
- ☐ de paramétrer les actions
- ☐ de localiser les objets
- ☐ de rendre le programme plus complexe
- ☐ de structurer le programme
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ formels
- ☐ variables
- ☐ effaçables
- ☐ effectifs

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie uniquement à la dernière itération
- ☐ vraie à chaque itération
- ☐ jamais vraie
- ☐ vraie uniquement à la première itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ la probabilité que a soit égal à b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le minimum de a et b
- ☐ le produit de a par b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N
- ☐ 0 et N-1
- ☐ 1 et N
- ☐ -1 et N+1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**
- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ est vérifié si l'ordinateur est en forme
- ☐ jamais vérifié
- ☐ parfois vérifié
- ☐ toujours vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016**Durée 1h30**

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Java
☐ Lisp
☐ C
☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ on ne peut pas
☐ en l'exécutant sur l'ordinateur
☐ avec le compilateur
☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une variable
☐ une constante
☐ une personne
☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques

Question 5 ♣ En C, un entier peut être de type :

- ☐ double
☐ long
☐ int
☐ très long
☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ 100
☐ -34
☐ -35
☐ -1



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ négatif
- ☐ impair
- ☐ pair
- ☐ positif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ -3 et 768
- ☐ 126 et 767
- ☐ 125 et 765
- ☐ aucune des deux

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont toujours exacts
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif donnée » et « résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de paramétrer les actions
- ☐ de rendre le programme moins lisible
- ☐ de structurer le programme
- ☐ de rendre le programme plus complexe
- ☐ de localiser les objets
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effectifs
- ☐ variables
- ☐ effaçables
- ☐ formels

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie uniquement à la dernière itération
- ☐ vraie uniquement à la première itération
- ☐ vraie à chaque itération
- ☐ jamais vraie

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le produit de a par b
- ☐ le minimum de a et b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ la probabilité que a soit égal à b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N
- ☐ 1 et N
- ☐ -1 et N+1
- ☐ 0 et N-1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 1 seul énoncé itératif **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ toujours vérifié
- ☐ parfois vérifié
- ☐ jamais vérifié
- ☐ est vérifié si l'ordinateur est en forme

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
- ☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ C
- ☐ Java
- ☐ Lisp
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ avec le compilateur
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ on ne peut pas
- ☐ en l'exécutant sur l'ordinateur

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une constante
- ☐ une personne
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une variable

Question 5 ♣ En C, un entier peut être de type :

- ☐ int
- ☐ double
- ☐ long
- ☐ très long
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -34
- ☐ -1
- ☐ -35
- ☐ 100



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ positif
- ☐ négatif
- ☐ pair
- ☐ impair

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ aucune des deux
- ☐ 126 et 767
- ☐ 125 et 765
- ☐ -3 et 768

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont toujours exacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et $(x \neq 1 \text{ ou } y \neq 0)$
- ☐ $x \neq 0$
- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de structurer le programme
- ☐ de rendre le programme moins lisible
- ☐ de localiser les objets
- ☐ de paramétrer les actions
- ☐ de rendre le programme plus complexe
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ variables
- ☐ effectifs
- ☐ effaçables
- ☐ formels

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie uniquement à la première itération
- ☐ vraie à chaque itération
- ☐ jamais vraie
- ☐ vraie uniquement à la dernière itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le minimum de a et b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le produit de a par b
- ☐ la probabilité que a soit égal à b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N-1
- ☐ 1 et N
- ☐ 0 et N
- ☐ -1 et N+1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 1 seul énoncé itératif **for**
- ☐ 3 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ est vérifié si l'ordinateur est en forme
- ☐ jamais vérifié
- ☐ toujours vérifié
- ☐ parfois vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016**Durée 1h30**

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Java
☐ Lisp
☐ C
☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ en l'exécutant sur l'ordinateur
☐ avec le compilateur
☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
☐ on ne peut pas

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une variable
☐ une personne
☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
☐ une constante

Question 5 ♣ En C, un entier peut être de type :

- ☐ très long
☐ double
☐ long
☐ int
☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -1
☐ 100
☐ -35
☐ -34



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ pair
- ☐ positif
- ☐ impair
- ☐ négatif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 126 et 767
- ☐ aucune des deux
- ☐ -3 et 768
- ☐ 125 et 765

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont toujours exacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « formel donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de rendre le programme plus complexe
- ☐ de paramétrer les actions
- ☐ de localiser les objets
- ☐ de rendre le programme moins lisible
- ☐ de structurer le programme
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ variables
- ☐ effaçables
- ☐ formels
- ☐ effectifs

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie uniquement à la première itération
- ☐ jamais vraie
- ☐ vraie à chaque itération
- ☐ vraie uniquement à la dernière itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ la probabilité que a soit égal à b
- ☐ le produit de a par b
- ☐ le minimum de a et b
- ☐ le pgcd (plus grand commun diviseur) de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 1 et N
- ☐ 0 et N
- ☐ -1 et N+1
- ☐ 0 et N-1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés **if-then-else**
- ☐ 1 seul énoncé itératif **for**
- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ est vérifié si l'ordinateur est en forme
- ☐ parfois vérifié
- ☐ jamais vérifié
- ☐ toujours vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ elle renvoie 1 si `x` $\in$ `t` et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
- ☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Lisp
- ☐ C
- ☐ Java
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ en l'exécutant sur l'ordinateur
- ☐ on ne peut pas
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ avec le compilateur

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une variable
- ☐ une personne
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une constante

Question 5 ♣ En C, un entier peut être de type :

- ☐ double
- ☐ très long
- ☐ long
- ☐ int
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -1
- ☐ 100
- ☐ -35
- ☐ -34



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ pair
- ☐ positif
- ☐ impair
- ☐ négatif

Question 8 À quelle valeur décimale correspond les valeurs 0176 et 0x2FF ?

- ☐ 126 et 767
- ☐ -3 et 768
- ☐ 125 et 765
- ☐ aucune des deux

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont toujours exacts
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif donnée » et « résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de rendre le programme plus complexe
- ☐ de rendre le programme moins lisible
- ☐ de localiser les objets
- ☐ de structurer le programme
- ☐ de paramétrer les actions
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effaçables
- ☐ variables
- ☐ formels
- ☐ effectifs

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ jamais vraie
- ☐ vraie uniquement à la dernière itération
- ☐ vraie à chaque itération
- ☐ vraie uniquement à la première itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ la probabilité que a soit égal à b
- ☐ le minimum de a et b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le produit de a par b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ -1 et N+1
- ☐ 0 et N
- ☐ 1 et N
- ☐ 0 et N-1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés **if-then-else**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ parfois vérifié
- ☐ jamais vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ toujours vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
- ☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ C
- ☐ Java
- ☐ Lisp
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ en l'exécutant sur l'ordinateur
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ avec le compilateur
- ☐ on ne peut pas

Question 4 En informatique, est-ce qu'un *type* ?

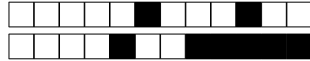
- ☐ une constante
- ☐ une variable
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une personne

Question 5 ♣ En C, un entier peut être de type :

- ☐ int
- ☐ double
- ☐ très long
- ☐ long
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ 100
- ☐ -1
- ☐ -35
- ☐ -34



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ pair
- ☐ négatif
- ☐ impair
- ☐ positif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 126 et 767
- ☐ 125 et 765
- ☐ aucune des deux
- ☐ -3 et 768

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont toujours exacts
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée » et « résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$
- ☐ $x = 0$ et $(x \neq 1 \text{ ou } y \neq 0)$
- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de structurer le programme
- ☐ de rendre le programme moins lisible
- ☐ de localiser les objets
- ☐ de rendre le programme plus complexe
- ☐ de paramétrer les actions
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ formels
- ☐ effectifs
- ☐ variables
- ☐ effaçables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

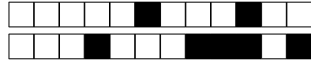
- ☐ vraie uniquement à la dernière itération
- ☐ vraie à chaque itération
- ☐ vraie uniquement à la première itération
- ☐ jamais vraie

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le produit de a par b
- ☐ la probabilité que a soit égal à b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le minimum de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N-1
- ☐ 1 et N
- ☐ 0 et N
- ☐ -1 et N+1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 3 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

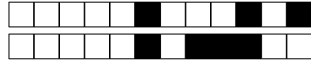
- ☐ toujours vérifié
- ☐ jamais vérifié
- ☐ parfois vérifié
- ☐ est vérifié si l'ordinateur est en forme

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
- ☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ C
- ☐ Lisp
- ☐ Java
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ avec le compilateur
- ☐ en l'exécutant sur l'ordinateur
- ☐ on ne peut pas

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une variable
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une personne
- ☐ une constante

Question 5 ♣ En C, un entier peut être de type :

- ☐ int
- ☐ très long
- ☐ long
- ☐ double
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -34
- ☐ -1
- ☐ 100
- ☐ -35



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ pair
- ☐ positif
- ☐ négatif
- ☐ impair

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ -3 et 768
- ☐ aucune des deux
- ☐ 125 et 765
- ☐ 126 et 767

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont toujours exacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée » et « résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et $(x \neq 1 \text{ ou } y \neq 0)$
- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de structurer le programme
- ☐ de rendre le programme moins lisible
- ☐ de paramétrer les actions
- ☐ de rendre le programme plus complexe
- ☐ de localiser les objets
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ formels
- ☐ effectifs
- ☐ variables
- ☐ effaçables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie uniquement à la dernière itération
- ☐ vraie à chaque itération
- ☐ jamais vraie
- ☐ vraie uniquement à la première itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le produit de a par b
- ☐ la probabilité que a soit égal à b
- ☐ le minimum de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N
- ☐ 0 et N-1
- ☐ 1 et N
- ☐ -1 et N+1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés itératifs **for**
- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 1 seul énoncé itératif **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ jamais vérifié
- ☐ parfois vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ toujours vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
- ☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Lisp
- ☐ Java
- ☐ C
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ on ne peut pas
- ☐ avec le compilateur
- ☐ en l'exécutant sur l'ordinateur

Question 4 En informatique, est-ce qu'un *type* ?

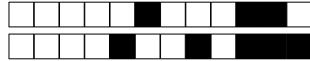
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une variable
- ☐ une constante
- ☐ une personne

Question 5 ♣ En C, un entier peut être de type :

- ☐ long
- ☐ double
- ☐ int
- ☐ très long
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -34
- ☐ -1
- ☐ -35
- ☐ 100



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ pair
- ☐ positif
- ☐ impair
- ☐ négatif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ aucune des deux
- ☐ 126 et 767
- ☐ 125 et 765
- ☐ -3 et 768

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont toujours exacts
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|a - b| < \epsilon$
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « formel donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et $(x \neq 1 \text{ ou } y \neq 0)$
- ☐ $x \neq 0$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de localiser les objets
- ☐ de rendre le programme plus complexe
- ☐ de rendre le programme moins lisible
- ☐ de paramétrer les actions
- ☐ de structurer le programme
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effaçables
- ☐ formels
- ☐ effectifs
- ☐ variables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie uniquement à la première itération
- ☐ jamais vraie
- ☐ vraie uniquement à la dernière itération
- ☐ vraie à chaque itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le minimum de a et b
- ☐ le produit de a par b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ la probabilité que a soit égal à b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N
- ☐ 0 et N-1
- ☐ 1 et N
- ☐ -1 et N+1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 3 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (i.e. `t[i]`) est :

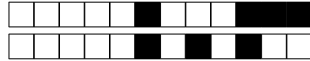
- ☐ toujours vérifié
- ☐ parfois vérifié
- ☐ jamais vérifié
- ☐ est vérifié si l'ordinateur est en forme

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
- ☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ C
- ☐ Lisp
- ☐ Java
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ en l'exécutant sur l'ordinateur
- ☐ on ne peut pas
- ☐ avec le compilateur
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents

Question 4 En informatique, est-ce qu'un *type* ?

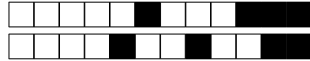
- ☐ une constante
- ☐ une personne
- ☐ une variable
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques

Question 5 ♣ En C, un entier peut être de type :

- ☐ double
- ☐ long
- ☐ très long
- ☐ int
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ 100
- ☐ -34
- ☐ -1
- ☐ -35



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ négatif
- ☐ positif
- ☐ pair
- ☐ impair

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 126 et 767
- ☐ aucune des deux
- ☐ -3 et 768
- ☐ 125 et 765

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont toujours exacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et $(x \neq 1 \text{ ou } y \neq 0)$
- ☐ $x \neq 0$
- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de structurer le programme
- ☐ de localiser les objets
- ☐ de rendre le programme plus complexe
- ☐ de rendre le programme moins lisible
- ☐ de paramétrer les actions
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ formels
- ☐ effaçables
- ☐ effectifs
- ☐ variables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie à chaque itération
- ☐ jamais vraie
- ☐ vraie uniquement à la première itération
- ☐ vraie uniquement à la dernière itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le produit de a par b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le minimum de a et b
- ☐ la probabilité que a soit égal à b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N
- ☐ 0 et N-1
- ☐ 1 et N
- ☐ -1 et N+1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 3 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (i.e. `t[i]`) est :

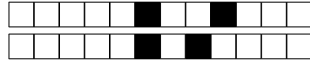
- ☐ jamais vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ parfois vérifié
- ☐ toujours vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
- ☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Java
- ☐ C
- ☐ Lisp
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ avec le compilateur
- ☐ en l'exécutant sur l'ordinateur
- ☐ on ne peut pas
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents

Question 4 En informatique, est-ce qu'un *type* ?

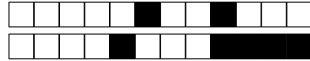
- ☐ une constante
- ☐ une variable
- ☐ une personne
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques

Question 5 ♣ En C, un entier peut être de type :

- ☐ double
- ☐ très long
- ☐ long
- ☐ int
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -1
- ☐ 100
- ☐ -35
- ☐ -34



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ positif
- ☐ négatif
- ☐ pair
- ☐ impair

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 126 et 767
- ☐ aucune des deux
- ☐ -3 et 768
- ☐ 125 et 765

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ les calculs avec les réels sont toujours exacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et $(x \neq 1 \text{ ou } y \neq 0)$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de rendre le programme moins lisible
- ☐ de rendre le programme plus complexe
- ☐ de localiser les objets
- ☐ de structurer le programme
- ☐ de paramétrer les actions
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ variables
- ☐ effaçables
- ☐ formels
- ☐ effectifs

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie uniquement à la première itération
- ☐ vraie à chaque itération
- ☐ vraie uniquement à la dernière itération
- ☐ jamais vraie

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ la probabilité que a soit égal à b
- ☐ le produit de a par b
- ☐ le minimum de a et b
- ☐ le pgcd (plus grand commun diviseur) de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N-1
- ☐ 1 et N
- ☐ -1 et N+1
- ☐ 0 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés **if-then-else**
- ☐ 3 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

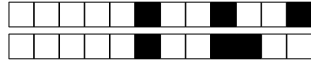
- ☐ jamais vérifié
- ☐ toujours vérifié
- ☐ parfois vérifié
- ☐ est vérifié si l'ordinateur est en forme

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016**Durée 1h30**

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ C
☐ Java
☐ Lisp
☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ en l'exécutant sur l'ordinateur
☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
☐ on ne peut pas
☐ avec le compilateur

Question 4 En informatique, est-ce qu'un *type* ?

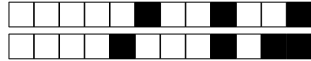
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
☐ une constante
☐ une personne
☐ une variable

Question 5 ♣ En C, un entier peut être de type :

- ☐ long
☐ int
☐ très long
☐ double
☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -35
☐ -34
☐ 100
☐ -1



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ négatif
- ☐ pair
- ☐ positif
- ☐ impair

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ aucune des deux
- ☐ -3 et 768
- ☐ 126 et 767
- ☐ 125 et 765

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont toujours exacts
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée » et « résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et $(x \neq 1 \text{ ou } y \neq 0)$
- ☐ $x \neq 0$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de localiser les objets
- ☐ de paramétrer les actions
- ☐ de structurer le programme
- ☐ de rendre le programme plus complexe
- ☐ de rendre le programme moins lisible
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effaçables
- ☐ variables
- ☐ formels
- ☐ effectifs

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie à chaque itération
- ☐ jamais vraie
- ☐ vraie uniquement à la dernière itération
- ☐ vraie uniquement à la première itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ la probabilité que a soit égal à b
- ☐ le minimum de a et b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le produit de a par b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ -1 et N+1
- ☐ 0 et N-1
- ☐ 1 et N
- ☐ 0 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ toujours vérifié
- ☐ jamais vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ parfois vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
- ☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Java
- ☐ C
- ☐ Lisp
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ on ne peut pas
- ☐ en l'exécutant sur l'ordinateur
- ☐ avec le compilateur
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une variable
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une personne
- ☐ une constante

Question 5 ♣ En C, un entier peut être de type :

- ☐ double
- ☐ int
- ☐ très long
- ☐ long
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -34
- ☐ 100
- ☐ -35
- ☐ -1



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ pair
- ☐ négatif
- ☐ impair
- ☐ positif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ aucune des deux
- ☐ 126 et 767
- ☐ -3 et 768
- ☐ 125 et 765

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont toujours exacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif donnée » et « résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x \neq 0$
- ☐ $x = 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de structurer le programme
- ☐ de localiser les objets
- ☐ de rendre le programme plus complexe
- ☐ de rendre le programme moins lisible
- ☐ de paramétrer les actions
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ formels
- ☐ variables
- ☐ effaçables
- ☐ effectifs

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ jamais vraie
- ☐ vraie uniquement à la première itération
- ☐ vraie à chaque itération
- ☐ vraie uniquement à la dernière itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le produit de a par b
- ☐ la probabilité que a soit égal à b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le minimum de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N-1
- ☐ 0 et N
- ☐ 1 et N
- ☐ -1 et N+1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 3 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (i.e. `t[i]`) est :

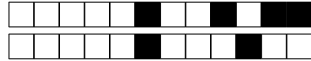
- ☐ est vérifié si l'ordinateur est en forme
- ☐ jamais vérifié
- ☐ toujours vérifié
- ☐ parfois vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si `x` $\in$ `t` et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
- ☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Lisp
- ☐ C
- ☐ Java
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ avec le compilateur
- ☐ en l'exécutant sur l'ordinateur
- ☐ on ne peut pas

Question 4 En informatique, est-ce qu'un *type* ?

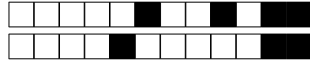
- ☐ une constante
- ☐ une personne
- ☐ une variable
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques

Question 5 ♣ En C, un entier peut être de type :

- ☐ très long
- ☐ int
- ☐ double
- ☐ long
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -34
- ☐ 100
- ☐ -35
- ☐ -1



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ positif
- ☐ impair
- ☐ négatif
- ☐ pair

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ -3 et 768
- ☐ 126 et 767
- ☐ aucune des deux
- ☐ 125 et 765

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont toujours exacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « formel donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de rendre le programme plus complexe
- ☐ de paramétrer les actions
- ☐ de structurer le programme
- ☐ de rendre le programme moins lisible
- ☐ de localiser les objets
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effaçables
- ☐ effectifs
- ☐ variables
- ☐ formels

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie à chaque itération
- ☐ jamais vraie
- ☐ vraie uniquement à la dernière itération
- ☐ vraie uniquement à la première itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ la probabilité que a soit égal à b
- ☐ le produit de a par b
- ☐ le minimum de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ -1 et N+1
- ☐ 1 et N
- ☐ 0 et N-1
- ☐ 0 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ toujours vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ jamais vérifié
- ☐ parfois vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
- ☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Java
- ☐ Lisp
- ☐ C
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ en l'exécutant sur l'ordinateur
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ avec le compilateur
- ☐ on ne peut pas

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une variable
- ☐ une personne
- ☐ une constante
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques

Question 5 ♣ En C, un entier peut être de type :

- ☐ long
- ☐ int
- ☐ très long
- ☐ double
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -34
- ☐ 100
- ☐ -1
- ☐ -35



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ impair
- ☐ positif
- ☐ négatif
- ☐ pair

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 126 et 767
- ☐ -3 et 768
- ☐ aucune des deux
- ☐ 125 et 765

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont toujours exacts
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « formel donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de structurer le programme
- ☐ de paramétrer les actions
- ☐ de rendre le programme moins lisible
- ☐ de localiser les objets
- ☐ de rendre le programme plus complexe
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ variables
- ☐ effaçables
- ☐ formels
- ☐ effectifs

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie à chaque itération
- ☐ vraie uniquement à la première itération
- ☐ jamais vraie
- ☐ vraie uniquement à la dernière itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le produit de a par b
- ☐ la probabilité que a soit égal à b
- ☐ le minimum de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N
- ☐ 1 et N
- ☐ -1 et N+1
- ☐ 0 et N-1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés **if-then-else**
- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ est vérifié si l'ordinateur est en forme
- ☐ parfois vérifié
- ☐ toujours vérifié
- ☐ jamais vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
- ☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Java
- ☐ C
- ☐ Lisp
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ en l'exécutant sur l'ordinateur
- ☐ avec le compilateur
- ☐ on ne peut pas
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une personne
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une constante
- ☐ une variable

Question 5 ♣ En C, un entier peut être de type :

- ☐ long
- ☐ très long
- ☐ int
- ☐ double
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -34
- ☐ 100
- ☐ -1
- ☐ -35



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ impair
- ☐ pair
- ☐ positif
- ☐ négatif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ aucune des deux
- ☐ 126 et 767
- ☐ -3 et 768
- ☐ 125 et 765

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont toujours exacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et $(x \neq 1 \text{ ou } y \neq 0)$
- ☐ $x \neq 0$
- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de paramétrer les actions
- ☐ de localiser les objets
- ☐ de structurer le programme
- ☐ de rendre le programme plus complexe
- ☐ de rendre le programme moins lisible
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ variables
- ☐ formels
- ☐ effaçables
- ☐ effectifs

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie uniquement à la première itération
- ☐ jamais vraie
- ☐ vraie uniquement à la dernière itération
- ☐ vraie à chaque itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le minimum de a et b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ la probabilité que a soit égal à b
- ☐ le produit de a par b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ -1 et N+1
- ☐ 0 et N-1
- ☐ 0 et N
- ☐ 1 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés **if-then-else**
- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ toujours vérifié
- ☐ parfois vérifié
- ☐ jamais vérifié
- ☐ est vérifié si l'ordinateur est en forme

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016**Durée 1h30**

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ C
☐ Lisp
☐ Java
☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
☐ en l'exécutant sur l'ordinateur
☐ on ne peut pas
☐ avec le compilateur

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une personne
☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
☐ une variable
☐ une constante

Question 5 ♣ En C, un entier peut être de type :

- ☐ int
☐ très long
☐ double
☐ long
☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ 100
☐ -1
☐ -34
☐ -35



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ pair
- ☐ positif
- ☐ négatif
- ☐ impair

Question 8 À quelle valeur décimale correspond les valeurs 0176 et 0x2FF ?

- ☐ 126 et 767
- ☐ 125 et 765
- ☐ aucune des deux
- ☐ -3 et 768

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ les calculs avec les réels sont toujours exacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et $(x \neq 1 \text{ ou } y \neq 0)$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ $x = 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de localiser les objets
- ☐ de paramétrer les actions
- ☐ de rendre le programme moins lisible
- ☐ de structurer le programme
- ☐ de rendre le programme plus complexe
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effaçables
- ☐ effectifs
- ☐ formels
- ☐ variables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ jamais vraie
- ☐ vraie uniquement à la première itération
- ☐ vraie à chaque itération
- ☐ vraie uniquement à la dernière itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ la probabilité que a soit égal à b
- ☐ le minimum de a et b
- ☐ le produit de a par b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N-1
- ☐ -1 et N+1
- ☐ 0 et N
- ☐ 1 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 1 seul énoncé itératif **for**
- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 2 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ toujours vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ jamais vérifié
- ☐ parfois vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ elle renvoie 1 si `x` $\in$ `t` et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016**Durée 1h30**

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ C
☐ Lisp
☐ Java
☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ avec le compilateur
☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
☐ en l'exécutant sur l'ordinateur
☐ on ne peut pas

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une personne
☐ une variable
☐ une constante
☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques

Question 5 ♣ En C, un entier peut être de type :

- ☐ int
☐ double
☐ très long
☐ long
☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -1
☐ -34
☐ -35
☐ 100



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ négatif
- ☐ pair
- ☐ positif
- ☐ impair

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ aucune des deux
- ☐ -3 et 768
- ☐ 125 et 765
- ☐ 126 et 767

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont toujours exacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « formel donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de paramétrer les actions
- ☐ de structurer le programme
- ☐ de localiser les objets
- ☐ de rendre le programme plus complexe
- ☐ de rendre le programme moins lisible
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé E sont des affirmations qui doivent :

- ☐ être toujours vraies avant l'exécution de E
- ☐ de temps en temps vraies avant et après l'exécution de E
- ☐ être toujours vraies, respectivement, avant et après l'exécution de E
- ☐ être toujours fausses avant et après l'exécution de E

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effaçables
- ☐ variables
- ☐ effectifs
- ☐ formels

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie à chaque itération
- ☐ vraie uniquement à la première itération
- ☐ jamais vraie
- ☐ vraie uniquement à la dernière itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le produit de a par b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ la probabilité que a soit égal à b
- ☐ le minimum de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N-1
- ☐ 0 et N
- ☐ -1 et N+1
- ☐ 1 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 3 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 2 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ jamais vérifié
- ☐ parfois vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ toujours vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
- ☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ C
- ☐ Java
- ☐ Lisp
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ on ne peut pas
- ☐ avec le compilateur
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ en l'exécutant sur l'ordinateur

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une personne
- ☐ une constante
- ☐ une variable
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques

Question 5 ♣ En C, un entier peut être de type :

- ☐ long
- ☐ très long
- ☐ int
- ☐ double
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ 100
- ☐ -1
- ☐ -35
- ☐ -34



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ impair
- ☐ négatif
- ☐ positif
- ☐ pair

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 125 et 765
- ☐ 126 et 767
- ☐ aucune des deux
- ☐ -3 et 768

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont toujours exacts
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de rendre le programme moins lisible
- ☐ de structurer le programme
- ☐ de rendre le programme plus complexe
- ☐ de localiser les objets
- ☐ de paramétrer les actions
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ variables
- ☐ effectifs
- ☐ effaçables
- ☐ formels

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie uniquement à la première itération
- ☐ vraie uniquement à la dernière itération
- ☐ vraie à chaque itération
- ☐ jamais vraie

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le produit de a par b
- ☐ le minimum de a et b
- ☐ la probabilité que a soit égal à b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N-1
- ☐ 0 et N
- ☐ 1 et N
- ☐ -1 et N+1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 1 seul énoncé itératif **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (i.e. `t[i]`) est :

- ☐ jamais vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ parfois vérifié
- ☐ toujours vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
- ☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Java
- ☐ Lisp
- ☐ C
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ on ne peut pas
- ☐ en l'exécutant sur l'ordinateur
- ☐ avec le compilateur
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une personne
- ☐ une constante
- ☐ une variable
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques

Question 5 ♣ En C, un entier peut être de type :

- ☐ long
- ☐ double
- ☐ très long
- ☐ int
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -35
- ☐ -34
- ☐ 100
- ☐ -1



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ négatif
- ☐ pair
- ☐ impair
- ☐ positif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 125 et 765
- ☐ aucune des deux
- ☐ -3 et 768
- ☐ 126 et 767

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont toujours exacts
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée » et « résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$
- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de rendre le programme plus complexe
- ☐ de paramétrer les actions
- ☐ de structurer le programme
- ☐ de rendre le programme moins lisible
- ☐ de localiser les objets
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé E sont des affirmations qui doivent :

- ☐ être toujours vraies avant l'exécution de E
- ☐ être toujours vraies, respectivement, avant et après l'exécution de E
- ☐ de temps en temps vraies avant et après l'exécution de E
- ☐ être toujours fausses avant et après l'exécution de E

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effaçables
- ☐ effectifs
- ☐ formels
- ☐ variables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ jamais vraie
- ☐ vraie uniquement à la dernière itération
- ☐ vraie uniquement à la première itération
- ☐ vraie à chaque itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le minimum de a et b
- ☐ le produit de a par b
- ☐ la probabilité que a soit égal à b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N-1
- ☐ -1 et N+1
- ☐ 0 et N
- ☐ 1 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 3 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ est vérifié si l'ordinateur est en forme
- ☐ parfois vérifié
- ☐ jamais vérifié
- ☐ toujours vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016**Durée 1h30**

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ C
☐ Lisp
☐ Java
☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ avec le compilateur
☐ en l'exécutant sur l'ordinateur
☐ on ne peut pas
☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
☐ une personne
☐ une constante
☐ une variable

Question 5 ♣ En C, un entier peut être de type :

- ☐ très long
☐ long
☐ double
☐ int
☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -35
☐ -1
☐ -34
☐ 100



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ négatif
- ☐ positif
- ☐ impair
- ☐ pair

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 126 et 767
- ☐ -3 et 768
- ☐ 125 et 765
- ☐ aucune des deux

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont toujours exacts
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif donnée » et « résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x \neq 0$
- ☐ $x = 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de rendre le programme moins lisible
- ☐ de localiser les objets
- ☐ de rendre le programme plus complexe
- ☐ de structurer le programme
- ☐ de paramétrer les actions
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effaçables
- ☐ formels
- ☐ effectifs
- ☐ variables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ jamais vraie
- ☐ vraie uniquement à la première itération
- ☐ vraie uniquement à la dernière itération
- ☐ vraie à chaque itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ la probabilité que a soit égal à b
- ☐ le minimum de a et b
- ☐ le produit de a par b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 1 et N
- ☐ 0 et N
- ☐ 0 et N-1
- ☐ -1 et N+1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 3 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ est vérifié si l'ordinateur est en forme
- ☐ jamais vérifié
- ☐ parfois vérifié
- ☐ toujours vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ Aucune de ces réponses n'est correcte.

**Examen du 28/10/2016****Durée 1h30**

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Lisp
☐ Java
☐ C
☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
☐ avec le compilateur
☐ en l'exécutant sur l'ordinateur
☐ on ne peut pas

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
☐ une personne
☐ une variable
☐ une constante

Question 5 ♣ En C, un entier peut être de type :

- ☐ int
☐ long
☐ très long
☐ double
☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ 100
☐ -35
☐ -1
☐ -34



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ positif
- ☐ pair
- ☐ négatif
- ☐ impair

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 126 et 767
- ☐ aucune des deux
- ☐ -3 et 768
- ☐ 125 et 765

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont toujours exacts
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « formel donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x = 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de rendre le programme plus complexe
- ☐ de localiser les objets
- ☐ de paramétrer les actions
- ☐ de rendre le programme moins lisible
- ☐ de structurer le programme
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effectifs
- ☐ formels
- ☐ effaçables
- ☐ variables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ jamais vraie
- ☐ vraie uniquement à la dernière itération
- ☐ vraie uniquement à la première itération
- ☐ vraie à chaque itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le minimum de a et b
- ☐ le produit de a par b
- ☐ la probabilité que a soit égal à b
- ☐ le pgcd (plus grand commun diviseur) de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 1 et N
- ☐ 0 et N
- ☐ -1 et N+1
- ☐ 0 et N-1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés **if-then-else**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ jamais vérifié
- ☐ toujours vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ parfois vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016**Durée 1h30**

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Java
☐ Lisp
☐ C
☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ en l'exécutant sur l'ordinateur
☐ avec le compilateur
☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
☐ on ne peut pas

Question 4 En informatique, est-ce qu'un *type* ?

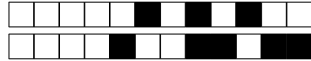
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
☐ une personne
☐ une variable
☐ une constante

Question 5 ♣ En C, un entier peut être de type :

- ☐ int
☐ double
☐ long
☐ très long
☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -34
☐ 100
☐ -1
☐ -35



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ négatif
- ☐ impair
- ☐ positif
- ☐ pair

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ -3 et 768
- ☐ aucune des deux
- ☐ 125 et 765
- ☐ 126 et 767

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont toujours exacts
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif donnée » et « résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et $(x \neq 1 \text{ ou } y \neq 0)$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de structurer le programme
- ☐ de rendre le programme moins lisible
- ☐ de rendre le programme plus complexe
- ☐ de localiser les objets
- ☐ de paramétrer les actions
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ variables
- ☐ effaçables
- ☐ formels
- ☐ effectifs

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

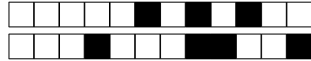
- ☐ vraie à chaque itération
- ☐ vraie uniquement à la dernière itération
- ☐ vraie uniquement à la première itération
- ☐ jamais vraie

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le minimum de a et b
- ☐ le produit de a par b
- ☐ la probabilité que a soit égal à b
- ☐ le pgcd (plus grand commun diviseur) de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N-1
- ☐ 1 et N
- ☐ 0 et N
- ☐ -1 et N+1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés **if-then-else**
- ☐ 1 seul énoncé itératif **for**
- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

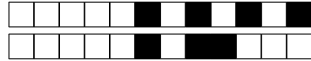
- ☐ toujours vérifié
- ☐ parfois vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ jamais vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
- ☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Java
- ☐ Lisp
- ☐ C
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ avec le compilateur
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ on ne peut pas
- ☐ en l'exécutant sur l'ordinateur

Question 4 En informatique, est-ce qu'un *type* ?

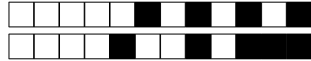
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une personne
- ☐ une constante
- ☐ une variable

Question 5 ♣ En C, un entier peut être de type :

- ☐ très long
- ☐ long
- ☐ int
- ☐ double
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -34
- ☐ -1
- ☐ -35
- ☐ 100



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ négatif
- ☐ positif
- ☐ impair
- ☐ pair

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 125 et 765
- ☐ 126 et 767
- ☐ -3 et 768
- ☐ aucune des deux

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont toujours exacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$
- ☐ $x = 0$ et $(x \neq 1 \text{ ou } y \neq 0)$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de rendre le programme moins lisible
- ☐ de paramétrer les actions
- ☐ de structurer le programme
- ☐ de localiser les objets
- ☐ de rendre le programme plus complexe
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ variables
- ☐ formels
- ☐ effaçables
- ☐ effectifs

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie à chaque itération
- ☐ vraie uniquement à la dernière itération
- ☐ vraie uniquement à la première itération
- ☐ jamais vraie

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le produit de a par b
- ☐ le minimum de a et b
- ☐ la probabilité que a soit égal à b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N
- ☐ 0 et N-1
- ☐ 1 et N
- ☐ -1 et N+1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 1 seul énoncé itératif **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

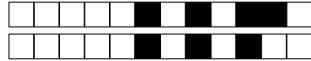
- ☐ jamais vérifié
- ☐ toujours vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ parfois vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
- ☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Java
- ☐ C
- ☐ Lisp
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ avec le compilateur
- ☐ on ne peut pas
- ☐ en l'exécutant sur l'ordinateur
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents

Question 4 En informatique, est-ce qu'un *type* ?

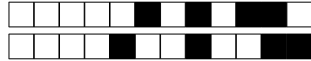
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une variable
- ☐ une constante
- ☐ une personne

Question 5 ♣ En C, un entier peut être de type :

- ☐ long
- ☐ double
- ☐ très long
- ☐ int
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -1
- ☐ -34
- ☐ -35
- ☐ 100



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ pair
- ☐ positif
- ☐ négatif
- ☐ impair

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ aucune des deux
- ☐ -3 et 768
- ☐ 125 et 765
- ☐ 126 et 767

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont toujours exacts
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x \neq 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de paramétrer les actions
- ☐ de localiser les objets
- ☐ de rendre le programme moins lisible
- ☐ de rendre le programme plus complexe
- ☐ de structurer le programme
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ formels
- ☐ effaçables
- ☐ variables
- ☐ effectifs

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie uniquement à la première itération
- ☐ vraie uniquement à la dernière itération
- ☐ jamais vraie
- ☐ vraie à chaque itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ la probabilité que a soit égal à b
- ☐ le produit de a par b
- ☐ le minimum de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N
- ☐ 0 et N-1
- ☐ 1 et N
- ☐ -1 et N+1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 1 seul énoncé itératif **for**
- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 2 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

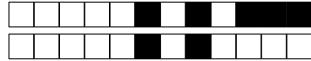
- ☐ est vérifié si l'ordinateur est en forme
- ☐ toujours vérifié
- ☐ parfois vérifié
- ☐ jamais vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
- ☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Java
- ☐ Lisp
- ☐ C
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ on ne peut pas
- ☐ en l'exécutant sur l'ordinateur
- ☐ avec le compilateur
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents

Question 4 En informatique, est-ce qu'un *type* ?

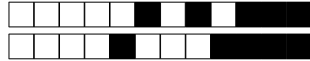
- ☐ une personne
- ☐ une variable
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une constante

Question 5 ♣ En C, un entier peut être de type :

- ☐ long
- ☐ très long
- ☐ int
- ☐ double
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ 100
- ☐ -1
- ☐ -34
- ☐ -35



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ pair
- ☐ impair
- ☐ négatif
- ☐ positif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 126 et 767
- ☐ aucune des deux
- ☐ 125 et 765
- ☐ -3 et 768

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ les calculs avec les réels sont toujours exacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée » et « résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ $x = 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de paramétrer les actions
- ☐ de structurer le programme
- ☐ de localiser les objets
- ☐ de rendre le programme plus complexe
- ☐ de rendre le programme moins lisible
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ variables
- ☐ formels
- ☐ effaçables
- ☐ effectifs

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie uniquement à la dernière itération
- ☐ jamais vraie
- ☐ vraie à chaque itération
- ☐ vraie uniquement à la première itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le produit de a par b
- ☐ le minimum de a et b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ la probabilité que a soit égal à b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N-1
- ☐ 1 et N
- ☐ 0 et N
- ☐ -1 et N+1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés **if-then-else**
- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés itératifs **for**
- ☐ 3 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

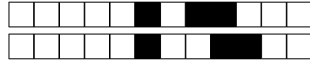
- ☐ toujours vérifié
- ☐ jamais vérifié
- ☐ parfois vérifié
- ☐ est vérifié si l'ordinateur est en forme

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016**Durée 1h30**

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ C
☐ Java
☐ Lisp
☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ en l'exécutant sur l'ordinateur
☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
☐ on ne peut pas
☐ avec le compilateur

Question 4 En informatique, est-ce qu'un *type* ?

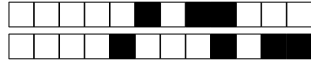
- ☐ une personne
☐ une variable
☐ une constante
☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques

Question 5 ♣ En C, un entier peut être de type :

- ☐ long
☐ double
☐ très long
☐ int
☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -34
☐ 100
☐ -1
☐ -35



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ positif
- ☐ négatif
- ☐ impair
- ☐ pair

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 125 et 765
- ☐ -3 et 768
- ☐ 126 et 767
- ☐ aucune des deux

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont toujours exacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|a - b| < \epsilon$
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « formel donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x \neq 0$
- ☐ $x \neq 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de localiser les objets
- ☐ de structurer le programme
- ☐ de rendre le programme moins lisible
- ☐ de rendre le programme plus complexe
- ☐ de paramétrer les actions
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ formels
- ☐ effaçables
- ☐ variables
- ☐ effectifs

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie à chaque itération
- ☐ vraie uniquement à la dernière itération
- ☐ jamais vraie
- ☐ vraie uniquement à la première itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le minimum de a et b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le produit de a par b
- ☐ la probabilité que a soit égal à b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N-1
- ☐ 1 et N
- ☐ -1 et N+1
- ☐ 0 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 2 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

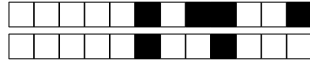
- ☐ toujours vérifié
- ☐ parfois vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ jamais vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ Aucune de ces réponses n'est correcte.

**Examen du 28/10/2016****Durée 1h30**

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ C
☐ Lisp
☐ Java
☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ on ne peut pas
☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
☐ avec le compilateur
☐ en l'exécutant sur l'ordinateur

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une variable
☐ une constante
☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
☐ une personne

Question 5 ♣ En C, un entier peut être de type :

- ☐ long
☐ double
☐ int
☐ très long
☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ 100
☐ -35
☐ -34
☐ -1



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ négatif
- ☐ pair
- ☐ impair
- ☐ positif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 126 et 767
- ☐ aucune des deux
- ☐ -3 et 768
- ☐ 125 et 765

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont toujours exacts
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif donnée » et « résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$
- ☐ $x = 0$ et $(x \neq 1 \text{ ou } y \neq 0)$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de localiser les objets
- ☐ de paramétrer les actions
- ☐ de rendre le programme plus complexe
- ☐ de rendre le programme moins lisible
- ☐ de structurer le programme
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effaçables
- ☐ formels
- ☐ variables
- ☐ effectifs

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie à chaque itération
- ☐ jamais vraie
- ☐ vraie uniquement à la première itération
- ☐ vraie uniquement à la dernière itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le produit de a par b
- ☐ le minimum de a et b
- ☐ la probabilité que a soit égal à b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N-1
- ☐ 0 et N
- ☐ -1 et N+1
- ☐ 1 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**
- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

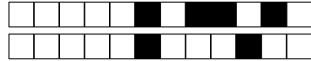
- ☐ parfois vérifié
- ☐ jamais vérifié
- ☐ toujours vérifié
- ☐ est vérifié si l'ordinateur est en forme

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ elle renvoie 1 si `x` $\in$ `t` et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
- ☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Lisp
- ☐ C
- ☐ Java
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ on ne peut pas
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ avec le compilateur
- ☐ en l'exécutant sur l'ordinateur

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une constante
- ☐ une personne
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une variable

Question 5 ♣ En C, un entier peut être de type :

- ☐ très long
- ☐ int
- ☐ long
- ☐ double
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -34
- ☐ 100
- ☐ -35
- ☐ -1



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ impair
- ☐ pair
- ☐ négatif
- ☐ positif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 126 et 767
- ☐ aucune des deux
- ☐ -3 et 768
- ☐ 125 et 765

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont toujours exacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de localiser les objets
- ☐ de rendre le programme moins lisible
- ☐ de structurer le programme
- ☐ de rendre le programme plus complexe
- ☐ de paramétrer les actions
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ variables
- ☐ effectifs
- ☐ formels
- ☐ effaçables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie uniquement à la première itération
- ☐ jamais vraie
- ☐ vraie uniquement à la dernière itération
- ☐ vraie à chaque itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ la probabilité que a soit égal à b
- ☐ le minimum de a et b
- ☐ le produit de a par b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N-1
- ☐ -1 et N+1
- ☐ 0 et N
- ☐ 1 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 1 seul énoncé itératif **for**
- ☐ 3 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ toujours vérifié
- ☐ jamais vérifié
- ☐ parfois vérifié
- ☐ est vérifié si l'ordinateur est en forme

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
- ☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Java
- ☐ C
- ☐ Lisp
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ on ne peut pas
- ☐ avec le compilateur
- ☐ en l'exécutant sur l'ordinateur
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une personne
- ☐ une variable
- ☐ une constante
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques

Question 5 ♣ En C, un entier peut être de type :

- ☐ int
- ☐ long
- ☐ très long
- ☐ double
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ 100
- ☐ -35
- ☐ -1
- ☐ -34



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ positif
- ☐ négatif
- ☐ pair
- ☐ impair

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 125 et 765
- ☐ -3 et 768
- ☐ 126 et 767
- ☐ aucune des deux

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ les calculs avec les réels sont toujours exacts
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de structurer le programme
- ☐ de localiser les objets
- ☐ de paramétrer les actions
- ☐ de rendre le programme moins lisible
- ☐ de rendre le programme plus complexe
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effectifs
- ☐ formels
- ☐ effaçables
- ☐ variables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie à chaque itération
- ☐ jamais vraie
- ☐ vraie uniquement à la dernière itération
- ☐ vraie uniquement à la première itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le produit de a par b
- ☐ la probabilité que a soit égal à b
- ☐ le minimum de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N-1
- ☐ 0 et N
- ☐ 1 et N
- ☐ -1 et N+1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ parfois vérifié
- ☐ jamais vérifié
- ☐ toujours vérifié
- ☐ est vérifié si l'ordinateur est en forme

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
- ☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Lisp
- ☐ Java
- ☐ C
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ en l'exécutant sur l'ordinateur
- ☐ on ne peut pas
- ☐ avec le compilateur
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une personne
- ☐ une constante
- ☐ une variable

Question 5 ♣ En C, un entier peut être de type :

- ☐ double
- ☐ très long
- ☐ long
- ☐ int
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -1
- ☐ -35
- ☐ 100
- ☐ -34



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ impair
- ☐ positif
- ☐ pair
- ☐ négatif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 125 et 765
- ☐ 126 et 767
- ☐ -3 et 768
- ☐ aucune des deux

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ les calculs avec les réels sont toujours exacts
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée » et « résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ $x = 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de rendre le programme plus complexe
- ☐ de paramétrer les actions
- ☐ de structurer le programme
- ☐ de localiser les objets
- ☐ de rendre le programme moins lisible
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ variables
- ☐ formels
- ☐ effaçables
- ☐ effectifs

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie à chaque itération
- ☐ jamais vraie
- ☐ vraie uniquement à la dernière itération
- ☐ vraie uniquement à la première itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ la probabilité que a soit égal à b
- ☐ le minimum de a et b
- ☐ le produit de a par b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N-1
- ☐ 0 et N
- ☐ 1 et N
- ☐ -1 et N+1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 3 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ jamais vérifié
- ☐ toujours vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ parfois vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
- ☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Lisp
- ☐ Java
- ☐ C
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ avec le compilateur
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ en l'exécutant sur l'ordinateur
- ☐ on ne peut pas

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une variable
- ☐ une constante
- ☐ une personne

Question 5 ♣ En C, un entier peut être de type :

- ☐ double
- ☐ très long
- ☐ int
- ☐ long
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -34
- ☐ -35
- ☐ -1
- ☐ 100



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ impair
- ☐ pair
- ☐ négatif
- ☐ positif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 125 et 765
- ☐ -3 et 768
- ☐ 126 et 767
- ☐ aucune des deux

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont toujours exacts
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$
- ☐ $x = 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x \neq 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de paramétrer les actions
- ☐ de structurer le programme
- ☐ de rendre le programme plus complexe
- ☐ de localiser les objets
- ☐ de rendre le programme moins lisible
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ variables
- ☐ formels
- ☐ effaçables
- ☐ effectifs

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie uniquement à la dernière itération
- ☐ vraie uniquement à la première itération
- ☐ vraie à chaque itération
- ☐ jamais vraie

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ la probabilité que a soit égal à b
- ☐ le minimum de a et b
- ☐ le produit de a par b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 1 et N
- ☐ -1 et N+1
- ☐ 0 et N-1
- ☐ 0 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 3 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ toujours vérifié
- ☐ jamais vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ parfois vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
- ☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Java
- ☐ C
- ☐ Lisp
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ on ne peut pas
- ☐ en l'exécutant sur l'ordinateur
- ☐ avec le compilateur

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une constante
- ☐ une personne
- ☐ une variable

Question 5 ♣ En C, un entier peut être de type :

- ☐ très long
- ☐ long
- ☐ double
- ☐ int
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -1
- ☐ -34
- ☐ -35
- ☐ 100



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ négatif
- ☐ positif
- ☐ pair
- ☐ impair

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ -3 et 768
- ☐ aucune des deux
- ☐ 126 et 767
- ☐ 125 et 765

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ les calculs avec les réels sont toujours exacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « formel donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ $x = 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de paramétrer les actions
- ☐ de rendre le programme plus complexe
- ☐ de localiser les objets
- ☐ de structurer le programme
- ☐ de rendre le programme moins lisible
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ formels
- ☐ effaçables
- ☐ effectifs
- ☐ variables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie à chaque itération
- ☐ vraie uniquement à la dernière itération
- ☐ vraie uniquement à la première itération
- ☐ jamais vraie

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le produit de a par b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le minimum de a et b
- ☐ la probabilité que a soit égal à b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ -1 et N+1
- ☐ 0 et N-1
- ☐ 0 et N
- ☐ 1 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés **if-then-else**
- ☐ 2 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**
- ☐ 3 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ jamais vérifié
- ☐ toujours vérifié
- ☐ parfois vérifié
- ☐ est vérifié si l'ordinateur est en forme

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
- ☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Java
- ☐ C
- ☐ Lisp
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ avec le compilateur
- ☐ on ne peut pas
- ☐ en l'exécutant sur l'ordinateur
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une variable
- ☐ une constante
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une personne

Question 5 ♣ En C, un entier peut être de type :

- ☐ très long
- ☐ double
- ☐ long
- ☐ int
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -1
- ☐ -34
- ☐ 100
- ☐ -35



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ pair
- ☐ impair
- ☐ négatif
- ☐ positif

Question 8 À quelle valeur décimale correspond les valeurs 0176 et 0x2FF ?

- ☐ 125 et 765
- ☐ -3 et 768
- ☐ aucune des deux
- ☐ 126 et 767

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont toujours exacts
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « formel donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de paramétrer les actions
- ☐ de rendre le programme moins lisible
- ☐ de localiser les objets
- ☐ de structurer le programme
- ☐ de rendre le programme plus complexe
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ variables
- ☐ effectifs
- ☐ formels
- ☐ effaçables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie à chaque itération
- ☐ vraie uniquement à la dernière itération
- ☐ vraie uniquement à la première itération
- ☐ jamais vraie

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le minimum de a et b
- ☐ le produit de a par b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ la probabilité que a soit égal à b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N-1
- ☐ 0 et N
- ☐ -1 et N+1
- ☐ 1 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés **if-then-else**
- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés itératifs **for**
- ☐ 3 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ jamais vérifié
- ☐ parfois vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ toujours vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016**Durée 1h30**

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ C
☐ Java
☐ Lisp
☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ avec le compilateur
☐ on ne peut pas
☐ en l'exécutant sur l'ordinateur
☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une variable
☐ une constante
☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
☐ une personne

Question 5 ♣ En C, un entier peut être de type :

- ☐ double
☐ long
☐ très long
☐ int
☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ 100
☐ -34
☐ -35
☐ -1



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ pair
- ☐ négatif
- ☐ positif
- ☐ impair

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ aucune des deux
- ☐ -3 et 768
- ☐ 125 et 765
- ☐ 126 et 767

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ les calculs avec les réels sont toujours exacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « formel donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et $(x \neq 1 \text{ ou } y \neq 0)$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de localiser les objets
- ☐ de rendre le programme plus complexe
- ☐ de rendre le programme moins lisible
- ☐ de paramétrer les actions
- ☐ de structurer le programme
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ formels
- ☐ variables
- ☐ effaçables
- ☐ effectifs

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie uniquement à la dernière itération
- ☐ jamais vraie
- ☐ vraie à chaque itération
- ☐ vraie uniquement à la première itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ la probabilité que a soit égal à b
- ☐ le minimum de a et b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le produit de a par b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ -1 et N+1
- ☐ 0 et N
- ☐ 0 et N-1
- ☐ 1 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 1 seul énoncé itératif **for**
- ☐ 3 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ jamais vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ toujours vérifié
- ☐ parfois vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
- ☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Java
- ☐ Lisp
- ☐ C
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ avec le compilateur
- ☐ on ne peut pas
- ☐ en l'exécutant sur l'ordinateur

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une variable
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une constante
- ☐ une personne

Question 5 ♣ En C, un entier peut être de type :

- ☐ long
- ☐ double
- ☐ très long
- ☐ int
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -1
- ☐ -34
- ☐ 100
- ☐ -35



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ impair
- ☐ positif
- ☐ pair
- ☐ négatif

Question 8 À quelle valeur décimale correspond les valeurs 0176 et 0x2FF ?

- ☐ 126 et 767
- ☐ 125 et 765
- ☐ aucune des deux
- ☐ -3 et 768

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ les calculs avec les réels sont toujours exacts
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$
- ☐ $x = 0$ et $(x \neq 1 \text{ ou } y \neq 0)$
- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de structurer le programme
- ☐ de rendre le programme plus complexe
- ☐ de rendre le programme moins lisible
- ☐ de localiser les objets
- ☐ de paramétrer les actions
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effaçables
- ☐ variables
- ☐ formels
- ☐ effectifs

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie uniquement à la première itération
- ☐ jamais vraie
- ☐ vraie uniquement à la dernière itération
- ☐ vraie à chaque itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ la probabilité que a soit égal à b
- ☐ le produit de a par b
- ☐ le minimum de a et b
- ☐ le pgcd (plus grand commun diviseur) de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 1 et N
- ☐ -1 et N+1
- ☐ 0 et N
- ☐ 0 et N-1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés **if-then-else**
- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ parfois vérifié
- ☐ jamais vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ toujours vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
- ☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Lisp
- ☐ C
- ☐ Java
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ on ne peut pas
- ☐ en l'exécutant sur l'ordinateur
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ avec le compilateur

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une constante
- ☐ une variable
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une personne

Question 5 ♣ En C, un entier peut être de type :

- ☐ double
- ☐ très long
- ☐ long
- ☐ int
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -1
- ☐ -35
- ☐ 100
- ☐ -34



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ impair
- ☐ pair
- ☐ négatif
- ☐ positif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ aucune des deux
- ☐ -3 et 768
- ☐ 125 et 765
- ☐ 126 et 767

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont toujours exacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de rendre le programme moins lisible
- ☐ de structurer le programme
- ☐ de paramétrer les actions
- ☐ de rendre le programme plus complexe
- ☐ de localiser les objets
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effaçables
- ☐ effectifs
- ☐ formels
- ☐ variables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie uniquement à la première itération
- ☐ vraie uniquement à la dernière itération
- ☐ vraie à chaque itération
- ☐ jamais vraie

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ la probabilité que a soit égal à b
- ☐ le produit de a par b
- ☐ le minimum de a et b
- ☐ le pgcd (plus grand commun diviseur) de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N-1
- ☐ 0 et N
- ☐ 1 et N
- ☐ -1 et N+1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**
- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ est vérifié si l'ordinateur est en forme
- ☐ parfois vérifié
- ☐ toujours vérifié
- ☐ jamais vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
- ☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Java
- ☐ Lisp
- ☐ C
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ en l'exécutant sur l'ordinateur
- ☐ avec le compilateur
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ on ne peut pas

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une variable
- ☐ une personne
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une constante

Question 5 ♣ En C, un entier peut être de type :

- ☐ long
- ☐ int
- ☐ double
- ☐ très long
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -1
- ☐ -35
- ☐ -34
- ☐ 100



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ pair
- ☐ positif
- ☐ impair
- ☐ négatif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 125 et 765
- ☐ -3 et 768
- ☐ aucune des deux
- ☐ 126 et 767

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont toujours exacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif donnée » et « résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de rendre le programme plus complexe
- ☐ de structurer le programme
- ☐ de localiser les objets
- ☐ de rendre le programme moins lisible
- ☐ de paramétrer les actions
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé E sont des affirmations qui doivent :

- ☐ être toujours vraies avant l'exécution de E
- ☐ être toujours vraies, respectivement, avant et après l'exécution de E
- ☐ de temps en temps vraies avant et après l'exécution de E
- ☐ être toujours fausses avant et après l'exécution de E

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ variables
- ☐ effectifs
- ☐ effaçables
- ☐ formels

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie uniquement à la première itération
- ☐ vraie à chaque itération
- ☐ jamais vraie
- ☐ vraie uniquement à la dernière itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le produit de a par b
- ☐ le minimum de a et b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ la probabilité que a soit égal à b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ -1 et N+1
- ☐ 0 et N-1
- ☐ 0 et N
- ☐ 1 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés **if-then-else**
- ☐ 1 seul énoncé itératif **for**
- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ parfois vérifié
- ☐ jamais vérifié
- ☐ toujours vérifié
- ☐ est vérifié si l'ordinateur est en forme

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ elle ne fait rien et ne renvoie rien
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
- ☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Java
- ☐ Lisp
- ☐ C
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ avec le compilateur
- ☐ on ne peut pas
- ☐ en l'exécutant sur l'ordinateur

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une variable
- ☐ une constante
- ☐ une personne
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques

Question 5 ♣ En C, un entier peut être de type :

- ☐ double
- ☐ long
- ☐ très long
- ☐ int
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -1
- ☐ -35
- ☐ -34
- ☐ 100



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ pair
- ☐ impair
- ☐ positif
- ☐ négatif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 126 et 767
- ☐ aucune des deux
- ☐ -3 et 768
- ☐ 125 et 765

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont toujours exacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ $x \neq 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de localiser les objets
- ☐ de structurer le programme
- ☐ de paramétrer les actions
- ☐ de rendre le programme moins lisible
- ☐ de rendre le programme plus complexe
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ variables
- ☐ formels
- ☐ effaçables
- ☐ effectifs

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie uniquement à la dernière itération
- ☐ vraie à chaque itération
- ☐ jamais vraie
- ☐ vraie uniquement à la première itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le produit de a par b
- ☐ le minimum de a et b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ la probabilité que a soit égal à b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N-1
- ☐ -1 et N+1
- ☐ 1 et N
- ☐ 0 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés **if-then-else**
- ☐ 2 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**
- ☐ 3 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ est vérifié si l'ordinateur est en forme
- ☐ parfois vérifié
- ☐ jamais vérifié
- ☐ toujours vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
- ☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ C
- ☐ Java
- ☐ Lisp
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ on ne peut pas
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ en l'exécutant sur l'ordinateur
- ☐ avec le compilateur

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une variable
- ☐ une personne
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une constante

Question 5 ♣ En C, un entier peut être de type :

- ☐ très long
- ☐ double
- ☐ long
- ☐ int
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -34
- ☐ 100
- ☐ -1
- ☐ -35



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ pair
- ☐ positif
- ☐ impair
- ☐ négatif

Question 8 À quelle valeur décimale correspond les valeurs 0176 et 0x2FF ?

- ☐ aucune des deux
- ☐ 126 et 767
- ☐ 125 et 765
- ☐ -3 et 768

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont toujours exacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée » et « résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x = 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de paramétrer les actions
- ☐ de rendre le programme plus complexe
- ☐ de localiser les objets
- ☐ de rendre le programme moins lisible
- ☐ de structurer le programme
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ variables
- ☐ formels
- ☐ effaçables
- ☐ effectifs

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ jamais vraie
- ☐ vraie uniquement à la première itération
- ☐ vraie uniquement à la dernière itération
- ☐ vraie à chaque itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ la probabilité que a soit égal à b
- ☐ le produit de a par b
- ☐ le minimum de a et b
- ☐ le pgcd (plus grand commun diviseur) de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 1 et N
- ☐ -1 et N+1
- ☐ 0 et N
- ☐ 0 et N-1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés itératifs **for**
- ☐ 3 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés **if-then-else**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ parfois vérifié
- ☐ jamais vérifié
- ☐ toujours vérifié
- ☐ est vérifié si l'ordinateur est en forme

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
- ☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ C
- ☐ Lisp
- ☐ Java
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ avec le compilateur
- ☐ on ne peut pas
- ☐ en l'exécutant sur l'ordinateur

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une constante
- ☐ une personne
- ☐ une variable
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques

Question 5 ♣ En C, un entier peut être de type :

- ☐ long
- ☐ double
- ☐ int
- ☐ très long
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ 100
- ☐ -1
- ☐ -35
- ☐ -34



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ positif
- ☐ négatif
- ☐ impair
- ☐ pair

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ aucune des deux
- ☐ 125 et 765
- ☐ -3 et 768
- ☐ 126 et 767

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont toujours exacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif donnée » et « résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et $y = 2$
- ☐ $x = 0$ et $(x \neq 1$ ou $y \neq 0)$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de rendre le programme moins lisible
- ☐ de paramétrer les actions
- ☐ de structurer le programme
- ☐ de localiser les objets
- ☐ de rendre le programme plus complexe
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effaçables
- ☐ effectifs
- ☐ formels
- ☐ variables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie à chaque itération
- ☐ vraie uniquement à la dernière itération
- ☐ jamais vraie
- ☐ vraie uniquement à la première itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le produit de a par b
- ☐ le minimum de a et b
- ☐ la probabilité que a soit égal à b
- ☐ le pgcd (plus grand commun diviseur) de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ -1 et N+1
- ☐ 1 et N
- ☐ 0 et N
- ☐ 0 et N-1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 3 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ jamais vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ parfois vérifié
- ☐ toujours vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
- ☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Lisp
- ☐ Java
- ☐ C
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ on ne peut pas
- ☐ en l'exécutant sur l'ordinateur
- ☐ avec le compilateur

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une constante
- ☐ une variable
- ☐ une personne
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques

Question 5 ♣ En C, un entier peut être de type :

- ☐ long
- ☐ très long
- ☐ int
- ☐ double
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -35
- ☐ 100
- ☐ -34
- ☐ -1



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ positif
- ☐ impair
- ☐ négatif
- ☐ pair

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ aucune des deux
- ☐ -3 et 768
- ☐ 125 et 765
- ☐ 126 et 767

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ les calculs avec les réels sont toujours exacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « formel donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x \neq 0$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de rendre le programme plus complexe
- ☐ de rendre le programme moins lisible
- ☐ de structurer le programme
- ☐ de paramétrer les actions
- ☐ de localiser les objets
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effaçables
- ☐ variables
- ☐ formels
- ☐ effectifs

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie uniquement à la dernière itération
- ☐ vraie à chaque itération
- ☐ jamais vraie
- ☐ vraie uniquement à la première itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le minimum de a et b
- ☐ la probabilité que a soit égal à b
- ☐ le produit de a par b
- ☐ le pgcd (plus grand commun diviseur) de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N
- ☐ -1 et N+1
- ☐ 1 et N
- ☐ 0 et N-1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés **if-then-else**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ est vérifié si l'ordinateur est en forme
- ☐ jamais vérifié
- ☐ parfois vérifié
- ☐ toujours vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
- ☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Lisp
- ☐ C
- ☐ Java
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ en l'exécutant sur l'ordinateur
- ☐ on ne peut pas
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ avec le compilateur

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une variable
- ☐ une constante
- ☐ une personne

Question 5 ♣ En C, un entier peut être de type :

- ☐ très long
- ☐ long
- ☐ int
- ☐ double
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -34
- ☐ -1
- ☐ -35
- ☐ 100



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ positif
- ☐ impair
- ☐ pair
- ☐ négatif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ -3 et 768
- ☐ aucune des deux
- ☐ 126 et 767
- ☐ 125 et 765

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ les calculs avec les réels sont toujours exacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|a - b| < \epsilon$
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x \neq 0$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de rendre le programme moins lisible
- ☐ de structurer le programme
- ☐ de paramétrer les actions
- ☐ de rendre le programme plus complexe
- ☐ de localiser les objets
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ variables
- ☐ effectifs
- ☐ effaçables
- ☐ formels

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie uniquement à la dernière itération
- ☐ vraie à chaque itération
- ☐ jamais vraie
- ☐ vraie uniquement à la première itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le minimum de a et b
- ☐ la probabilité que a soit égal à b
- ☐ le produit de a par b
- ☐ le pgcd (plus grand commun diviseur) de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ -1 et N+1
- ☐ 1 et N
- ☐ 0 et N
- ☐ 0 et N-1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés **if-then-else**
- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés itératifs **for**
- ☐ 3 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ toujours vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ parfois vérifié
- ☐ jamais vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016**Durée 1h30**

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ C
☐ Java
☐ Lisp
☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ on ne peut pas
☐ avec le compilateur
☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
☐ en l'exécutant sur l'ordinateur

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une variable
☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
☐ une constante
☐ une personne

Question 5 ♣ En C, un entier peut être de type :

- ☐ long
☐ int
☐ très long
☐ double
☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -34
☐ -35
☐ 100
☐ -1



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ négatif
- ☐ pair
- ☐ positif
- ☐ impair

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ -3 et 768
- ☐ aucune des deux
- ☐ 125 et 765
- ☐ 126 et 767

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont toujours exacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « effectif donnée » et « résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de structurer le programme
- ☐ de rendre le programme plus complexe
- ☐ de localiser les objets
- ☐ de paramétrer les actions
- ☐ de rendre le programme moins lisible
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ effectifs
- ☐ effaçables
- ☐ variables
- ☐ formels

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie uniquement à la dernière itération
- ☐ vraie uniquement à la première itération
- ☐ vraie à chaque itération
- ☐ jamais vraie

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le produit de a par b
- ☐ la probabilité que a soit égal à b
- ☐ le minimum de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ -1 et N+1
- ☐ 0 et N
- ☐ 0 et N-1
- ☐ 1 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés itératifs **for**
- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ jamais vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ parfois vérifié
- ☐ toujours vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ vrai
- ☐ faux

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Java
- ☐ C
- ☐ Lisp
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ avec le compilateur
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ en l'exécutant sur l'ordinateur
- ☐ on ne peut pas

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une personne
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une constante
- ☐ une variable

Question 5 ♣ En C, un entier peut être de type :

- ☐ long
- ☐ int
- ☐ double
- ☐ très long
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -34
- ☐ -1
- ☐ 100
- ☐ -35



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ positif
- ☐ pair
- ☐ impair
- ☐ négatif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ aucune des deux
- ☐ 126 et 767
- ☐ 125 et 765
- ☐ -3 et 768

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ les calculs avec les réels sont toujours exacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x = 0$ et $(x \neq 1 \text{ ou } y \neq 0)$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de rendre le programme plus complexe
- ☐ de structurer le programme
- ☐ de localiser les objets
- ☐ de rendre le programme moins lisible
- ☐ de paramétrer les actions
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé E sont des affirmations qui doivent :

- ☐ être toujours fausses avant et après l'exécution de E
- ☐ de temps en temps vraies avant et après l'exécution de E
- ☐ être toujours vraies, respectivement, avant et après l'exécution de E
- ☐ être toujours vraies avant l'exécution de E

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ formels
- ☐ effectifs
- ☐ effaçables
- ☐ variables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie à chaque itération
- ☐ jamais vraie
- ☐ vraie uniquement à la première itération
- ☐ vraie uniquement à la dernière itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ la probabilité que a soit égal à b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le minimum de a et b
- ☐ le produit de a par b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 1 et N
- ☐ 0 et N-1
- ☐ -1 et N+1
- ☐ 0 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés **if-then-else**
- ☐ 1 seul énoncé itératif **for**
- ☐ 2 énoncés itératifs **for**
- ☐ 3 énoncés itératifs **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ toujours vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ parfois vérifié
- ☐ jamais vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
- ☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ C
- ☐ Lisp
- ☐ Java
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ avec le compilateur
- ☐ on ne peut pas
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ en l'exécutant sur l'ordinateur

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une variable
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une personne
- ☐ une constante

Question 5 ♣ En C, un entier peut être de type :

- ☐ très long
- ☐ int
- ☐ double
- ☐ long
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -35
- ☐ 100
- ☐ -1
- ☐ -34



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ impair
- ☐ pair
- ☐ positif
- ☐ négatif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ aucune des deux
- ☐ 125 et 765
- ☐ -3 et 768
- ☐ 126 et 767

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont toujours exacts
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de rendre le programme moins lisible
- ☐ de structurer le programme
- ☐ de localiser les objets
- ☐ de paramétrer les actions
- ☐ de rendre le programme plus complexe
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ formels
- ☐ effaçables
- ☐ variables
- ☐ effectifs

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ jamais vraie
- ☐ vraie à chaque itération
- ☐ vraie uniquement à la première itération
- ☐ vraie uniquement à la dernière itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le minimum de a et b
- ☐ la probabilité que a soit égal à b
- ☐ le produit de a par b
- ☐ le pgcd (plus grand commun diviseur) de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N
- ☐ 1 et N
- ☐ -1 et N+1
- ☐ 0 et N-1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés **if-then-else**
- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ est vérifié si l'ordinateur est en forme
- ☐ parfois vérifié
- ☐ toujours vérifié
- ☐ jamais vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016**Durée 1h30**

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ C
☐ Java
☐ Lisp
☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ avec le compilateur
☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
☐ en l'exécutant sur l'ordinateur
☐ on ne peut pas

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une variable
☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
☐ une personne
☐ une constante

Question 5 ♣ En C, un entier peut être de type :

- ☐ int
☐ long
☐ double
☐ très long
☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -1
☐ -35
☐ 100
☐ -34



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ positif
- ☐ impair
- ☐ négatif
- ☐ pair

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ aucune des deux
- ☐ 125 et 765
- ☐ -3 et 768
- ☐ 126 et 767

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont toujours exacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée » et « résultat »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « formel donnée »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x = 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de paramétrer les actions
- ☐ de localiser les objets
- ☐ de rendre le programme plus complexe
- ☐ de structurer le programme
- ☐ de rendre le programme moins lisible
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ formels
- ☐ effaçables
- ☐ effectifs
- ☐ variables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie à chaque itération
- ☐ vraie uniquement à la dernière itération
- ☐ vraie uniquement à la première itération
- ☐ jamais vraie

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le produit de a par b
- ☐ la probabilité que a soit égal à b
- ☐ le minimum de a et b
- ☐ le pgcd (plus grand commun diviseur) de a et b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N
- ☐ 1 et N
- ☐ -1 et N+1
- ☐ 0 et N-1

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés **if-then-else**
- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés itératifs **for**
- ☐ 1 seul énoncé itératif **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ jamais vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ parfois vérifié
- ☐ toujours vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
- ☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ Lisp
- ☐ C
- ☐ Java
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ en l'exécutant sur l'ordinateur
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents
- ☐ avec le compilateur
- ☐ on ne peut pas

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une constante
- ☐ une personne
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une variable

Question 5 ♣ En C, un entier peut être de type :

- ☐ double
- ☐ int
- ☐ long
- ☐ très long
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -1
- ☐ -35
- ☐ -34
- ☐ 100



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ impair
- ☐ pair
- ☐ négatif
- ☐ positif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ -3 et 768
- ☐ 126 et 767
- ☐ 125 et 765
- ☐ aucune des deux

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ les calculs avec les réels sont toujours exacts
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée » et « résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x \neq 0$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de localiser les objets
- ☐ de rendre le programme moins lisible
- ☐ de paramétrer les actions
- ☐ de structurer le programme
- ☐ de rendre le programme plus complexe
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ être toujours vraies avant l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ de temps en temps vraies avant et après l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ variables
- ☐ effaçables
- ☐ formels
- ☐ effectifs

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie uniquement à la première itération
- ☐ jamais vraie
- ☐ vraie uniquement à la dernière itération
- ☐ vraie à chaque itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ la probabilité que a soit égal à b
- ☐ le minimum de a et b
- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le produit de a par b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ -1 et N+1
- ☐ 1 et N
- ☐ 0 et N-1
- ☐ 0 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 2 énoncés itératifs **for**
- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**
- ☐ 1 seul énoncé itératif **for**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ jamais vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ toujours vérifié
- ☐ parfois vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si x est le minimum de t , et 0 sinon
- ☐ elle renvoie 1 si tous les éléments du tableau t sont égaux à x
- ☐ Aucune de ces réponses n'est correcte.



Info-C

Xidian - Polytech

Examen du 28/10/2016

Durée 1h30

Nom :

Prénom :

ID :

Aucun document n'est autorisé. L'usage du téléphone mobile est interdit.

Les questions faisant apparaître le symbole ♣ peuvent présenter zéro, une ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Question 1

gcc est un compilateur qui permet de traduire un programme C en langage machine :

- ☐ faux
- ☐ vrai

Question 2 ♣ Parmi ces langages, lequel est un langage *procédural* ?

- ☐ C
- ☐ Lisp
- ☐ Java
- ☐ Aucune de ces réponses n'est correcte.

Question 3 Comment peut-on garantir la *validité* de l'exécution d'un programme ?

- ☐ on ne peut pas
- ☐ en l'exécutant sur l'ordinateur
- ☐ avec le compilateur
- ☐ de façon analytique, en vérifiant la validité des antécédents et des conséquents

Question 4 En informatique, est-ce qu'un *type* ?

- ☐ une constante
- ☐ une personne
- ☐ un ensemble fini de valeurs qui possèdent les mêmes caractéristiques
- ☐ une variable

Question 5 ♣ En C, un entier peut être de type :

- ☐ long
- ☐ double
- ☐ très long
- ☐ int
- ☐ Aucune de ces réponses n'est correcte.

Question 6 À quelle valeur décimale correspond la configuration binaire (en complément à 2 sur 8 bits) suivante : 11011101.

- ☐ -34
- ☐ 100
- ☐ -1
- ☐ -35



Question 7 L'expression $(x \& 1) == 1$ est vraie si l'entier x est :

- ☐ pair
- ☐ négatif
- ☐ impair
- ☐ positif

Question 8 À quelle valeur décimale correspondent les valeurs 0176 et 0x2FF ?

- ☐ 125 et 765
- ☐ aucune des deux
- ☐ -3 et 768
- ☐ 126 et 767

Question 9 ♣

Cochez les affirmations suivantes valides au sujet des *réels* informatiques.

- ☐ la soustraction de 2 réels presque identiques peut donner un résultat faux
- ☐ les calculs avec les réels sont approchés ou inexacts
- ☐ les calculs avec les réels sont toujours exacts
- ☐ vérifier l'égalité de 2 réels a et b s'écrit : $|(a - b)| < \epsilon$
- ☐ Aucune de ces réponses n'est correcte.

Question 10 Dans l'appel de fonction `printf("%d\n", x)`, x est :

- ☐ un paramètre « effectif résultat »
- ☐ un paramètre « effectif donnée »
- ☐ un paramètre « formel donnée »
- ☐ un paramètre « effectif donnée » et « résultat »

Question 11 ♣

```
if (x==0) {  
    if (x==1 && y==0)  
        printf("Ok\n");  
    else  
        printf("not Ok\n");  
}
```

Dans le code précédent, le message « not Ok » est affiché quand :

- ☐ $x \neq 0$
- ☐ $x \neq 0$ et $y = 2$
- ☐ $x = 0$ et ($x \neq 1$ ou $y \neq 0$)
- ☐ $x = 0$ et $y = 2$
- ☐ Aucune de ces réponses n'est correcte.

**Question 12 ♣**

Dans un langage procédural, une routine (fonction ou procédure) permet :

- ☐ de structurer le programme
- ☐ de rendre le programme plus complexe
- ☐ de localiser les objets
- ☐ de rendre le programme moins lisible
- ☐ de paramétrer les actions
- ☐ Aucune de ces réponses n'est correcte.

Question 13

L'antécédent $\{P\}$ et le conséquent $\{Q\}$ d'un énoncé **E** sont des affirmations qui doivent :

- ☐ de temps en temps vraies avant et après l'exécution de **E**
- ☐ être toujours vraies, respectivement, avant et après l'exécution de **E**
- ☐ être toujours fausses avant et après l'exécution de **E**
- ☐ être toujours vraies avant l'exécution de **E**

Question 14 Lors d'un *appel* de fonction ou de procédure, les paramètres sont appelés :

- ☐ variables
- ☐ formels
- ☐ effectifs
- ☐ effaçables

Question 15

L'*Invariant* d'un énoncé itératif (*i.e.* **tantque**) est une affirmation :

- ☐ vraie à chaque itération
- ☐ jamais vraie
- ☐ vraie uniquement à la première itération
- ☐ vraie uniquement à la dernière itération

Question 16

```
int mystere(int a, int b) {  
    while (a != b) {  
        if (a > b)  
            a = a - b;  
        else  
            b = b - a;  
    }  
    return a;  
}
```

Que calcule la fonction donnée ci-dessus ?

- ☐ le pgcd (plus grand commun diviseur) de a et b
- ☐ le minimum de a et b
- ☐ la probabilité que a soit égal à b
- ☐ le produit de a par b

**Question 17**

En C, les indices du premier et du dernier élément du tableau `int t[N]` sont *respectivement* égaux à :

- ☐ 0 et N-1
- ☐ 0 et N
- ☐ -1 et N+1
- ☐ 1 et N

Question 18

Pour traiter *tous* les éléments de la matrice `int m[M][N]`, il faut :

- ☐ 1 seul énoncé itératif **for**
- ☐ 3 énoncés itératifs **for**
- ☐ 2 énoncés itératifs **for**
- ☐ 2 énoncés **if-then-else**

Question 19

En C, à l'exécution, l'accès à un élément d'indice i (*i.e.* `t[i]`) est :

- ☐ toujours vérifié
- ☐ est vérifié si l'ordinateur est en forme
- ☐ jamais vérifié
- ☐ parfois vérifié

Question 20 ♣

```
int mystere(int t[N], int x) {  
    int i;  
    for (i=0; i<N; i++) {  
        if (t[i]==x)  
            return 1;  
    }  
    return 0;  
}
```

Que fait la fonction C donnée ci-dessus ?

- ☐ elle ne fait rien et ne renvoie rien
- ☐ elle renvoie 1 si tous les éléments du tableau `t` sont égaux à `x`
- ☐ elle renvoie 1 si $x \in t$ et 0 sinon
- ☐ elle renvoie 1 si `x` est le minimum de `t`, et 0 sinon
- ☐ Aucune de ces réponses n'est correcte.