Université de Xidian
Polytech'Sophia

Université de Xidian
2018–2019

Examen de Langage C *(V. Granet)*

Durée : 1h30
Aucun document autorisé
Mobiles interdits

► 1. `gcc` est un compilateur qui permet de traduire un programme C en langage machine. Cochez une réponse :

- ☐ vrai
- ☐ false

Réponse : vrai

.....

► 2. Sur 8 bits, donnez la représentation binaire en complément à 2 des trois entiers 35, −35 et −1.

35 =
-35 =
-1 =

$$35 = 2^5 + 2^1 + 2^0 = 00100011$$
$$-35 = 11011101$$
$$-1 = 11111111$$

.....

► 3. À quelles valeurs décimales correspondent les deux entiers `0176` et `0x7E` ?

$$0176 = 126$$
$$0x7E = 126$$

.....

► 4. Parmi les égalités suivantes, cochez celle qui est valide :

- ☐ **`non (p ou q) = non p ou non q`**
- ☐ **`non (p ou q) = non p et non q`**
- ☐ **`non (p ou q) = non p ou q`**

Réponse : **`non (p ou q) = non p et non q`**

.....

► 5. Donnez les valeurs de `x` et `y` après l'exécution du code C suivant :


```
x = 10;
y = x--;
// x = ?
// y = ?
```

```
// x = 9
// y = 10
```

.....

► 6. Dans l'appel de fonction `printf("%d\n", x)`, quelle est la nature de `x` ? Cochez la bonne réponse :

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

Réponse : un paramètre « effectif donnée »

.....

► 7. 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`

Réponse : être toujours vraies, respectivement, avant et après l'exécution de `E`

.....

► 8. soit le code C :

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

le message « `not Ok` » est affiché quand (cochez une réponse) :

- ☐ $x \neq 0$
- ☐ $x = 0$ et $y = 2$
- ☐ $x \neq 0$ et $y = 2$

Réponse : lorsque $x = 0$ et $y = 2$

.....

► 9. Dans le code C suivant, quelle est la valeur de `z` si `x=1` et `y=0` ?

```
if (x==0 && y==0) z=0;
else
   if (y!=0) z=1;
   else z=2;
//
// z = ?
//
```

Réponse : $z = 2$

Soit la déclaration de type suivante :

```
enum lesJours {lundi, mardi, mercredi, jeudi, vendredi, samedi, dimanche};
```

▶ 10. Écrivez la procédure `ecrireNomDuJour` qui écrit en toutes lettres sur la sortie standard le nom du jour de la semaine, de type `lesJours`, passé en paramètre.

```
/*
 * Antécédent : j ∈ lesJours
 * Rôle : écrit le nom du jour sur la sortie standard
 */
void ecrireNomDuJour(enum lesJours j) {
  switch (j) {
  case lundi : printf("lundi\n"); break;
  case mardi : printf("mardi\n"); break;
  case mercredi : printf("mercredi\n"); break;
  case jeudi : printf("jeudi\n"); break;
  case vendredi : printf("vendredi\n"); break;
  case samedi : printf("samedi\n"); break;
  case dimanche : printf("dimanche\n"); break;
  }
}
```

▶ 11. Écrivez la fonction `lireJour` qui lit sur l'entrée standard un entier ($\in [0;6]$) et qui renvoie la valeur correspondante dans le type `lesJours`. Vous vérifierez la validité de l'entier lu, en cas d'erreur vous écrirez un message sur la sortie d'erreur standard, et terminerez le programme.

```
/*
 * Rôle : lit un entier sur la sortie standard et renvoie
 *        sa valeur correspondante dans le type lesJours
 */
enum lesJours lireJour(void) {
  int j;
  scanf("%d", &j);
  if (j<0 || j>6) {
    fprintf(stderr, "%d : numéro du jour invalide\n", j);
    exit(EXIT_FAILURE);
  }
  return j;
}
```

▶ 12. Écrivez la fonction `main` dans laquelle vous déclarerez une variable `jour` de type `lesJours` et à laquelle vous affecterez la valeur `samedi`.

```
int main(void) {
  enum lesJours jour = samedi;
  return EXIT_SUCCESS;
}
```

▶ 13. Complétez la fonction `main` pour lire sur l'entrée standard un entier et afficher sur la sortie le nom du jour correspondant. Vous utiliserez les routines `lireJour` et `ecrireNomDuJour` précédentes.


```
ecrireNomDuJour(lireJour());
```