

Contrôle de Info C

Durée : 1h30

Aucun document autorisé

Nom :

Prénom :

- 1. Dans la fonction suivante, insérez les affirmations qui donnent sa sémantique et qui prouvent la validité de l'énoncé itératif.

*// Antécédent: ...**// Conséquent: ...*

```
int mystere(int n) {
```

```
    int i=0, j=1, s=0;
```

// Invariant :

```
    while(i<n) {
```

*// ...**// ...*

```
        s = s + j;
```

*// ...**// ...*

```
        j = j + 2;
```

*// ...**// ...*

```
        i = i + 1;
```

// ...

```
    }
```

// ...

```
    return s;
```

```
}
```

1. choisir $[a, b]$ qui contient a priori un zéro ;
2. choisir le milieu m de l'intervalle $[a, b]$;
3. si $f(m) = 0$ alors on a trouvé un zéro
4. si $f(a)f(m) < 0$ alors définir le nouvel intervalle $[a, m]$;
5. si $f(b)f(m) < 0$ alors définir le nouvel intervalle $[m, b]$;
6. recommencer à partir de 2.

-
- This image shows a single sheet of white paper with horizontal blue or grey ruling lines. The lines are evenly spaced and run across the width of the page. There are approximately 20 lines visible. The paper has a slight shadow on the right side, suggesting it's resting on a surface.

- 123 +34 -230 77 +10000 -255

Vous devrez gérer les dépassements de capacité. Les constantes `INT_MAX` et `INT_MIN` du fichier `limits.h` définissent les bornes de l'intervalle du type `int`. Si le nombre n'appartient pas à l'intervalle, vous signalerez l'erreur par un message approprié sur la sortie d'erreur standard.

[illegible]

► 4. Écrivez un programme qui lit toute l'entrée standard qui écrit sur la sortie standard le nombre de sous-suites, appelées motifs, bâtis sur le modèle précédent. Pensez à programmer un automate fini. Par exemple, votre programme écrira 3, si l'entrée standard contient :

les motifs reconnus sont en rouge ci-dessous :

[illegible]

[illegible]

5. À l'aide d'un `define`, définissez la constante `NB_BITS` égale au nombre d'éléments du tableau nécessaires à la représentation d'un entier non signé.

-

- ```
unsigned int valeurDecimale(const short bits[])
```

This image shows a single sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins or other markings on the paper.

- ```
void decalageDroit(short bits[], const int n)
```

This image shows a single sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.