

J1MI2013: Algorithmes et Programmes

Test 1

15 février 2016 - Durée 15 minutes - Aucun document autorisé

Nom :

Prénom :

Exercice 1. On appelle *logarithme entier* (à base 2) de n (n étant un entier strictement positif) le plus petit entier k tel que l'on ait $2^k \geq n$. Par exemple, le logarithme entier à base 2 de 23 est 5 car on a $2^4 < 23 \leq 2^5$.

Écrire une fonction `logEntier(int n)` qui retourne le logarithme entier à base 2 de n .

Exercice 2. Écrire une fonction `somme(int n)` qui affiche la somme des carrés des entiers positifs impairs inférieurs ou égaux à n (par exemple, l'appel `somme(6)` devrait afficher 35, car $1^2 + 3^2 + 5^2 = 35$).

Exercice 3. On considère la fonction `mystere` suivante :

```
int mystere(int n, int k){
    int s = 0;
    int i = k;
    while (s + k/i < n && i > 0){
        s = s + k/i;
        i = i-1;
    }
    return s;
}
```

Exécuter la fonction pas à pas (donner, pour chaque passage dans la boucle `while`, les valeurs des variables `s` et `i` à la fin du corps de boucle, soit après l'exécution de l'instruction `i = i-1`), et donner le résultat retourné, pour l'appel `mystere(4,3)`. Faire de même pour l'appel `mystere(6,3)`