

IF1 : interrogation

Groupe M3, durée 45 min

Le 27 novembre 2008

Exercice 1

Écrire une fonction `cube` qui prend un entier `n` en argument et qui renvoie la somme des `n` premiers cubes si `n` ≥ 1 , et 0 sinon. Ainsi, `cube(5)` vaudra 225 (soit $1^3 + 2^3 + 3^3 + 4^3 + 5^3$) tandis que `cube(-5)` vaut 0.

Exercice 2

Écrire un programme qui demande un entier `n` à l'utilisateur, puis lui demande `n` fois d'entrer un entier ainsi qu'une position `i` $\in \{1, \dots, n\}$ (pour l'achage), et qui affiche ensuite les entiers selon l'ordre spécifié. Ainsi, si l'utilisateur entre

```
4
10
3
12
2
15
4
16
1
```

(la première ligne correspondant au nombre d'entiers à entrer, puis les lignes paires sont les entiers eux-mêmes et les lignes impaires leur position), il souhaite afficher l'entier 10 en position 3, l'entier 12 en position 2, 15 en position 4 et 16 en première position. Le programme devra alors afficher :
16 12 10 15.

Exercice 3

Un entier `n` est dit *parfait* s'il est égal à la somme de ses diviseurs propres (c'est-à-dire autres que lui-même). Par exemple, 6 est parfait car ses diviseurs propres sont 1, 2 et 3, et on a $6 = 1 + 2 + 3$.

1. Écrire une fonction `diviseur` prenant en entrée deux entiers `a` et `b` et qui renvoie Vrai si `a` divise `b`, Faux sinon.
2. Écrire une fonction `parfait` qui prend en entrée un entier `n` et qui renvoie Vrai si `n` est parfait, Faux sinon.
3. Écrire un programme qui demande un entier `n` à l'utilisateur, et qui affiche d'abord tous les nombres parfaits $\leq n$ pairs, puis tous les nombres parfaits $\leq n$ impairs.