



UFR SCIENCES ET TECHNOLOGIES
INFORMATIQUE

L2I LICENCE I
TD Algorithmique I
Fiche de TD N° 2
Les Structures répétitives

Exercice 1 : cube d'un nombre

Ecrire un algorithme qui calcule le cube d'un nombre plusieurs fois. Le programme s'arrête quand l'utilisateur saisie 0.

Exercice 2 : Nombres pairs

Ecrire une fonction ou procédure qui permet d'entrer deux valeurs M et N et d'afficher toutes les valeurs paires entre M et N si $M < N$.

Exercice 3 : Multiplication par additions successives

1. Ecrire un algorithme qui calcule le produit de deux nombres entiers positifs par additions successives du multiplicande autant de fois que nécessaire.

$$a \times b = \underbrace{a + a + a + \dots + a}_{b \text{ fois}}$$

Les données à saisir seront validées par une structure répétitive.

2. Écrire un programme qui calcule le produit de n nombres entiers positifs pas additions successives selon le principe suivant :

$$\underbrace{a_1 \times a_2}_{P_1} \times a_3 \times a_4 \times \dots \times a_n$$

$$\underbrace{P_1 \times a_3}_{P_2}$$

\vdots

$$\underbrace{P_{n-2} \times a_n}_{P_{n-1}}$$

Le produit de deux nombres est calculé par additions successives comme dans la question 1. On veillera à valider les données saisies par des structures répétitives.

Exercice 4 : Nombres parfaits et nombres amicaux

1. Définir un algorithme qui détermine si un entier N est parfait ou non.
Un entier est dit parfait s'il est égal à la somme de ses diviseurs sauf lui-même.
Exemple : 6 est un nombre parfait car on a : $6 = 1 + 2 + 3$.

2. Deux nombres entiers n et m sont dits amicaux si la somme des diviseurs de n (n non compris) vaut m et la somme des diviseurs de m (m non compris) vaut n .
Exemple : 220 et 284.

Donner un algorithme qui détermine si deux nombres sont amicaux.

Exercice 5 : Partie entière d'un nombre

Ecrire un algorithme qui calcule la partie entière d'un nombre positif N saisi par un utilisateur.

Exercice 6: Manipulation d'entiers

Soit N un entier donné.

1. Donner un algorithme qui détermine le nombre de chiffres composant l'entier N .
Donner l'organigramme correspondant.
2. Donner un algorithme qui calcule la somme des chiffres composant l'entier N .

Exercice 7: PGCD de deux nombres

On veut écrire un algorithme qui calcule le **PGCD** de deux entiers strictement positifs lus au clavier en utilisant l'algorithme d'Euclide qui est le suivant :

1. **Premier pas** : si n et m sont deux entiers, on peut se ramener au cas où ils sont positifs en prenant leurs valeurs absolues, ce que nous faisons grâce à la fonction *abs*.
2. **Deuxième pas** : si $m = 0$, le **PGCD** est évidemment n . Sinon, nous effectuons la division euclidienne de n par m . Si on écrit $n = m \cdot q + r$, on voit que tout $\text{pgcd}(n, m) = \text{pgcd}(m, r)$. Comme on diminue strictement la valeur de n d'une étape à la suivante, on est sûr d'arriver au bout d'un nombre fini d'opérations au cas où $m = 0$. Le **PGCD** est alors la valeur de n .

Exercice 8 : Affichage

Écrire un programme qui affiche :

```
*
* *
* * *
* * * *
* * * * *
* * * *
* * *
* *
*
```

Exercice 9 : Table de multiplication

Écrire un programme qui permet de faire tenir les tables de multiplication, par 1, 2, ..., 10 sur l'écran.

Les premières lignes doivent être par exemple :

```
1 * 1 = 1 | 2 * 1 = 2 | 3 * 1 = 3 | 4 * 1 = 4 | 5 * 1 = 5
1 * 2 = 2 | 2 * 2 = 4 | 3 * 2 = 6 | 4 * 2 = 8 | 5 * 2 = 10
..... | ..... | ..... | ..... | .....
```

NB : Pour nous protéger contre le Coronavirus (Covid-19) respectons les mesures barrières et la distanciation physique.