

TD3 : les structures répétitives

Exercice 1 :

Écrire un algorithme qui calcul la somme des N premiers nombres entiers.

Exercice 2 :

Écrire un algorithme qui recherche le minimum et le maximum dans un ensemble de N nombres.

Exercice 3 :

Écrire un algorithme qui calcul le quotient et le reste de la division de deux entiers A et B sans utiliser l'opération de division.

Exercice 4 :

Écrire un algorithme qui détermine tous les diviseurs d'un entier X donné.

Exercice 5 :

Écrire un algorithme qui détermine si un nombre entier X est premier ou non.

Exercice 6 :

Écrire un algorithme qui calcule la somme des chiffres qui composent un entier naturel N .

Exercice 7 :

Écrire un algorithme qui permet à l'utilisateur de saisir un texte caractère par caractère ; le texte se termine par un point '.', et qui affiche à la fin le nombre d'apparition de la lettre 'a'.

Exercice 8 :

Écrire un algorithme permettant de calculer la valeur de l'expression E , telle que $E = (1 + 2) \times (1 + 2 + 3) \times (1 + 2 + 3 + 4) \times \dots \times (1 + 2 + 3 + \dots + (N - 2) + (N - 1) + N)$, et ($N \geq 2$).

Exercice 9 :

Écrire un algorithme qui affiche un carré de taille ($N \times N$) rempli du caractère '*'. Exemple de sortie pour ($N = 5$) :

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

Exercices (Devoirs non corrigés) :

10. Écrire un algorithme qui affiche un carré de taille ($N \times N$) rempli du caractère '*'. Exemple de sortie pour ($N = 5$) :

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

11. Écrire un algorithme qui génère et affiche une matrice identité de taille ($N \times N$). Exemple de sortie pour ($N = 4$) :

```
1 0 0 0
0 1 0 0
0 0 1 0
0 0 0 1
```

12. Écrire un algorithme qui affiche les (N) premières lignes du triangle de Pascal en utilisant uniquement les boucles. Exemple de sortie pour (N = 5) :

```
1
1 1
1 2 1
1 3 3 1
1 4 6 4 1
```

13. Ecrire un programme C qui permet de lire un entier entre 10 et 20 et d'afficher la somme harmonique :

$$S = 1 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{n}$$

14. Soit la suite U_n définie par :

$$\begin{cases} U_0 = 1 \\ U_{n+1} = 5U_n + 3 \end{cases}$$

Ecrire un algorithme qui permet de lire n et de calculer la suite U_n pour un rang n .