

TD 3 : Les boucles

Exercice 1

1. Écrivez un algorithme qui demande à l'utilisateur un nombre compris entre 1 et 20 jusqu'à ce que la réponse convienne.
2. Écrivez un algorithme qui demande un nombre compris entre 10 et 20, jusqu'à ce que la réponse convienne. En cas de réponse supérieure à 20, on fera apparaître un message : "Plus petit !", et inversement, "Plus grand !" si le nombre est inférieur à 10.
3. Écrivez un algorithme qui calcule et affiche la somme des N premiers entiers positifs (N est à saisir par l'utilisateur).
4. Écrivez un algorithme qui demande un nombre de départ, et qui ensuite affiche les dix nombres suivants. Par exemple, si l'utilisateur entre le nombre 17, le programme affichera les nombres de 18 à 27.

Exercice 2

1. Écrivez un algorithme qui récupère un entier et qui affiche sa table de multiplication.
2. Écrivez un algorithme qui récupère un entier et qui affiche son factoriel.
3. Écrivez un algorithme qui récupère 20 entiers et qui affiche le plus grand d'entre eux.
4. Écrivez un algorithme qui affiche le plus grand nombre saisi par l'utilisateur (on ne connaît pas combien de nombres seront saisis par l'utilisateur).

Exercice 3

1. Écrivez un algorithme qui récupère un entier saisi par l'utilisateur et qui affiche si cet entier est premier ou pas.

2. Écrivez un algorithme qui récupère un entier n saisi par l'utilisateur et qui affiche tous les entiers premiers inférieurs à n.
3. Écrivez un algorithme qui affiche les diviseurs d'un entier saisi par l'utilisateur. Avant la fin de son exécution, l'algorithme doit afficher le nombre de diviseurs qu'il a trouvé.
4. On peut dire que deux entiers m et n sont **amis** si la somme des diviseurs de m est égale à n et si la somme des diviseurs de n est égale à m (sans prendre en compte les diviseurs 1 et le nombre lui-même). Par exemple, 48 et 75 sont des amis parce que :
 - $2 + 3 + 4 + 6 + 8 + 12 + 16 + 24 = 75$ (les diviseurs de 48)
 - $3 + 5 + 15 + 25 = 48$ (les diviseurs de 75)

Écrivez un algorithme qui récupère deux entiers et qui nous affiche s'ils sont amis ou pas.

5. Écrivez un algorithme qui récupère un entier n et qui calcule la somme suivante :

$$S = 1^2 + 2^2 + 3^2 + \dots + n^2$$

Exercice 4

Écrivez un algorithme qui affiche "Bonjour" à condition que l'utilisateur le mot de passe correcte (P@\$\$w0Rd). Le nombre de tentatives est limité à 3. Si les tentatives sont gaspillées, l'utilisateur devra répondre à une question secrète (Nebuchadnezzar).

Exercice 5

1. Écrivez un algorithme qui lit la taille n d'un tableau T , il saisit les n éléments du tableau T , il effectue la somme des n éléments du tableau et il affiche cette somme.
2. Écrivez un algorithme qui lit la taille n de deux tableaux $T1$ et $T2$, il effectue la lecture de ces deux tableaux, ensuite il effectue la somme des tableaux $T1$ et $T2$ dans un tableau T et il affiche le tableau T .
3. Écrivez un algorithme qui lit la taille n d'un tableau T , il saisit les n éléments du tableau T et affiche le plus grand et le plus petit élément du tableau.
4. Écrivez un algorithme qui récupère les notes d'une classe. Par la suite, l'algorithme doit afficher les notes supérieures à la moyenne.

5. Refaire les exercices en utilisant des tableaux dynamiques.

Exercice 6

1. Écrivez un algorithme qui récupère deux matrices et qui affiche leurs sommes.
2. Écrivez un algorithme qui récupère deux matrices et qui affiche leurs produits.
3. Écrivez un algorithme qui récupère une matrice et qui affiche sa transposée.
4. Écrivez un algorithme qui récupère une matrice et qui affiche sa trace ($Trace(A) = \sum_{i=1}^n (a_{i,i})$).
 - A : une matrice carrée
 - n : la dimension
 - a : un élément de la matrice