

## Série N°3 : Les structures répétitives

### Exercice 1

Ecrire un algorithme (et un programme) qui affiche les entiers de 1 à n en utilisant :

- une boucle **for**
- une boucle **while**
- une boucle **do-while**

### Exercice 2

1. Ecrire un programme qui demande un nombre de départ, et qui calcule la somme des entiers jusqu'à ce nombre. Par exemple, si l'on entre 5, le programme doit calculer :  
 $1 + 2 + 3 + 4 + 5 = 15$
2. Ecrire un programme qui demande un nombre de départ, et qui calcule sa factorielle. la factorielle de 8, notée 8 !, vaut  $1 \times 2 \times 3 \times 4 \times 5 \times 6 \times 7 \times 8$

NB : on souhaite afficher uniquement le résultat, pas la décomposition du calcul.

### Exercice 3

Ecrire un programme 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 4

Ecrire un programme 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

### Exercice 5

Ecrire un programme qui demande successivement 20 nombres à l'utilisateur, et qui lui dise ensuite quel était le plus grand parmi ces 20 nombres :

Entrez le nombre numéro 1 : 12

Entrez le nombre numéro 2 : 14

etc.

Entrez le nombre numéro 20 : 6

Le plus grand de ces nombres est : 14

Modifiez ensuite l'algorithme pour que le programme affiche de surcroît en quelle position avait été saisie ce nombre :

C'était le nombre numéro 2

### Exercice 6

Ecrire un programme qui affiche à l'écran les 20 premiers termes de la table de multiplication par 7, mais qui n'affiche pas ceux qui sont un multiple de 3.

### Exercice 7

Un nombre différent de 1 est premier lorsque qu'il n'est divisible que par 1 et par lui-même. Ecrire un algorithme qui lit un nombre et détermine s'il est premier ou non.

### Exercice 8

Lire un nombre. S'il est pair, afficher tous les nombres pairs qui lui sont inférieurs, sauf le zéro.

