

En algorithmique, on est amené à répéter un certain nombre de fois, la même suite d'opérations. tant qu'une certaine condition n'est pas vérifiée. On utilise la boucle **while**. Il faut avoir recours à une condition d'arrêt.

Elle prend la forme :

```
while condition :
    Instruction 1
    Instruction 2
```

Un exemple de script

```
r=float(input('r:'))
n=int(input('n:'))
while (r>=n):
    r=r-n
print(r)
```

**Exercice 1.** Écrire une suite d'instructions qui demande à l'utilisateur un nombre inférieur ou égal à 100, et, en cas de réponse incorrecte, affiche "enter un nombre inférieur à 100" jusqu'à ce que la réponse soit bonne.

**Exercice 2.** Créer un programme avec la boucle **while** calculant et affichant les termes de la suite  $u$ , définie par  $\begin{cases} u_0 = 6 \\ u_{n+1} = 2u_n - 5 \end{cases}$  jusqu'à l'indice  $n = 15$ .

**Exercice 3.** 1. Entrer dans l'éditeur les instructions suivantes :

```
n=int(input('Donner une valeur de n :'))
S=0
k=1
while k<=n:
    S=S+k
    k=k+1
print(S)
```

2. Tester avec différentes valeurs pour  $n$ .

3. À quoi correspond la valeur de  $S$  à la fin de la boucle? Vérifier à l'aide d'une formule sur les sommes usuelles.

4. Calculer les sommes suivantes à l'aide d'un boucle **while**.

$$\star \sum_{k=1}^{10} k^2$$

$$\star \sum_{k=1}^{10} k^3$$

**Exercice 4.** Soit  $a > 0$ , et,  $\forall n \in \mathbb{N}^*$ ,  $S_n = \sum_{k=1}^n \frac{1}{k}$ .

Créer un programme qui renvoie la plus petite valeur de  $n$  telle que l'on ait  $S_n > a$ .

### Opérations sur les listes :

**Exercice 5.** Créer une liste de  $n$  éléments, où  $n$  est un entier choisi par l'utilisateur et les éléments de la liste sont entrés les uns après les autres.

**Exercice 6.** Créer une liste contenant 10 nombres flottants et créer un script qui permet de trouver les deux nombres les plus proches.