

# DM 2nde n°5, algorithmes, boucle *while*. A rendre par binôme.

## 1 Boucles non bornées

Voici ce qu'il faut savoir à ce sujet :

**MÉMO**

► Pour écrire certains programmes, il est parfois nécessaire de répéter une ou plusieurs instructions un nombre inconnu de fois. Lorsque le nombre de répétitions n'est pas connu à l'avance, on utilise une **boucle non bornée** qui est parcourue jusqu'à ce qu'une certaine condition ne soit plus vérifiée. Tant que cette condition est vérifiée, la boucle continue.

► La syntaxe d'une boucle non bornée en langage naturel et en Python

Langage naturel	Langage Python
Tant que condition faire instruction(s)	while condition : instruction(s)

► Il n'existe pas d'instruction pour définir la fin de la boucle. C'est l'**indentation**, c'est-à-dire le décalage vers la droite d'une ou plusieurs lignes, qui permet de marquer la fin de la boucle.

**Exemple :** On veut déterminer le plus petit entier naturel  $n$  tel que la somme de tous les nombres entiers de 0 à  $n$  soit strictement supérieure à 1 000. Pour cela, on écrit le script ci-contre.

- Tant que la condition  $\text{somme} \leq 1\,000$  reste vraie, le script ajoute  $n$  à la somme. Dès que la variable  $\text{somme}$  devient supérieure à 1 000, la boucle s'arrête.
- La valeur de la variable  $n$  à la fin de l'exécution du script est 45. Cette variable  $n$  compte le nombre de fois où la boucle a été réalisée : c'est une **variable compteur**.

```
1 n = 0
2 somme = 0
3 while somme <= 1000 :
4     n = n + 1
5     somme = somme + n
```

## 2 Exercices

*Exercice 1.* Un jardinier souhaite creuser un puits dans son jardin. Le prix du forage dépend de la profondeur du puits. Le premier mètre coûte 100 euros, le second mètre coûte 20 euros, le troisième 40 euros et ainsi de suite en ajoutant 20 euros tous les mètres

1. Quel est le coût du forage d'un puits de dix mètres de profondeur ?
2. Le budget alloué à ce projet ne doit pas dépasser 700 euros ? Le jardinier veut connaître la profondeur maximale qu'il peut atteindre avec une telle somme. Il écrit le script suivant :

```
profondeur=1
prix=100
while prix<=700 :
    prix=prix+20
    profondeur=profondeur+1
```

- (a) Quelle est la valeur de la variable *profondeur* après l'exécution du script ?
- (b) Quelle profondeur maximale pourra atteindre le jardinier avec son budget ?

*Exercice 2.* Nous lançons deux dés à six faces parfaitement équilibrées et nous additionnons les deux résultats obtenus. Le script suivant modélise les lancers de ces deux dés et simule le nombre *n* de lancers qu'il a fallu pour obtenir la somme 12. Copier et compléter ce script.

```
from random import randint
s=???
n=????
while s ???? :
    de1=randint(1,6)
    de2=randint(1,6)
    s=de1+de2
    n=n+1
```

*Exercice 3.* Considérons une piste de course comprenant six cases. Un lièvre et une tortue souhaitent faire une course selon les règles suivantes : nous lançons un dé à six faces, parfaitement équilibré.

- Lorsque le résultat du dé est 6 le lièvre franchit directement la ligne d'arrivée et gagne.
  - lorsque le résultat du dé est un chiffre compris entre 1 et 5, la tortue avance d'une case.
- Pour que le lièvre gagne, il suffit donc que le 6 sorte avant que la tortue ait parcourue les six cases. Pour que la tortue gagne, le 6 ne doit jamais sortir au cours des 6 premiers lancers.

Le but de cet exercice est de déterminer qui, de la tortue ou du lièvre, gagne le plus souvent.

### Partie 1 :

1. Compléter le script pour qu'il permette de réaliser une partie et affiche la séquence de résultat obtenus avec le dé.

```
from random import randint
de=???
print(de)
lancer=1
while lancer<6 and ??? :
    de=????
    print(de)
    lancer=lancer+1
if de == ? :
    print('lièvre gagne')
else :
    print('tortue gagne')
```

2. Peut-on utiliser une boucle *for... in range(6)* pour traduire la réalisation d'une partie ?
3. Quel est le rôle de l'instruction conditionnelle *if* dans le script ci-dessus.
4. Ecrire le script dans un éditeur Python. Réaliser dix parties et compléter le tableau ci-dessous.

Partie	1	2	3	4	5	6	7	8	9	10
Résultat des dés										
Gagnant										

5. Avec le nombre de partie jouées, peut-on conjecturer le gagnant le plus fréquent ?

**Partie 2 :**

Nous souhaitons créer un script qui renvoie les fréquences des victoires du lièvre et de la tortue.

1. Compléter le script ci-dessous

```
from random import randint
def partie(n) :
    L,T=0,0
    for i in range (???,????) :
        de=0
        case=0
        while != ?? and case != ?? :
            de=???
            if de! = 6 :
                case= ?
            if case ==6 :
                T= ??
            else :
                L= ??
    return( ?,?)
```

2. Quel est le rôle des variables  $L$  et  $T$  ?
3. Ecrire ce script dans un éditeur Python et compléter le tableau ci-dessous.

Nombre de parties	10	100	500	1000	5000	10000
Fréquence victoire tortue						
Fréquence victoire lièvre						

4. Pouvons-nous conjecturer le gagnant le plus fréquent ?

**Partie 3 (bonus) :**

1. Nous lançons un dé à six faces parfaitement équilibré.
  - (a) Quelle est la probabilité d'obtenir la face 6 ?
  - (b) Quelle est la probabilité d'obtenir la face 1, 2, 3, 4 ou 5 ?
2. Proposer un arbre pondéré qui décrit la course entre la tortue et le lièvre.
3. A l'aide de cet arbre, déterminer la probabilité que la tortue gagne, puis la probabilité que le lièvre gagne. Conclure.