
Algorithmique et Programmation 1 – TD - TP 3

BOUCLES

Exercice 1 - Premières boucles **while**

Indiquez ce que les programmes suivants affichent :

1. Suite d'instruction 1 :

```
i = 1
while i <= 5 :
    print(i)
    i = i+1
```

2. Suite d'instructions 2 :

```
i = 1
while i <= 5 :
    i = i+1
    print(i)
```

3. Suite d'instructions 3 :

```
i = 1
while i > 0 :
    print(i)
    i = i + 1
```

4. Suite d'instructions 4 :

```
i = 1
while i > 10 :
    print(i)
    i = i + 1
```

Exercice 2 - Premières boucles **for**

Indiquez ce que les programmes suivants affichent :

1. Suite d'instruction 1 :

```
for i in range(50, 100, 10) :
    print(i)
```

2. Suite d'instructions 2 :

```
for i in range(50, 100, 10) :
    print(10)
```

3. Suite d'instructions 3 :

```
x = 0
for i in range(0, 5, 1) :
    x = x + 5
print(x)
```

4. Suite d'instructions 4 :

```
x = 0
for i in range(5) :
    x = x + 5
    print(x)
```

Exercice 3 - Table de simulation

Ecrire les tables de simulation des deux programmes suivants, et en déduire ce qui est affiché en sortie.

1. Programme 1 :

```
n = 6
i = 1
compte = 0

while i <= n :
    if n % i == 0 :
        compte = compte + 1
    i = i + 1

print(compte)
```

2. Programme 2 :

```
n = int(input("Donnez un entier : "))

compte = 0

while n % 2 == 0 :
    compte = compte + 1
    n = n / 2

print(compte)
```

Exercice 4 - Comprendre et modifier un programme

Soit le programme suivant :

```
s = 0

for i in range(0, 101, 2) :
    s = s + i * i

print(s)
```

1. Expliquez ce que fait ce programme
2. Modifiez ce programme de façon à ce qu'il calcule la somme des cubes de 0 à 42

Elément de cours : utiliser sep et end dans la fonction print

Il est possible de modifier le **séparateur** et la **fin de ligne** dans la commande print. Quelques exemples :

- `print(3, 5)` affiche 3 5
- `print(3, 5, sep="+++")` affiche 3+++5
- `print(3, 5, sep="-", end="@")` affiche 3-5@ sans retour à la ligne ensuite

Exercice 5 - Comprendre et modifier un programme (2)

Soit le programme suivant :

```
nombre_notes = 5

note_min = 20
note_max = 0
indice_min = 0
indice_max = 0

for i in range(1, nombre_notes + 1) :
    print("Veuillez entrer la note numéro", i)
    note_saisie=float(input())

    while note_saisie > 20 or note_saisie < 0 :
        print("Attention! une note est comprise entre 0 et 20")
        print("Veuillez entrer la note numéro", i)
        note_saisie=float(input())

    if note_min > note_saisie :
        note_min = note_saisie
        indice_min = i
    if note_max < note_saisie :
        note_max = note_saisie
        indice_max = i

print(nombre_notes, note_min, indice_min, note_max, indice_max)
```

1. Expliquez ce que fait ce programme
2. Recopiez-le dans un fichier `notes.py`, et vérifiez que votre réponse précédente est correcte
3. Modifiez le `print` final afin que l'affichage des résultats soient plus clairs
4. Ajoutez le calcul de la moyenne des notes
5. Modifiez à présent ce programme de façon à ce qu'on ne connaisse pas le nombre de notes à saisir par l'utilisateur, mais que la saisie s'interrompe dès que l'utilisateur tape une note négative ou une note strictement supérieure à 20.

Exercice 6 - Comprendre et modifier un programme (3)

Soit le programme suivant :

```
hauteur = int(input("Entrez un entier : "))

for i in range(1, hauteur + 1) :
    print("*" * i)
```

1. Expliquez ce que fait ce programme
2. Recopiez-le dans un fichier `triangle.py`, et vérifiez que votre réponse précédente est correcte
3. Modifiez ce programme de façon à afficher un triangle isocèle. Exemple de déroulé :

```
Entrez un entier : 6
      *
     **
    ***
   ****
  *****
 *****
*****
```

Exercice 7 - Somme

Ecrire un programme qui fait la somme des entiers tapés au clavier par l'utilisateur, et ce tant qu'il ne tape pas 0. Exemple de déroulé :

```
Donnez le premier entier : 23
Donnez l'entier suivant : 5
Donnez l'entier suivant : -6
Donnez l'entier suivant : 2
Donnez l'entier suivant : 0
La somme des entiers saisis est 24
```

Exercice 8 - Le chiffre 42

Ecrire un programme qui lit 5 entiers tapés au clavier par l'utilisateur, puis affiche "Gagné" si le chiffre 42 se trouvait parmi ceux-là, "Perdu" sinon. Attention, il faut lire les 5 entiers, puis ensuite seulement afficher le résultat. Exemple de déroulé :

```
Veillez entrer l'entier numéro 1 : 65
Veillez entrer l'entier numéro 2 : 42
Veillez entrer l'entier numéro 3 : 32
Veillez entrer l'entier numéro 4 : 8
Veillez entrer l'entier numéro 5 : 74
Gagné
```

Modifiez ensuite votre programme pour qu'il affiche "Gagné" dès que le chiffre 42 est entré par l'utilisateur, "Perdu" si 5 chiffres différents de 42 ont été saisis. Exemple de déroulé :

```
Veillez entrer l'entier numéro 1 : 65
Veillez entrer l'entier numéro 2 : 42
Gagné
```

Exercice 9 - Suite

Ecrire un programme qui demande à l'utilisateur un entier u_0 et un entier $term$ et qui affiche les $term$ premiers termes de la suite définie comme suit :

$$\begin{cases} u_{n+1} = 3u_n + 1 & \text{si } u_n \text{ est impair} \\ u_{n+1} = u_n/2 & \text{si } u_n \text{ est pair} \end{cases}$$

Exemple de déroulé :

```
Donnez le premier terme u_0 de la suite : 6
Donnez le nombre de termes de la suite que vous souhaitez calculer: 4
Le terme 1 de la suite est égal à 3.0
Le terme 2 de la suite est égal à 10.0
Le terme 3 de la suite est égal à 5.0
Le terme 4 de la suite est égal à 16.0
```

Exercice 10 - Le blé du joueur d'échec

On raconte qu'il y a environ 14 siècles, un Prince de Perse voulait récompenser l'inventeur du jeu des échecs. Ce dernier lui demanda le cadeau suivant : 1 grain de blé sur la première case de l'échiquier, 2 sur la deuxième, 4 sur la troisième et ainsi de suite en doublant jusqu'à la 64ème case.

Sachant qu'un grain de blé pèse en moyenne un quart de gramme, combien de temps aurait-il fallu pour satisfaire le vœu de l'inventeur au rythme actuel de la production mondiale de blé qui est d'environ 758 millions de tonnes par an (en 2018) ?