Evaluation C Corrige T6 1 Les boucles FOR

Thème 6 - Langages et programmation

Evaluation Les Boucles FOR en Python -**Evaluation**

Devoir à faire avec des points bonus sur les derniers exercices.

N'oublier pas d'enregistrer régulièrement votre travail. A la fin penser à rendre votre travail. Bon courage à vous.

Exercice 1: La punition

Ecrire un programme qui demande à l'utilisateur un message et le nombre de foisqu'il faut afficher ce message.

```
& Script Python
msg=input('Donner le texte à repéter : ')
nb_de_fois=int(input("donner le nombre de fois qu'il faut répéter le texte : " ))
for k in range(nb_de_fois):
    print(msg)
```

Exercice 2 : Ecrire un programme qui calcule et affiche le produit 87x52 en n'effectuant que des additions.

```
% Script Python
resultat=0
for k in range(87):
    resultat+=52
print(resultat)
```

Exercice 3 : Ecrire un programme qui demande à l'utilisateur de saisir un nombre entier, puis qui affiche la table de multiplication correspondande. Quel table voulez-vous afficher ? 7 7 x 1 = 7 7 x 2 = 14 7 x 3 = 21 ...

```
& Script Python
nb=int(input("Quelle table de multiplication voulez-vous afficher ? )"))
print(f"Table de {nb} : ")
for k in range(11):
    print(f''\{nb\} * \{k\} = \{nb*k\}'')
```

Exercice 4 : Ecrire un programme qui affiche tous les entiers impairs de 0 à 187 inclus.

```
% Script Python
```

```
for k in range(1,188,2):
    print(k, end=' ')
```

Exercice 5 : Ecrire un programme qui affiche tous les carrés des nombres compris entre 0 et 29.

```
for k in range(30):
    print(k**2, end=' ')
```

Exercice 6:

1) Ecrire un programme qui affiche un rectangle de 5 lignes et 10 colonnes avec le caractère # comme ci-dessous :

###########

###########

###########

###########

###########

2) Améliorer ce programme pour qu'il demande d'abord à l'utilisateur de saisir la largeur et la hauteur du rectangle. On peut aussi lui demander avec quel caractère il souhaite remplir le rectangle.

```
# 1ère question :
for ligne in range(5):
   for colonne in range(10):
        print("#",end='')
    print("")
```

```
# 2ème question
largeur = int(input("Veuillez saisir la largeur du rectangle : "))
hauteur = int(input("Veuillez saisir la hauteur du rectangle : "))
caractère = input("Veuillez saisir le caractère de remplissage : ")
for ligne in range(hauteur):
    for colonne in range(largeur):
        print(caractère, end='')
    print(''')
```

Exercice 7:

- 1) Ecrire un programme qui calcule la moyenne de 7 notes.
- 2) Améliorer ce programme pour qu'il demande d'abord à l'utilisateur de saisir le nombre de notes à rentrer puis qui calcule la moyenne de ses notes.

```
Script Python

somme = 0
for i in range(7):
   note = float(input("Saisissez la note n°"+str(i+1)+" : "))
   somme = somme + note # ou somme += note
moyenne = somme / 5
print(f"La moyenne des notes est {moyenne}")
```

% Script Python

```
somme = 0
nb_2_notes=int(input("Combien de notes voulez-vous rentrer ? "))
for i in range(nb_2_notes):
   note = float(input("Saisissez la note n°"+str(i+1)+" : "))
   somme = somme + note  # ou somme += note
moyenne = somme / nb_2_notes
print(f"La moyenne des notes est {moyenne}")
```

Exercice 8 : Ecrire un programme qui affiche le résultat suivant :

36 33 30 27 24 21 18 15 12 9 6 3 0

```
Script Python

for k in range(36, -1, -3):
    print(k, end=' ')
```

Exercice 9 : Ecrire un programme qui dessine un triangle comme ci-dessous avec une hauteur saisie par l'utilisateur.

```
****

***

***
```

```
hauteur=int(input("Veuillez saisir la hauteur du triangle : "))
for ligne in range(hauteur, 0, -1):
    print("*"*ligne)
```

Exercice 10 : Ecrire un programme qui dessine une figure comme ci-dessous avec une hauteur et une largeur de 6 (on a un carré).

```
******

****&&

***&&&

***&&&

**&&&&

*&&&&&&
```

```
hauteur=int(input("Veuillez saisir la hauteur du triangle : "))
for ligne in range(0, hauteur+1):
    fin_ligne=hauteur-ligne
    print("*"*fin_ligne+"&"*ligne)
```

Exercice 11 : Ecrire un programme qui dessine une figure comme ci-dessous avec une hauteur saisie par l'utilisateur.

&

&&

&&& &&&& &&&&& &&&&&&

```
hauteur=int(input("Veuillez saisir la hauteur du triangle : "))
for ligne in range(hauteur+1):
    print("*"*ligne)
```

Exercice 12 : Ecrire un programme qui demande 7 valeurs à l'utilisateur et qui affiche la somme de ces valeurs.

```
somme = 0
for i in range(7):
   note = float(input("Saisissez la note n°"+str(i+1)+" : "))
   somme += note # ou somme += note
print(f"La moyenne des notes est {somme}")
```

Exercice 13 : Ecrire un programme simule le lancer de 500 dés et qui affiche la moyenne des résultats obtenus.

```
Script Python

from random import *

somme = 0
for i in range(500):
    dé = randint(1,6)
    somme = somme + dé
moyenne = somme / 1000
print(f"La moyenne des 1000 lancers est {moyenne}")
```

Exercice 14 : Ecrire un programme qui demande à l'utilisateur un entier a et un entier n et qui calcule \$a^n\$ à l'aide d'une boucle for (on verra plus tard un algorithme plus rapide que celui-ci).

Remarque: il est interdit d'utiliser ** ici!

```
a = int(input("Veuillez saisir la valeurs de a (la base) : "))
n = int(input("Veuillez saisir la valeur de n (l'exponsant) : "))
puissance = 1
for i in range(n):
    puissance = puissance * a
print(f"{a} puissance {n} est égal à {puissance}")
```

Exercice 15 : Ecrire un programme qui demande de saisir un entier positif n et qui calcule la factorielle de n. La factorielle de n, notée \$n!\$ est définie par \$n! = 1\times 2\times 3\times...\times n\$. On a par exemple \$3! = 6\$.

```
& Script Python
```

```
n = int(input("Veuillez saisir un entier positif : "))
factorielle = 1
for i in range(1, n+1):
    factorielle *=i
print("La factorielle de ",n,"est",factorielle)
```

Exercice 16 : Ecrire un programme qui demande de saisir un entier positif n et qui calcule la **somme des carrés S** de 1 jusqu'à \$n^2\$.

```
$S(n)=1^2 + 2^2 + 3^2 +...+ n^2$.
```

```
On a par exemple S(3) = 14
```

```
script Python

nb=int(input('Donner le nombre voulu : '))
resultat=0
for k in range(nb+1):
    resultat+=k**2
print(resultat)
```

1. 4- Dessiner avec Turtle

La bibliothèque **turtle** permet de dessiner à l'écran.

Le petit programme commenté ci-dessous permet d'obtenir un tracé intéressant :

```
from turtle import * # On importe la bibliothèque turtle

speed(10)  # On règle la vitesse du tracé (un entier compris entre 1 et 10)

color('red', 'yellow')  # On fixe la couleur du tracé à 'rouge' et la couleur de rempissage à

'jaune'

begin_fill()

for i in range(36):
    forward(200)  # La tortue avance de 200 pixels
    left(170)  # La tortue tourne vers la gauche de 170°

end_fill()

done()  # Il faut terminer par done() pour lancer l'exécution du tracé
```

Voici quelques fonctionnalités de turtle :

Fonction	Description
forward(x)	Déplace la tortue de x pixels en marche avant.
backward(x)	Déplace la tortue de x pixels en marche arrière.
[left(x)]	Fait pivoter la tortue d'un angle de x degrés vers la gauche.
right(x)	Fait pivoter la tortue d'un angle de x degrés vers la droite.
goto(x,y)	Déplace la tortue au point de corrdonnées (x,y). Attention, l'axe des y est orienté vers le haut de l'écran comme en mathématiques!
penup()	Lève le crayon (la tortue arrêtera de tracer).
pendown()	Abaisse le crayon. La tortue se remettra à tracer.

hideturtle()	Cache la tortue.
showturtle()	Fait réapparaitre la tortue.
speed(n)	Règle la vitesse du tracé. n est un entier compris entre 0 et 10. 1 correspond à une vitesse lente, 10 correspond à une vitesse rapide. 0 permet d'obtenir un tracé instantané.
done()	Lance l'exécution du tracé.
circle(r)	Trace un cercle de rayon r. On peut ajouter une deuxième paramètre facultatif a si on souhaite tracer seulement tracer un arc de cercle d'angle a.
color(couleu r1,couleur2)	Définir la couleur du tracé (couleur1) et la couleur de remplissage (couleur2).
width(n)	Règle l'épaisseur du trait à n pixels.

Exercice 17 : Ecrire un programme qui utilise une boucle for pour tracer un carré de côté 100 pixels, avec un contour bleu et colorié en vert.

```
& Script Python
```

Exercice 18 : Ecrire un programme qui trace 10 cercles dont les rayons sont 10, 20, 30, ..., 100 et espacé de 2 fois le rayon

Choisir une couleur de tracé et une couleur de remplissage.

```
from turtle import * # On importe la bibliothèque turtle

speed(10)
color('blue','pink')
for k in range(10,110,10):

    begin_fill()
    circle(k)
    forward(20)
    end_fill()

done()
```

Exercice 19 : Dessiner le drapeau de l'allemagne avec Turtle.

```
from turtle import *

up()
goto(-250,0)
down()

color('black', 'black')
begin_fill()
for i in range(2):
    forward(300)
    left(90)
```

```
forward(75)
    left(90)
end_fill()
right(90)
forward(75)
left(90)
color('black', 'red')
begin_fill()
for i in range(2):
    forward(300)
    left(90)
    forward(75)
    left(90)
end_fill()
right(90)
forward(75)
left(90)
color('black', 'yellow')
begin_fill()
for i in range(2):
    forward(300)
    left(90)
    forward(75)
    left(90)
end_fill()
forward(100)
done()
```

& Script Python

```
#Avec une fonction
def rectangle(largeur, longueur, couleur1, couleur2):
    color(couleur1, couleur2)
    begin_fill()
    for i in range(2):
        forward(longueur)
        left(90)
        forward(largeur)
        left(90)
    end_fill()
up()
goto(-250,0)
down()
largeur=75
longueur=300
rectangle(largeur, longueur, 'black', 'black')
right(90)
forward(largeur)
left(90)
rectangle(largeur, longueur, 'black', 'red')
right(90)
forward(largeur)
left(90)
```

```
rectangle(largeur, longueur, 'black', 'yellow')
done()
```

Exercice 20 : Ecrire un programme qui dessine le diamant ci-dessous.

```
$\script Python

from turtle import *

speed(10)
color('blue')
goto(-250,0)
goto(250,0)
for i in range(11):
    goto(0,100)
    goto(i*50-250,0)
    goto(0,-100)
done()
```

& Script Python