

PARCOURS / ETAPE : Code UE : 4TPU275DSI
Epreuve : Programmation et Applications Interactives
Date : 06/04/2023 **Heure :** 9h00 **Durée :** 1h30
 Documents : autorisés
 Epreuve de M/Mme : Philippe Blasi

Exercice 1(8 points)

Analyser le code source ci-dessous et répondre aux questions suivantes :

- Quel est l'affichage obtenu à l'écran lors de l'exécution du programme ?
- Résumer (en une ou deux phrases) le rôle de chacune des fonctions **f1** et **f2** ?
- Est-ce que la fonction **f1** pose problème (par exemple, une erreur à l'exécution ou alors un résultat incohérent) lorsque la valeur fournie pour **p** est différente de celle fournie pour **q** ? Justifier la réponse, et si cette réponse est « OUI », proposer une version modifiée de la fonction qui corrige ce problème.
- Même question pour la fonction **f2**

```
#
-----
from ezCLI import *
#
-----
def f1(nb_row:int,nb_col:int) -> list:
    mat = [[0 for col in range(nb_col)] for row in range(nb_row)]
    for loop in range(nb_row*nb_col):
        row, col = k//nb_col, k%nb_col
        if row % 2 > 0:
            col = nb_col-col-1
        mat[row][col] = loop
    return mat
#
-----
def f2(mat:list) -> list:
    nb_row, nb_col = len(mat), len(mat[0])
    for row in range(nb_row):
        for col in range(row):
            mat[row][col], mat[col][row] = mat[col][row], mat[row][col]
    return mat
#
-----
mat = f1(5,5) # try with mat = f1(6,4)
print(grid(mat))
mat = f2(mat)
print(grid(mat))
#
-----
```

Exercice 2 (8 points)

A	B	C
1	2	
3	4	
D	E	F

A. Écrire une fonction **main()** permettant d'obtenir l'interface suivante (les widgets sont tous des boutons).

B. Modifier la fonction main pour créer l'interface en fonction des paramètres suivant reçus : **main(haut:str,bas:str,nb:int,col:int)**. Voici deux exemples des interfaces obtenues en fonction des paramètres.

A	B	C	D	E
1	2	3		
4	5	6		
7	8	9		
G	H	I	J	K

main("ABCDE","GHIJK",9,3)

A	B			
1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
C	D			

main("AB","CD",15,5)

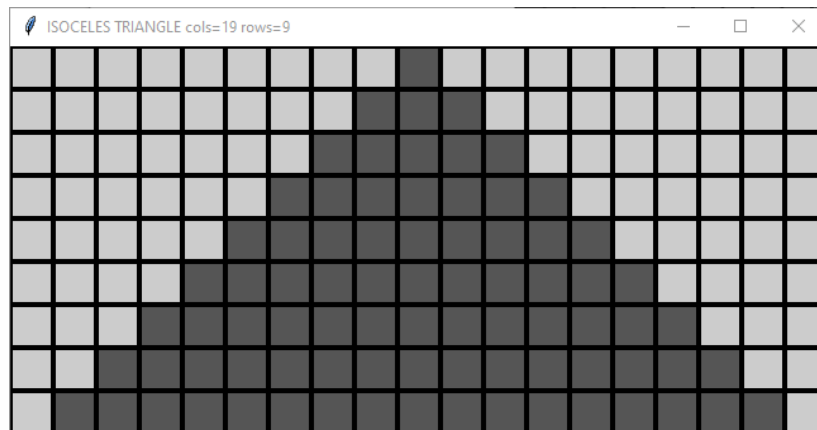
Exercice 3 (9 points)

On souhaite écrire une série de fonctions permettant d'afficher différents motifs dans des grilles rectangulaires dans une fenêtre, en utilisant au choix, la bibliothèque ezTK ou la bibliothèque tkinter. La taille de la grille est définie par nb_row et nb_col, tandis que le paramètre size (de valeur 30 par défaut) définit la taille en pixels de chaque case. Les cases sont colorées soit en gris clair (couleur = #CCC) pour la couleur de fond, soit en gris foncé (couleur = #555) pour dessiner les motifs. Elles sont entourées par une bordure noire de 2 pixels de large.

- A. Écrire une fonction `right_rectangle(nb_col:int,nb_row:int,size:int)` permettant d'afficher un triangle rectangle. Voici un exemple du résultat obtenu pour `right_rectangle(12,9,30)`.



- B. Écrire une fonction `isosceles_triangle(nb_col:int,nb_row:int,size:int)` permettant d'afficher un triangle isocèle. Voici un exemple du résultat obtenu pour `isosceles_triangle(19,9,30)`.



- C. Écrire une fonction `diagonal(nb_col:int,nb_row:int,size:int,space:int)` permettant d'afficher des diagonales espacées de `space` cases. Voici un exemple du résultat obtenu pour `diagonal(20,10,30,4)`.

