

AGR. Exercices. Corrigés

Exercice 1 : Vrai / Faux

Q.1 : FAUX. Exemple :

```
# Une boucle infinie
liste1 = [0]
for i in liste1 :
    # On parcourt liste1 mais on ajoute un élément à chaque fois !
    liste1.append(2)

print("La boucle est terminée")
```

Q.2 : VRAI. La liste étant non triée, il s'agit d'un parcours séquentiel donc pour une durée proportionnelle au nombre d'éléments de la liste.

Q.3 : VRAI. En moyenne, la valeur cherchée se situe au centre de la liste.

Q.4 : VRAI. Voir cours.

Q.5 : FAUX. Il suffit que l'élément cherché soit le premier de la liste.

Q.6 : VRAI. Voir le tableau comparatif du nombre maximal de boucles du cours.

Exercice 2 : QCM

Q.1 : Réponse 1. Voir cours.

Q.2 : Réponse 1. Aide : exécuter le programme pas à pas.

```
for i in range(5) :
    for j in range(i+1,5) :
        print(i+j,end=" ") # Le end = " " met un espace en séparateur.
```

1 2 3 4 3 4 5 5 6 7

Q.3 : Réponse 1. A chaque doublement de la valeur de la variable x, on effectue un tour en plus. Tout comme pour la recherche dichotomique où un doublement du nombre d'éléments d'une liste n'ajoute qu'un au coût de la recherche.

Q.4 : Réponse 4. Le programme compte le nombre d'éléments 'c' dans le tableau.

Q.5 : Réponse 4. La variable i varie de 0 à n-1 mais j de 0 à n-2 seulement (attention au range() !).

Exercice 3 : Moyenne pondérée

1/ $M = (17 \times 1 + 11 \times 2 + 12 \times 1 + 18 \times 1) / 5 = 14$. La moyenne est de 14/20.

2/ Voilà un exemple de programme de calcul de moyenne pondérée.

```
def moyennePonderee(notes, coefficients) :
    somPond = 0
    somCoef = 0
    nbreNotes = len(notes)

    for i in range(nbreNotes) :
        # On ajoute à SomPond la nouvelle note multipliée
        # par son coefficient
        somPond = somPond + notes[i]*coefficients[i]
        # On calcule la somme totale des coefficients
        somCoef = somCoef + coefficients[i]

    return somPond/somCoef

tabNotes = [17,12,11,18]
tabCoefficients = [1,2,1,1]

print("La moyenne est de :",moyennePonderee(tabNotes,tabCoefficients))
```

La moyenne est de : 14.0

3/ Voici un exemple de programme modifié.

```
def moyennePonderee(evaluations) :  
    somPond = 0  
    somCoef = 0  
  
    for note,coef in evaluations :  
        # On ajoute à SomPond la nouvelle note multipliée  
        # par son coefficient  
        somPond = somPond + note*coef  
        # On calcule la somme totale des coefficients  
        somCoef = somCoef + coef  
  
    return somPond/somCoef  
  
tabEvaluations = [ (17,1) , (12,2) , (11,1) , (18,1) ]  
  
print("La moyenne est de :",moyennePonderee(tabEvaluations))
```

La moyenne est de : 14.0

4/ Non car deux notes -respectivement coefficients- peuvent être identiques et la clé d'un dictionnaire est unique.

Exercice 4 : Détermination d'une valeur approchée de $\sqrt{3}$.

On choisit la fonction f définie par $f(x) = x^2 - 3$ et on résout l'équation $f(x) = 0$.

On peut choisir comme bornes 0 et 2 car $f(0) = -3$ et $f(2) = 1$. D'autre part, la fonction f est strictement croissante sur l'intervalle $[0,2]$, on peut donc effectuer une recherche dichotomique.

Le programme est identique que celui du cours !

Exercice 5 : Recherche d'extremum

Voici un exemple de programme

```
def f(x) :  
    return -x**2 + 4*x - 3  
  
def maximum(bornMin,bornMax,pas) :  
    absCourante = bornMin # Première valeur de l'intervalle  
    maxAbs = bornMin # Première valeur (par défaut)  
    maxOrd = f(bornMin) # Image de la première valeur par f  
  
    while absCourante < bornMax :  
        absCourante += pas  
  
        # Si la nouvelle image est supérieure à maxOrd  
        if f(absCourante) > maxOrd :  
            # On conserve l'antécédent  
            maxAbs = absCourante  
            # et l'image  
            maxOrd = f(absCourante)  
  
    return maxAbs,maxOrd  
  
borneInf = 1  
borneSup = 3  
pas = 0.00001  
print("le sommet de la fonction f est :", maximum(borneInf,borneSup,pas))
```

le sommet de la fonction f est : (2.0000000000006551, 1.0)