Sujet 2

/20

Nom : …… **Prénom :** ……Groupe : ……

EXERCICE 1 : QCM (6 points)

Q1 : Que va afficher le code suivant : **print ("bac"*2)**

Réponse : " **bacbac** "

Q2 : Quelle est la valeur de l'expression suivante : **55%10**

Réponse : 5

Q3 : Que contiendra x après l'exécution de : **x = int(93.6)**

Réponse : 93

Q4 : Que renvoi la fonction **range(6)**

Réponse : [0, 1, 2, 3, 4, 5]

Q5: Que va affiche le code suivant :

```
for i in range(8):
    print (i)
```

Réponse : Affiche les valeurs de 0 à 7

Q6 : Le mode d'ouverture « 'a' » (ajout) des fichiers veut dire que le fichier :

- ⊕ doit exister au préalable et que l'on pourra ajouter à sa fin des données
- ☐ ne doit pas exister au préalable et que l'on pourra ajouter à sa fin des données
- ☐ ne doit pas exister au préalable et que l'on pourra uniquement lire des données
- ☐ doit exister au préalable et que l'on pourra uniquement lire des données

Q7 : Soit la liste suivante :

>>> **Betails=["13 chèvres", "12 Poules","20 Canards","15 Mouton"]**

que représente :

Betails[0]

Réponse : "13 chèvres"

Betails[-2]

Réponse : "20 Canards"

Betails[0 :2]

Réponse : ['13 chèvres', '12 Poules']

Betails[5]

Réponse : Erreur

Q8 : Comment faire pour ajouter **"12 dindes"** à la liste « ***betails*** »?

Réponse : Betails.append("12 dindes")

Q9 : Comment faire pour supprimer l'élément "**20 Canards** " de la liste « ***betails*** »?

Réponse : Betails.remove("20 Canards ")

Q10 : Comment faire pour supprimer l'élément de position 2 de la liste « ***betails*** »?

Réponse : Betails.pop(2)

Q11 : Soit le dictionnaire « fruits » représentant des fruits et les quantités correspondant:

>>> fruits= ["carotte" :12, "poivron" :10, "oignon" :17, "choux" :20]

Que représente :

fruits["carotte"]

Réponse : 12

fruits.keys()

Réponse : ["carotte", "poivron", "oignon", "choux"]

fruits.values()

Réponse : [12, 10, 17, 20]

Comment faire pour afficher toutes les clés de votre dictionnaire en utilisant la boucle « for » ?

```
for c in fruits.keys() :
    print(c)
```

Comment faire pour afficher toutes les valeurs de votre dictionnaire en utilisant la boucle « for » ?

```
for v in fruits.keys() :
    print(v)
```

Q12 : Que va afficher le code suivant :

```
>>> liste = [1,4,5]
>>> liste.append('14')
>>> print(liste)
```

Réponse : [1,4,5,"14"]

Q13 : Donnez la commande vous permettant de créer ne matrice nommée « ***mat*** » et qui a les données suivantes :

12	14
15	17
13	44

```
>>> mat = list()
>>> mat = [[12,14] ,[15,17] ,[13,44]] ……..
```

Q14 : Soit les commandes suivantes :

```
>>> couleur = "bleu"
>>> print(couleur.index('b'))
```

Indiquez ce qu'affiche la dernière commande :

- ☐ 1
- ☐ 2
- ☑ 0

Q16 : Soit les commandes suivantes :

```
>>> texte = "***Bonjour***"
>>> print(texte.strip("*"))
```

Indiquez ce que va afficher la dernier commande:

"Bonjour"

Q17 : Soit les commandes suivantes :

```
>>> notes = [12,14]
>>> print(notes*2)
```

Indiquez ce que va afficher la dernier commande:

[12, 14, 12, 14]

Q17 : Soit les commandes suivantes :

```
>>> notes = [12,14]
>>> print(notes*2)
```

Indiquez ce que va afficher la dernier commande:

[12, 14, 12, 14]

EXERCICE 2 (2 points) : On vous demande d'écrire un programme qui lit à partir du clavier les notes d'un élève. Ces notes doivent être saisies en une seule fois en les séparant par un simple espace. Vous devez ensuite calculer la moyenne de l'élève.

Indication : *La méthode permettant de générer une liste à partir d'une chaine est split()*

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

EXERCICE 3 (2 points) : On vous demande d'améliorer le programme ci-dessus en lisant le texte depuis un fichier nommé « Fichier.data ».

.....
.....
.....

.....
.....
.....
.....
.....
.....
.....
.....

EXERCICE 4 (2 points) : On vous demande d'améliorer le programme ci-dessus en définissant une fonction nommée « **nbOccurences() »** et qui a pour argument une chaine de caractère (un texte) et retournant le nombre d'occurrences du mot « math » dans cette chaine de caractères.

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

```
ligneTexte = input("donnez une ligne de texte")
print("le nombre de voyelles dans votre ligne de texte est : ", nbOccurence(ligneTexte))
```

```
def nbVoyelles(t):
    voyelles = "aeiou"
    nbVoyelles=0
    for e in t:
        if e in voyelles:
            nbVoyelles=nbVoyelles+1
    return nbVoyelles
f = open("texte.txt","r")
texte = f.read()
print("le nombre de voyelles dans votre texte est : ", nbVoyelles(texte))
```

EXERCICE 3 (2 points) : On vous demande d'écrire un programme calculant la factorielle d'un entier naturel n. Vous devez permettre à l'utilisateur de calculer autant de factorielles qu'il le souhaite.

Indication de notation : Respect de la syntaxe + Bonne logique du programme + vérification de la validité des données saisies au clavier.

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

EXERCICE 4 (2 points) : On vous demande d'écrire un programme calculant le produit scalaire de deux vecteurs V1 et V2. On supposera que ces deux vecteurs ont le même nombre de composantes et seront représentés par deux listes de réels. On vous donne la formule du produit scalaire :

$$P = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} (y_1, y_3, y_3) = x_1 y_1 + x_2 y_2 + x_3 y_3$$

v1 = [2, 1, 1]

v2 = [3, 1, 5]

EXERCICE 5 (2 points) : On vous demande d'écrire un programme qui lit un texte depuis le clavier et qui calcul le nombre de caractères de ponctuation.

caracterePonctuation = [" . " , "\," , " ?" , " !" , " :"]

EXERCICE 6 (5 points) : On veut gérer les résultats d'une course de marathon. A cet effet, nous avons les noms des coureurs et les temps qu'ils ont mis. On vous demande, en vous servant d'un dictionnaire de :

1. Lire à partir du clavier les données (noms des coureurs avec les temps qu'ils ont mis). Il faut aussi lire le record national.
2. Afficher les résultats (noms des coureurs avec les temps qu'ils ont mis)
3. Afficher le coureur qui a eu la médaille d'or
4. Afficher le coureur qui a eu la médaille d'argent
5. Afficher le coureur qui a eu la médaille de bronze
6. Afficher le meilleur temps et indiquer si ce temps a battu le record national.

Indication : les temps doivent être indiqués en « heures:minutes:secondes » (exemple : "**2 :45 :13**" équivaut : 2h, 45 minutes set 13 secondes)