

Calculatrices, téléphones, traducteurs et ordinateurs interdits.

Attention : chaque partie doit être rédigée sur une copie séparée

N.B. : on s'attachera à fournir tout commentaire utile et à écrire de façon claire et lisible.

1^{ère} partie : Structures itératives (7 points)

1. Somme d'entiers

Ecrire un algorithme en pseudo-code qui demande un nombre entier compris entre 1 et 10 inclus et qui calcule la somme des entiers jusqu'à ce nombre inclus. Par exemple, si l'on entre 9, le programme affiche : 45.

2. Achats

Ecrire un algorithme en pseudo-code qui demande à l'utilisateur de saisir le prix du kilo en euros ainsi que la quantité au kg des achats d'un client (le prix et la quantité seront des entiers et le prix à 0 indiquera la fin de la saisie). L'algorithme devra alors calculer la somme qu'il doit, lire la somme qu'il paye, et simuler la remise de la monnaie en affichant le nombre de coupures de chaque sorte de "10 Euros", "5 Euros" et "1 Euro" à rendre. Attention, nous considérons que la somme saisie qu'il paye est supérieure à la somme qu'il doit.

----- { *prendre une nouvelle copie* } -----

2^{ème} partie : Listes et tableaux (7 points)

On souhaite réaliser un programme Python qui permet l'exploitation de données stockées dans un tableau. Nous voulons connaître les nombres des valeurs identiques ainsi que la plus longue suite d'éléments consécutifs. Le tableau contient des entiers appartenant à [0, 9].

1. Saisie et affichage des données.

- a. Le tableau peut contenir au maximum 100 valeurs, mais l'utilisateur peut vouloir traiter un nombre inférieur à 100. Il doit avoir la possibilité de définir ce nombre. Attention : vous devez vérifier qu'il ne définit pas un nombre de données supérieur à 100.
- b. Lors de la saisie, il faut vérifier que seulement des valeurs dans [0, 9] sont acceptées.
- c. La saisie démarre avec un tableau vide, complété par l'utilisateur jusqu'au nombre de valeurs souhaitée (1.a).

2. Comptage des nombres de valeurs saisies.

Les nombres de valeurs [0, 9] stockées dans le tableau doivent être calculés et affichés. Vous ne pouvez pas utiliser la méthode `count` Python.

Par exemple, pour le tableau de données : [1, 1, 0, 1, 1, 8, 8, 8, 4]

Votre programme doit afficher : [1, 4, 0, 0, 1, 0, 0, 0, 3, 0]

3. Recherche de la plus longue suite d'éléments consécutifs ayant la même valeur dans le tableau

Le programme doit aussi afficher la valeur, le début de la suite ainsi que sa longueur.

Par exemple, pour le tableau de la question 2 : [1, 1, 0, 1, 1, 8, 8, 8, 4]

Votre programme doit afficher : Elément = 8, Max = 3, Début = 5

----- { *prendre une nouvelle copie* } -----

3^{ème} partie : Chaines de caractères (6 points + 1 point bonus)

Soit la chaine de caractères suivante :

```
s = "Les jeux préférés de xxx sont : Minecraft, xx, Tic Tac Toe.  
Mais xxx devrait réviser INF1"
```

1. Que valent `s[10]`, `s[4:8]`, `s[-4:-1]`, `s[1:4:-1]`, `s[10:2:-2]` ?
2. Ecrire un algorithme en pseudo-code qui affiche l'indice de la première occurrence de "x" dans `s` ou -1 s'il n'y a pas de "x".
3. Ecrire un programme Python qui remplace tous les "x" par des "y" dans la chaine de caractères `s`.
4. Ecrire un programme Python qui remplace tous les "xxx" par des "Bob" et tous les "xx" par des "Froggy", dans la chaine de caractères `s`.