

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<sup>ère</sup> 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.

```
Variables N, i, Som : Entier
Debut
    Ecrire "Entrez un nombre : "
    Lire N
    TantQue N < 1 ou N > 10 faire
        Ecrire "Entrez un nombre : "
        Lire N
    FinTantQue
    Som ← 0
    Pour i ← 1 à N
        Som ← Som + i
    Ecrire "La somme est : ", Som
Fin
```

Ou (solution meilleure)

```
Variables N, i, Som : Entier
Debut
    Répéter
        Ecrire "Entrez un nombre : "
        Lire N
    Jusqu'à N >= 1 et N <= 10
    Som ← 0
    Pour i ← 1 à N
        Som ← Som + i
    Ecrire "La somme est : ", Som
Fin
```

**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.

#### Variables

E, Q, somdue, M, Reste, Nb10E, Nb5E : **Entier**

#### Debut

E ← 1

somdue ← 0

**TantQue** E <> 0 **faire**

**Ecrire** "Entrez le prix : "

**Lire** E

**Si** E <> 0 **alors**

**Ecrire** " Entrez la quantité en kg :"

**Lire** Q

        somdue ← somdue + E \*Q

**FinSi**

**FinTantQue**

**Ecrire** "Vous devez :", somdue, " euros"

**Ecrire** "Montant versé :"

**Lire** M

Reste ← M - somdue

Nb10E ← 0

**TantQue** Reste >= 10 **faire**

    Nb10E ← Nb10E + 1

    Reste ← Reste - 10

**FinTantQue**

Nb5E ← 0

**Si** Reste >= 5

    Nb5E ← 1

    Reste ← Reste - 5

**FinSi**

**Ecrire** "Rendu de la monnaie :"

**Ecrire** "Billets de 10 E : ", Nb10E

**Ecrire** "Billets de 5 E : ", Nb5E

**Ecrire** "Pièces de 1 E : ", reste

#### Fin

Ou (solution meilleure)

#### Variables

E, Q, somdue, M, Reste, Nb10E, Nb5E : **Entier**

#### Debut

somdue ← 0

**Répéter**

**Ecrire** "Entrez le prix en euros : "

**Lire** E

**Si** E <> 0 **alors**

**Ecrire** " Entrez la quantité en kg :"

**Lire** Q

    somdue ← somdue + E \*Q

**FinSi**

**jusqu'à** E = 0

```

Ecrire "Vous devez :", somdue, " euros"
Ecrire "Montant versé :"
Lire M
Reste ← M - somdue
Nb10E ← 0
TantQue Reste >= 10
Nb10E ← Nb10E + 1
Reste ← Reste - 10
FinTantQue
Nb5E ← 0
Si Reste >= 5
Nb5E ← 1
Reste ← Reste - 5
FinSi
Ecrire "Rendu de la monnaie :"
Ecrire "Billets de 10 E : ", Nb10E
Ecrire "Billets de 5 E : ", Nb5E
Ecrire "Pièces de 1 E : ", reste
Fin

```

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

## 2<sup>ème</sup> 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.

- 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.
- Lors de la saisie, il faut vérifier que seulement des valeurs dans [0, 9] sont acceptées.
- La saisie démarre avec un tableau vide, complété par l'utilisateur jusqu'au nombre de valeurs souhaitée (1.a).

```

#####
#### 2ème partie : Listes et tableaux ####
#####
tab = [] # tableau de données
T_MAX = int(100)

print('*****')
print('*** Question 1 ***')
print('*****')
# def de la taille réelle du tableau <
taille_reelle = int(input(f'Taille tableau < {T_MAX} : '))
while (taille_reelle < 0) or (taille_reelle > T_MAX):
    taille_reelle = int(input(f'Taille tableau < {T_MAX} : '))

# Saisie des données
for i in range(taille_reelle):
    data = int(input(f'Valeur {i} (> 0 et < 9) :'))

```

```

while (data < 0) or (data > 9):
    data = int(input(f'Valeur {i} (> 0 et < 9) :'))
    tab.append(data)
print(tab)

```

## 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]

```

print('*****')
print('*** Question 2 ***')
print('*****')
# comptage des valeurs 1ère version non-optimale
compteur = [0] * 10
for nb in range(9):
    c = 0
    for i in range(taille_reelle):
        if nb == tab[i]:
            c += 1
        compteur[nb] = c
print(compteur)

# comptage des valeurs 2ème - a - version optimale
compteur = [0] * 10
for i in range(taille_reelle):
    compteur[tab[i]] += 1
print(compteur)

# comptage des valeurs 2ème - b - version optimale encore mieux
compteur = [0] * 10
for nb in tab:
    compteur[nb] += 1
print(cont)

```

## 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 :                    Élément = 8, Max = 3, Début = 5

```

print('*****')
print('*** Question 3 ***')
print('*****')
max = 0
# Nb max de valeurs
for i in range(1, 9):
    if compteur[max] < compteur[i]:
        max = i
print('Valeur max = ', max, 'Nb max = ', compteur[max])

fin = 1
nb = 1
maxi = 1
# Plus longue suite de nb
for i in range(1, taille_reelle):
    if tab[i] == tab[i - 1] :
        nb += 1

```

```

else :
    nb = 1
    if nb > maxi:
        maxi = nb
        fin = i
if (taille_reelle >= 2) :
    print('Elément = ', tab[fin], 'Max = ', maxi, 'Début = ', fin-maxi+1)
else:
    print('Elément = ', tab[0], 'Max = ', tab[0], 'Début = ', 0)

```

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

### 3<sup>ème</sup> 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] ?

```

#####
#### 3ème partie : Chaines de caractères ####
#####
s = "Les jeux préférés de xxx : Minecraft, Tic Tac Toe. Mais xxx devrait
plutôt réviser INF1"

print('*****')
print('*** Question 1 ***')
print('*****')
# Que vaut (0.25)
print(s[10])
# réponse : "r"

# Que vaut (0.25)
print(s[4:8])
# réponse : "jeux"

# Que vaut (0.25)
print(s[-4:-1])
# réponse : "INF"

# Que vaut (0.25)
print(s[1:4:-1])
# réponse : ""

# Que vaut (0.25)
print(s[10:2:-2])
# réponse : "r uj"

```

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".

```

print('*****')
print('*** Question 2 ***')
print('*****')
# Ecrire un programme qui cherche l'indice la première occurrence de x (1.5)
i = 0
trouve = False
s2 = "Il fait beauxxx"
while i < len(s) and not trouve :
    if s[i] == 'x' :
        trouve = True

```

```

else :
    i += 1
if trouve :
    print("x se trouve à : ", i)
else :
    print("Non trouve")

```

3. Ecrire un programme Python qui remplace tous les "x" par des "y" dans la chaîne de caractères s.

```

print('*****')
print('*** Question 3 ***')
print('*****')
# Ecrire un programme Python qui remplace les 'x' par des 'y' (1.5)
for i in range(len(s)) :
    if s[i] == 'x' :
        s = s[:i] + "y" + s[i+1:]
print(s)

```

4. Ecrire un programme Python qui remplace tous les "xxx" par des "Bob" et tous les "xx" par des "Froggy", dans la chaîne de caractères s.

```

print('*****')
print('*** Question 4 ***')
print('*****')
# Ecrire un programme Python qui remplace tous les "xxx" par des "Bob" et
tous les "xx" par des "Froggy", dans la chaîne de caractères s. (1.5)
s = "Les jeux préférés de xxx sont : Minecraft, xx, Tic Tac Toe. Mais xxx
devrait plutôt réviser INF1 ou xx"
for i in range(len(s)-2) :
    if s[i:i+3] == 'xxx' :
        s = s[:i] + "Bob" + s[i+3:]
    elif s[i:i+2] == 'xx' :
        s = s[:i] + "Froggy" + s[i+2:]
if s[-2:] == "xx" :
    s = s[:-2] + "Froggy"
print(s)

```