

Examen Médian

Durée : 1 heure 30

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 : Listes et tableaux (6 points)

Soit l'algorithme ci-dessous, qui prend en entrée un tableau, tab, de n nombres entiers :

Entrée : tab, n

j <- 0

k <- 0

y <- 0

z <- 0

pour i allant de 1 à n faire

 si tab[i] % 2 = 0 alors

 j <- j + 1

 tab1[j] <- tab[i]

 y <- y + tab[i]

 sinon

 k <- k + 1

 tab2[k] <- tab[i]

 z <- z + tab[i]

 fin_si

fin_pour

si j > 0 alors

 y <- y / j

fin_si

```

si k > 0 alors
    z <- z / k
fin_si
afficher("point 1 : ", y)
afficher("point 2 : ", z)
si y>z alors
    afficher("point 3-1 : Voici les nombres de la série 1 :" et rester à la ligne)
    pour i allant de 1 à j faire
        afficher(tab1[i] et rester à la ligne)
    fin_pour
sinon
    afficher("point 3-2 : Voici les nombres de la série 2 :" et rester à la ligne)
    pour i allant de 1 à k faire
        afficher(tab2[i] et rester à la ligne)
    fin_pour
fin_si

```

1. Considérer les entrées suivantes :
 - Entrée 1 : n = 10 ; tab = [1 3 5 7 9 2 6 3 8 4]
 - Entrée 2 : n = 5 ; tab = [3 2 6 11 7]
 Pour chacune de ces entrées, quel affichage on aura aux points d'affichage :
 Point 1, Point 2, Point 3-1 et Point 3-2 ?
2. Décrire ce que fait cet algorithme.
3. Ecrire le programme python correspondant.

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

2^{ème} partie : Structures itératives (6 points)

1. Qu'est-ce qu'une structure itérative en algorithmique ?
2. Donner les trois structures itératives identifiées en algorithmique et préciser leurs différences ainsi que les conditions idéales d'utilisation.
3. Pour chacune des trois structures identifiées en algorithmique, donner sa mise en œuvre en python.

4. Écrire un programme en python qui affiche les 20 premiers termes de la table de multiplication par 7, en signalant au passage (à l'aide d'un astérisque) ceux qui sont des multiples de 3. Justifiez le choix de la structure utilisée.

Exemple : 7 14 21 * 28 35 42 * 49 56 63 * 70 77 84 * 91 98 105 * 112 119 126
* 133 140

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

3^{ème} partie : Chaines de caractères (8 points)

Soit les chaines de caractères suivantes :

S1 = "123456789"

S2 = "abcdefghi"

S3 = "Université de Technologie de Compiègne"

S4 = "portez ce vieux whisky au juge blond qui fume"

1. En utilisant le slicing Python extraire la chaine "5678" de s1.
2. En utilisant le slicing Python, comment extraire de la chaine s2 une lettre sur deux à l'envers et obtenir "igeca" ?
3. Ecrire le programme Python qui permet de créer une chaine de caractère s5 qui mélange des chaines de caractères de même longueur. En mélangeant s1 et s2, on obtient : "1a2b3c4d5e6f7g8h9i".
4. Modifier le programme précédent pour mélanger des chaines de longueurs différentes. En mélangeant s1 et s3, on obtient ; "1U2n3i4v5e6r7s8i9té de Technologie de Compiègne".
5. Ecrire un algorithme ou programme Python qui affiche les lettres majuscules de la chaine s3, ici "UTC"
6. Un pangramme est une phrase comportant au moins une fois chaque lettre de l'alphabet. Par exemple, s4 est un pangramme. Ecrire un algorithme ou programme Python qui affiche si une chaine de caractère est un pangramme ou non. On considère que toutes les lettres sont des minuscules.