

UE ALGORITHMIQUE

Devoir - Session 2

Temps de travail estimé : **2 h 30 à 3 heures** – Evaluation à distance à effectuer **individuellement**

Il sera tenu compte de la lisibilité des algorithmes. Pour chaque algorithme réalisé, on définira soigneusement les variables, les fonctions et les actions introduites pour résoudre le problème.

Les exercices sont indépendants et peuvent être traités dans n'importe quel ordre.

Dans un exercice, pour résoudre une question, vous pouvez utiliser une action ou une fonction définie dans une question précédente. Le barème donné est indicatif.

SUJET (3 pages)

Exercice 1 : Plateaux d'une séquence d'entiers (5 points)

Étant donnée une séquence d'entiers **non vide représentée dans un fichier**, on appelle **plateau** une sous-séquence d'éléments consécutifs ayant tous une même valeur h ; h est appelée la hauteur du plateau.

Par exemple, dans la séquence [4, 4, -1, 5, 5, 5, 0, 0, 7, 7, 2, 2, 2, 5, 5, 5] on trouve successivement un plateau de hauteur 4 et de longueur 2, un plateau de hauteur -1 et de longueur 1, un plateau de hauteur 5 et de longueur 3, un plateau de hauteur 0 et de longueur 2, etc. Dans cet exemple la longueur maximale des plateaux est 3, et il y a trois plateaux de longueur 3, de hauteurs 5, 2 et 5.

On considère une séquence d'entiers non vide représentée dans un fichier.

a) Ecrire un algorithme itératif qui affiche la hauteur et la longueur du premier plateau de la séquence. Dans l'exemple ci-dessus, le résultat affiché est :

hauteur : 4, longueur : 2

b) Ecrire un algorithme itératif qui calcule et affiche, pour une séquence donnée, la longueur maximale des plateaux et le nombre de plateaux de longueur maximale. Dans l'exemple ci-dessus, le résultat affiché est :

*longueur maximale d'un plateau : 3
nombre de plateaux de longueur maximale : 3*

Exercice 2 : Somme des chiffres d'un entier (3 points)

a) Ecrire la fonction récursive **sommeChiffres** qui calcule la somme des chiffres d'un entier :

sommeChiffres : fonction (x : entier ≥ 0) \rightarrow entier ≥ 0

// *SommeChiffres(x)* renvoie la somme des chiffres composant x

Exemples : `sommeChiffres(195728) = 32`

b) Ecrire une version itérative de cette même fonction.

Exercice 3 : Points-cols (6 points)

Soit t un tableau d'entiers à deux dimensions composé de N lignes et de M colonnes, supposé déjà initialisé.

t : tableau sur $[0..N-1 ; 0..M-1]$ d'entiers

Écrire un algorithme qui recherche dans t les éléments qui sont à la fois un maximum sur leur ligne et un minimum sur leur colonne. Ces éléments sont appelés des points-cols. Afficher les positions et les valeurs de tous les points-cols trouvés, et le nombre de points-cols trouvés.

Exemple :

t vaut :

4	1	7	3	1
3	-8	9	3	-1
3	4	9	3	6
2	1	2	1	0

Les éléments en **gras** sont des points-cols:

L'algorithme affiche :

il y a un point-col en ligne 3 colonne 0 valant 2

il y a un point-col en ligne 3 colonne 2 valant 2

nombre de points-cols trouvés : 2

Conseil :

Pour simplifier l'écriture de l'algorithme principal, on pourra se définir des fonctions intermédiaires qui seront décrites séparément.

Exercice 4 : Les plus grands (6 points)

On considère une séquence d'entiers représentée dans un fichier. On souhaite mémoriser dans un tableau la liste des N ($N=10$) plus grands entiers distincts de cette séquence. Si la séquence ne comporte que k entiers distincts ($k < N$), on ne mémorisera que ces k entiers.

Écrire l'algorithme qui lit une séquence d'entiers représentée sur un fichier et mémorise dans un tableau t défini sur l'intervalle $[0..N-1]$, les k plus grands entiers distincts ($k \leq N$). A la fin du parcours, l'algorithme affichera les entiers mémorisés dans t . L'ordre des éléments mémorisés dans t n'a pas d'importance.

Vous définirez et réaliserez toutes les actions et fonctions que vous jugerez nécessaires pour résoudre le problème.

Exemple 1 :

Fichier lu : 1 4 9 2 6 3 6 -1 7 20 10 2 15 6 14 8 -2 7 19

Le contenu de t ($N=10$) pourrait être le suivant :

8	4	9	19	6	14	7	20	10	15
0									N-1

l'algorithme affiche :

Les 10 plus grands entiers sont : 8 4 9 19 6 14 7 20 10 15

Exemple 2 :

Fichier lu : 1 4 9 2 6 3 6 7 2 15 2 7 1 4 6 3

Le contenu de t (N=10) pourrait être le suivant :

1	4	9	2	6	3	7	15		
0							k		N-1

l'algorithme affiche :

Les 8 plus grands entiers sont : 1 4 9 2 6 3 7 15