

Séance 10 : Vendredi 08/03/2019

Exercices du TD5 : Tableau 1D

1. Déclarer un tableau de 9 réels et l'initialiser avec la valeur 0.

```

Déclaration : T : Tableau [9] de réels
Procédure InitTableau (T : Tableau [9] de réels)
Données : aucune
Précondition : aucune
Donnée / Résultat : tableau T initialisé a 0
Variable locale : i : entier
Début
    Pour i allant de 0 à 8 par pas de 1 faire
        T[i] ← 0
    FinPour
Fin InitTableau

```

En C on peut écrire `float T={0}` mais attention cela ne fonctionne qu'avec la valeur 0 sinon il faut donner TOUTES les valeurs initiales du tableau.

2. Écrire l'algorithme d'un sous-programme `RemplirTableau` qui remplit un tableau de 9 notes (réelles) avec des valeurs demandées à l'utilisateur.

12	18,5	13,25	8,75	16	15	13,5	12	17
----	------	-------	------	----	----	------	----	----

```

Procédure RemplirTableau (T : Tableau [9] de réels)
Données : aucune
Précondition : aucune
Donnée / Résultat : tableau rempli de valeurs lues
Variable locale : i : entier
Début
    Pour i allant de 0 à 8 par pas de 1 faire
        Afficher (« Donner la ième valeur »)
        Saisir (T[i])
    FinPour
Fin remplirTableau

```

3. Écrire l'algorithme d'un sous-programme `AfficheTableau` qui permet d'afficher les 9 valeurs contenues dans le tableau.

```

Procédure AfficheTableau (T : Tableau [9] de réels)
Précondition : aucune
Données / Résultat : tableau
Description : affichage du contenu du tableau
Variable locale : i : entier
Début
    Pour i allant de 0 à 8 par pas de 1 faire
        Afficher (T[i])
    FinPour
Fin

```

4. Écrire l'algorithme d'un sous-programme `RechercheMin` qui permet de retourner l'indice de la plus petite valeur contenue dans le tableau.

Fonction **RechercheMin** (T : Tableau [9] de réels) : entier

Précondition : aucune
 Donnée / résultat : Tableau T
 Résultat : indice du plus petit élément du tableau T
 Variables locales : i, ind_min : entier, min : réel
 Début
 min ← T[0]
 ind_min ← 0
 Pour i allant de 1 à 8 par pas de 1 faire
 Si (min > T[i]) alors
 min ← T[i]
 ind_min ← i
 Fin Si
 Fin Pour
 Retourner ind_min
 Fin RechercheMin

Et si on voulait **RechercheMax !!!**

Fonction RechercheMax (T : Tableau [9] de réels) : entier
 Précondition : aucune
 Donnée / Résultat : Tableau T
 Résultat : indice du plus grand élément du tableau T
 Variable locale : i, ind_max : entier, max : réel
 Début
 max ← T[0]
 ind_max ← 0
 Pour i allant de 1 à 8 par pas de 1 faire
 Si (max < T[i]) alors
 max ← T[i]
 ind_max ← i
 Fin Si
 Fin Pour
 Retourner ind_max
 Fin RechercheMax

5. Écrire l'algorithme d'un sous-programme `SommeTableau` qui calcule la somme des valeurs du tableau précédent : `SommeTableau = 126`.

Fonction SommeTableau (T : Tableau [9] de réels) : réel
 Précondition : aucune
 Donnée / Résultat : Tableau T
 Résultat : somme des éléments du tableau T
 Variable locale : i, som : entier
 Début
 som ← 0
 Pour i allant de 0 à 8 par pas de 1 faire
 som ← som + T[i]
 Fin Pour
 Retourner som
 Fin SommeTableau

6. Écrire l'algorithme d'un sous-programme permettant de renverser le contenu d'un tableau contenant des caractères.

Avant exécution :

T1 :

A	B	C	D	E	F	G	H	I
---	---	---	---	---	---	---	---	---

Après exécution :

T2 :

I	H	G	F	E	D	C	B	A
---	---	---	---	---	---	---	---	---

Procédure renverse (T : Tableau [9] de caractères)
 Précondition : aucune
 Données / résultat : T
 Description : renverse tous les éléments du tableau T
 Variable locale : i : entier tampon : caractère
 Début
 Pour i allant de 0 à 3 par pas de 1 faire // 3 étant (taille/2 -1)
 tampon ← T[i]
 T[i] ← T[8-i]
 T[8-i] ← tampon
 FinPour
 Fin Reverse

On peut aussi faire une version où on utilise un deuxième tableau pour stocker le résultat.

7. Écrire l'algorithme d'un sous-programme `Cumul` qui remplacera les valeurs de chaque case par la somme des valeurs des cases précédentes.

Initial	12	8	5,5	14	13	10,5	7	16	9
Final	12	20	25,5	39,5	52,5	63	70	86	95

Procédure cumul (T : Tableau [9] d'entiers)
 Données / résultat : T
 Précondition : aucune
 Description : cumule les valeurs de T
 Variable locale : i : entier
 Début
 Pour i allant de 1 à 9 par pas de 1 faire
 T[i] ← T[i-1]+T[i]
 FinPour
 Fin Reverse

8. On dispose de deux tableaux de tailles identiques T1 et T2 contenant les notes de contrôle continu et d'examen d'un étudiant pour N UE données (N passé en paramètre). On souhaite à partir de ces deux tableaux constituer un troisième tableau (noté T3) contenant les moyennes pondérées par les coefficients suivants : Contrôle continu (T1) : coefficient 1, et Examen terminal (T2) : coefficient 2.

Écrire l'algorithme d'un sous-programme `CalculMoyenneTableau` permettant à partir des deux premiers tableaux de remplir un troisième tableau T3 en utilisant les coefficients donnés.

Exemple de calcul : $14 = (12*1 + 15*2) / 3$ avec ici N = 6.

T1 :	12	8	5,5	14	13	10,5			
------	----	---	-----	----	----	------	--	--	--

T2 :	15	3	12	17	8	10			
------	----	---	----	----	---	----	--	--	--

T3 :	14	4,66	9,83	16	9,66	10,16			
------	----	------	------	----	------	-------	--	--	--

Procédure CalculMoyenneTableau (T1 : Tableau [MAX_TAB] de réel, T2 : Tableau [MAX_TAB] de réel, T3 : Tableau [MAX_TAB] de réel, N entier)

Précondition : aucune

Donnée : N

Donnée / Résultat : T1 / T2 / T3

Variable locale : i : entier

Début

 Pour i allant de 0 à N-1 par pas de 1 faire

$T3[i] \leftarrow (T1[i] + 2 * T2[i]) / 3$

 FinPour

Fin CalculMoyenne

Pour s'entraîner

9. Écrire l'algorithme d'un sous-programme qui calcule et "retourne" un tableau contenant les N premiers termes de la suite U_n définie par :

Procédure suite (T : Tableau [MAX_TAB] de réels, N entier)

Précondition : aucune

Données / résultat : T

Donnée : N entier

Description : remplit T avec les valeurs de la suite

Variable locale : i : entier

Début

$T[0] \leftarrow 1$

 Pour i allant de 1 à N-1 par pas de 1 faire

$T[i] \leftarrow T[i-1] / i$

 FinPour

Fin Suite