

Programmation et Structure de Données

Structure de données de Base
Les Tableaux (partie 2)

Exemples

Exemple 1 :

Écrire un algorithme qui déclare et remplit un tableau contenant les voyelles de l'alphabet latin.

Algo EX1

Var

T : Tableau[8] de caractère;

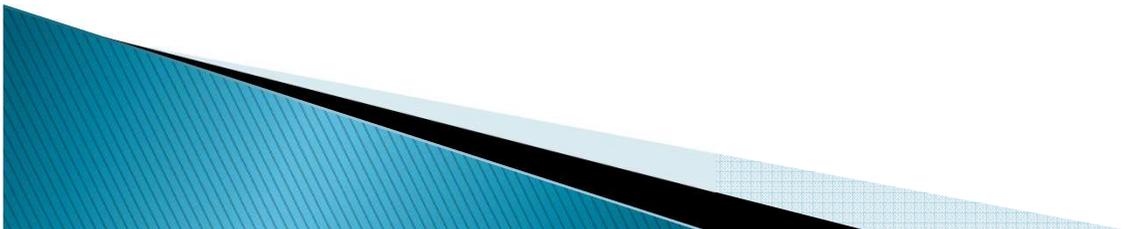
Début

T[0]='A'; T[1]='I'; T[2]='O'; T[3]='U';

T[4]='E'; T[5]='É'; T[6]='Y'; T[7]='\0';

Fin.

RQ : Seulement dans le cas des tableaux de caractères (chaines de caractères) on doit ajouter le caractère spécial '\0' en fin du tableau pour marquer la fin de la chaine de caractère.



Exemples

Exemple 2:

Écrire un algorithme qui déclare et remplisse un tableau de N valeurs numériques lues par l'utilisateur (N entier entré par l'utilisateur).

Algo EX2

Const dim ← 100;

Var

 i, N : entier;

 T : Tableau[dim] d'entier;

Début

 faire

 écrire ("Enter un nombre entre 1 et ", dim);

 lire (N);

 Tque (N < 1 ou N > dim);

 pour(i de 0 à N-1) faire

 écrire ("Enter l'element numéro", i+1 " :");

 lire (T[i]);

 Fpour;

Fin.

Exemples

Exemple 3:

Écrire l'algorithme d'un sous programme qui prend un tableau en argument puis renvoi le produit des valeurs de ce tableau.

Fonction prod (**T:Tableau[]** de entier, N:entier) : entier

Début

Var i,S : entier;

S ← 1;

pour(i de 0 à N-1) faire

 S ← S*T[i];

Fpour;

Return S;

Fin;

RQ : Si un tableau est passé en argument d'un sous programme on laisse la dimension vide, pour qu'il fonctionne avec n'importe quelle dimension, mais si on précise la dimension il travail seulement avec les tableaux dont la dimension est mentionnée et il renvoi erreur dans le cas d'une autre dimension .

Fonction prod_rec (T:Tableau[] de entier, N:entier) : entier

Début

si (N = 1) alors retourner T[0];

sinon retourner T[N-1] * prod_rec(T,N-1);

Fin;

Exemples

Exemple 4:

Écrire un algorithme d'un sous programme qui prend deux tableaux en argument puis renvoi le schtroumpf de ces deux tableaux.

Le schtroumpf est égale à la somme des produits de chaque élément du 1ere tableau avec chaque élément du 2eme tableau 2, et additionner le tout.

Exemple : T1 = (4 8 7 12) et T2 = (3 6)

Le Schtroumpf = $(3*4)+(3*8)+(3*7)+(3*12)+(6*4)+(6*8)+(6*7)+(6*12) = 279$

Fonction Sch (T1:Tableau[] de entier, N:entier
T2:Tableau[] de entier, M:entier) : entier

Début

Var S,i,j : entier;

S = 0;

pour (i de 0 à N-1) faire

pour (j de 0 à M-1) faire

S ← S + T1[i] * T2[j];

Fpour;

Fpour;

Return S;

Fin;

Exemples

Exemple 5:

Écrire un programme C++ qui permet de lire une chaîne de caractères stockée dans un tableau puis la renverser et afficher le résultat. Exemple :

Avant exécution T = (A B C D E F) Après exécution T = (F E D C B A)

```
main()
{
    const int dim=5;
    char T[dim+1]; int i,N; char c;
    do
    { if(strlen(T)>dim)cout <<"\nTaille Incorrect!!!";
      cout <<"\nEntrer une chaîne de taille inférieure à " <<dim<<": "; cin >> T;
    } while (strlen(T)>dim);
    N = strlen(T);
    for(i=0;i<N/2;i++)
    { c = T[i];
      T[i] = T[N-i-1];
      T[N-i-1] = c;
    }
    cout <<"\nAprès inversion : " <<T;
}
```

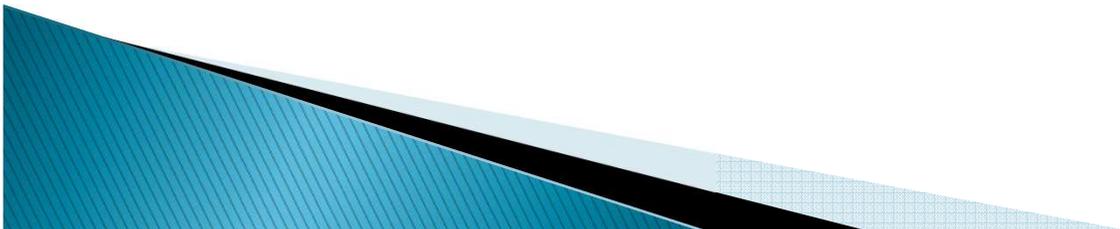
Exemples

Exemple 6:

Écrire un programme en C++ qui permet de construire le triangle de Pascal de degré N (donné par l'utilisateur) en utilisant une matrice carrée $(N+1) \times (N+1)$.

Exemple : Triangle de Pascal de degré 0, 1, 2, 3, 4 et 5 :

N = 0	1					
N = 1	1	1				
N = 2	1	2	1			
N = 3	1	3	3	1		
N = 4	1	4	6	4	1	
N = 5	1	5	10	10	5	1



Exemples

Solution Triangle de Pascal :

```
const int dim = 100;
main()
{
    int i,j,N;    int P[dim][dim];
    do
    {
        cout <<"Entrer le degre du triangle entre 0 et " << dim-1 <<" : "; cin >> N;
    } while (N<0 || N>(dim-1));
    for(i=0; i<=N; i++)
        for(j=0; j<=i; j++)
            if(i==j || j==0) P[i][j]=1;
            else P[i][j] = P[i-1][j] + P[i-1][j-1];
    for(i=0; i<=N; i++)
    {
        for(j=0; j<=i; j++) cout <<P[i][j] << setw(3);
        cout <<"\n";
    }
}
```