Tablouri bidimensionale- notiuni de baza

Un tablou bidimensional= o matrice, este o structura de date interna si omogena(adica toate elementele au ac. tip). Elementele dintr-o matrice pot fi identificate dupa linia si coloana pe care se afla. Se numeste matrice cu m linii si n coloane, un tablou cu m linii si n coloane.

Declararea tablourilor bidimensionale (matrice) se face în C/C++ similar cu a tablourilor unidimensionale, dar trebuie precizate două dimensiuni fizice, maximale: numărul maxim de linii și numărul maxim de coloane ale matricei:

Declarare matrice:

**tipDeBază denumire[NumarLinii][NumarColoane];**

tipdebaza – repr. tipul elementelor matricei

denumire- repr. numele variabilei cu care vom identifica matricea in program

de exemplu: int a[5][10];

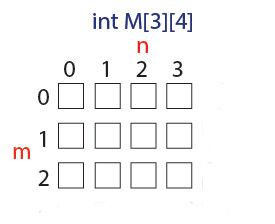
Mai sus s-a declarat un tablou bidimensional (o matrice) **a** cu **5 linii** și **10 coloane** cu elemente numere intregi.

-float B[5][10]; (declara o matrice de 5 linii si 10 coloane continand numere reale)

-char C[20][12]; (declara o matrice de 20 de linii si 12 coloane continand caractere alfanumerice).

**Observatie:** Numerotarea liniilor si coloanelor unei matrici incepe de la valoarea 0. Astfel, prima linie si prima coloana a matricii este linia 0 si coloana 0, a doua linie este linia 1, respectiv coloana este 1, etc.

Numarul total de elemente din matrice este dat de produsul mxn.



**Observatie:** Putem numerota liniile si coloanele de la 1, in acest caz, pentru a nu avea probleme de memorie, declaram matricea cu numar in plus de linii si coloane(m+1, n+1). Astfel, daca vrem sa salvam o matrice cu 3 linii si 4 coloane, incepand numerotarea de la 1, vom scrie m[4][5] in loc de m[3][4].

Mai jos putem vedea citirea si afisarea unei matrici salvata cu numerotoarea incepand de la 1.

#include <iostream>

using namespace std;

int n,m,i,j;

int main()

{

cout<<”nr. linii:”; //se citesc numarul de linii si coloane

cin>>m;

cout<<”nr. coloane:”;

c>>n;

inta[m][n];

//citirea pe rand a elementelor matricii- linie cu linie

for(i=1;i<=m;i++)

for(j=1;j<=n;j++)

{cout<<”a[”<< i<<j<<”]=”;cin>>a[i][j];}

//afisarea matricii

for(i=1;i<=m;i++)

{for(j=1;j<=n;j++)

cout<<a[i][j]<<" ";

cout<<endl;

}

}

MATRICI PATRATICE

Daca nr. de linii este egal cu nr. de coloane atunci matricea se numeste matrice patratica(m=n). Intr-o astfel de matrice distingem:

-diagonala principala , formata din elentele care au indicele de linie egal cu cel de coloana, adica elementele de forma a[i][j] cu i=j, elemente pe care le vom referi mai simplu cu un singur indice: a[i][i]. In consecinta pentru accesarea si prelucrarea elementelor diagonalei principale intr-o matrice vom folosi o singura structura for....

Aplicatie 1.

***Sa se calculeze suma elementelor de pe diagonala principala a unei matrici patratice.***

#include <iostream>

using namespace std;

int main()

{

int n,i,j,a[20][20],s=0;

cout<<"n=";cin>>n;

for(i=1;i<=n;i++)

for(j=1;j<=n;j++)

{

cout<<"a["<<i<<"]["<<j<<"]=";

cin>>a[i][j];

}

cout<<"Matricea data este:"<<endl;// se afiseaza matricea ca tablou

for(i=1;i<=n;i++)

{

for(j=1;j<=n;j++)

cout<<a[i][j]<<" ";

cout<<endl;

}

for(i=1;i<=n;i++)//se parcurg el. diag. principale si se insumeaza

s=s+a[i][i];

cout<<endl<<"Suma elementelor de pe diagonala principala a matricii este:"<<s<<endl;

return 0;

}

**Ex de date de executie:**

Daca se citeste n=3

Si elementele matricei introduse , pe rand , de la tastatura ar fi:

1. 2 3 4 5 6 7 8 9

Atunci s-ar afisa matricea cu 3 lini si 3 coloane care ar contine el. citite mai sus: 1 2 3

4 5 6

7 8 9 , iar suma el diag. principale ar fi s= 1+5+9=15.

**Zonele determinate de diagonala principala :**

-Pe diagonala principală sunt caracterizate de4 conditia: i=j

-Sub diagonala principala caracterizate de4 conditia: i>j

-Deasupra diagonalei principale caracterizate de4 conditia: i<j

**Exemplu 2.**

Se citeste o matrice, sa se afiseze elementele de deasupra si de sub diagonala principala si deasemenea sa se calculeze si sa se afiseze suma lor.

#include <iostream>

using namespace std;

int main()

{

int x[30][30],n,m,ok,aux,s=0;

int i,j;

cout<<"n=";cin>>n;

for(i=1;i<=n;i++)

for(j=1;j<=n;j++)

{cout<<"x["<<i<<"]["<<j<<"]=";

cin>>x[i][j];

}

cout<<"matricea este\n";

for(int i=1;i<=n;i++)

{

for(int j=1;j<=n;j++)

cout<<x[i][j]<<" ";

cout<<"\n";

}

cout<<"Elementele de deasupra diagonalei principale sunt:\n";

for(int i=1;i<=n;i++)

{for(int j=1;j<=n;j++)

if(i<j){cout<<x[i][j]<<" ";s=s+x[i][j];}

cout<<endl<<”suma el. de deasupra diagonalei principale:”<<s;

cout<<endl;

cout<<endl<<"Elementele de sub diagonala principala sunt :\n";

int t=0;//t - variabila pentru suma el. de sub diag. principala

for(int i=1;i<=n;i++)

{for(int j=1;j<=n;j++ )if(i>j)

{cout<<x[i][j]<<" ";t=t+x[i][j];}

cout<<endl;

}

cout<<endl<<"suma elementelor de sub diagonala principala:"<<t;}

tema

1. Se dă o matrice x, cu n linii şi n coloane, cu elemente întregi. Să se determine numărul elementelor din matrice care sunt < 24 şi se găsesc deasupra diagonalei principale .
2. Se dă o matrice x, cu n linii şi n coloane, cu elemente întregi. Să se determine elemental minim(maxim) din matrice.
3. Indicatie: seinitializeaza min=a[1][1], elemental din coltul stanga sus ,se parcurge integral matricea verificand daca minimul ales nu este bun (**min>a[i][j]** ), caz in care va fi inlocuit. La fel se procedeaza pentru a determina el. maxim din matrice.