/\* Wap in c++ to perform basic operations on **2-D matrices** using **UDF** \*/

#include<iostream.h>

#include<process.h>

#include<conio.h>

int a[80][80],b[80][80]; // 2-D matrices

int i=0, j=0;

int m, n, p, q; // Matrix 1 : m= no. of Rows , n=no. of Columns

// Matrix 2 : p= no. of Rows , q=no. of Columns

int mc, oc, sum=0;

// Function Declarations

void add(int a[80][80],int b[80][80]);

void subtract(int a[80][80],int b[80][80]);

void multiply(int a[80][80],int b[80][80]);

void multiply1(int b[80][80],int a[80][80]);

void equivalency ( int a[80][80],int b[80][80]);

void upper\_tri(int a[80][80]);

void lower\_tri(int a[80][80]);

void diagonal\_1\_a(int a[80][80]);

void row\_sum\_a(int a[80][80]);

void column\_sum\_a(int a[80][80]);

void main() // main body

{clrscr();

int ch,op; char choice;

l: cout<<"\n\nDescribe the Size of matrix 1 : ";

cout<<"\n\n\n\tEnter the number of rows : ";

cin>>m;

cout<<"\n\n\tEnter the number of columns : ";

cin>>n;

cout<<"\n\nEnter the elements of matrix 1 : \n\n";

for(i=0; i<m; i++)

for(j=0; j<n; j++)

cin>>a[i][j];

cout<<"\n\n\n\nDescribe the size of matrix 2 : ";

cout<<"\n\n\n\tEnter the number of rows : ";

cin>>p;

cout<<"\n\n\tEnter the number of columns : ";

cin>>q;

cout<<"\n\nEnter the elements of matrix 2 : \n\n";

for(i=0; i<p; i++)

for(j=0; j<q; j++)

cin>>b[i][j];

for(i=0; i<80; i++)

cout<<"-";

cout<<"\n\nMatrix 1 is : \n\n";

for(i=0; i<m; i++)

{

for(j=0; j<n; j++)

{

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

}

cout<<"\n\n";

}

cout<<"\n\nMatrix 2 is : \n\n";

for(i=0; i<p; i++)

{

for(j=0; j<q; j++)

{

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

}

cout<<"\n\n";

}

do{

cout<<"\n\nChoose from the following menu : ";

cout<<"\n\n1. Add two 2-D arrays ";

cout<<"\n\n2. Subtract two 2-D arrays ";

cout<<"\n\n3. Multiply two 2-D arrays " ;

cout<<"\n\n4. Check equivalency of two 2-D matrices.";

cout<<"\n\n5. Upper triangular matrix ";

cout<<"\n\n6. Lower Triangular matrix ";

cout<<"\n\n7. Display and find sum of diagonal elements ";

cout<<"\n\n8. Display and find row wise sum of 2-D array ";

cout<<"\n\n9. Display and find column wise sum of 2-D array ";

cout<<"\n\n10. Exit ";

cout<<"\n\n\n\nEnter your choice : ";

cin>>ch;

switch(ch)

{

case 1 : if(m==p)

if(n==q)

add(a,b);

else

{

cout<<"\n\nNumber of columns of matrices are not same. \nEnter the matrices again.";

goto l;

}

else

{

cout<<"\n\nNumber of rows of matrices are not same. \nEnter the matrices again.";

goto l;

}

break;

case 2 : if(m==p)

if(n==q)

{

kk:

cout<<"\n\n\t1. Subtract Matrix 2 from Matrix 1 ";

cout<<"\n\n\t1. Subtract Matrix 1 from Matrix 2 ";

cout<<"\n\n\tEnter your choice : ";cin>>mc;

switch(mc)

{

case 1 : subtract(a,b);break;

case 2 : subtract(b,a);break;

default: cout<<"\n\nPlease enter desired keyword. ";goto kk;

}

}

else

{

cout<<"\n\nNumber of columns of matrices are not same. \nEnter the matrices again.";

goto l;

}

else

{

cout<<"\n\nNumber of rows of matrices are not same. \nEnter the matrices again.";

goto l;

}

break;

case 3 : cout<<"\n\n\t(i). Multiply matrix 1 with matrix 2 ";

cout<<"\n\n\t(ii). Multiply matrix 2 with matrix 1 ";

cout<<"\n\n\tEnter your choice : "; cin>>op;

switch(op)

{

case 1 : if(n==p)

multiply(a,b);

else

{

cout<<"\n\nMultiplicaion not possible, as no. of columns of first matrix != no. of rows of second matrix. \nEnter the matrices again.";

goto l;

}

break;

case 2 : if(m==q)

multiply1(b,a);

else

{

cout<<"\n\nMultiplicaion not possible, as no. of columns of first matrix != no. of rows of second matrix. \nEnter the matrices again.";

goto l;

}

break;

}

break;

case 4: if(m==p)

if(n==q)

equivalency(a,b);

else

cout<<"\n\nMatrices are not equivalent.";

else

cout<<"\n\nMatrices are not equivalent.";

break;

case 5:

cout<<"\n\n\t(i). Display Upper triangular of matrix 1 ";

cout<<"\n\n\t(ii).Display Upper triangular of matrix 2 ";

cout<<"\n\n\tPlease enter your choice : "; cin>>mc;

switch(mc)

{

case 1 : if(m==n)

upper\_tri(a);

else

cout<<"\n\nUpper Triangle cannot be displayed. Matrix should be square. ";

break;

case 2 : if(p==q)

upper\_tri(b);

else

cout<<"\n\nUpper Triangle cannot be displayed. Matrix should be square. ";

break;

}

break;

case 6: cout<<"\n\n\t(i). Display Lower triangular matrix 1 ";

cout<<"\n\n\t(ii).Display Lower triangular matrix 2 ";

cout<<"\n\n\tEnter your choice : "; cin>>mc;

switch(mc)

{

case 1 : if(m==n)

lower\_tri(a);

else

cout<<"\n\nLower Triangle cannot be displayed. Matrix should be square. ";

break;

case 2 : if(p==q)

lower\_tri(b);

else

cout<<"\n\nLower Triangle cannot be displayed. Matrix should be square. ";

break;

}

break;

case 7:

cout<<"\n\n\t(i). Diagonals of matrix 1 ";

cout<<"\n\n\t(ii).Diagonals of matrix 2 ";

cout<<"\n\n\tEnter your choice : "; cin>>oc;

switch(oc)

{

case 1 : cout<<"\n\n\t\t1. Find sum of diagonal (top-left to bottom-right) ";

cout<<"\n\n\t\t2. Find sum of diagonal (bottom-left to top-right) ";

cout<<"\n\n\t\tEnter your choice : "; cin>>mc;

diagonal\_1\_a(a);

break;

case 2 : cout<<"\n\n\t\t1. Find sum of diagonal (top-left to bottom-right) ";

cout<<"\n\n\t\t2. Find sum of diagonal (bottom-left to top-right) ";

cout<<"\n\n\t\tEnter your choice : "; cin>>mc;

diagonal\_1\_a(b);

break;

}

break;

case 8:

cout<<"\n\n\t1. Display and find row wise sum of a matrix 1 : ";

cout<<"\n\n\t2. Display and find row wise sum of a matrix 2 : ";

cout<<"\n\n\tEnter your choice : "; cin>>oc;

switch(oc)

{

case 1 : row\_sum\_a(a); break;

case 2 : row\_sum\_a(b); break;

}

break;

case 9: cout<<"\n\n\t1. Display and find column wise sum of a matrix 1 : ";

cout<<"\n\n\t2. Display and find column wise sum of a matrix 2 : ";

cout<<"\n\n\tEnter your choice : "; cin>>oc;

switch(oc)

{

case 1 : column\_sum\_a(a); break;

case 2 : column\_sum\_a(b); break;

}

break;

case 10 : exit(10);

}

cout<<"\n\n\nWant to choose again => "; cin>>choice;

}while(choice=='y'||choice=='Y');

getch();

}

//----------------------------------------------------------- Function Definitions ---------------------------------------------------------

void add(int a[80][80],int b[80][80])

{

int c[80][80];

for(i=0; i<m; i++)

for(j=0; j<n; j++)

c[i][j]=a[i][j]+b[i][j];

cout<<"\n\n\nResultant matrix by addition is : \n\n";

for(i=0; i<m; i++)

{

for(j=0; j<n; j++)

{

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

}

cout<<"\n";

}

}

void subtract(int a[80][80],int b[80][80])

{int l,u;

if (mc==1)

{

l=m;

u=n;

}

else

l=p;

u=q;

int c[80][80];

for(i=0; i<l; i++)

for(j=0; j<u; j++)

c[i][j]=a[i][j]-b[i][j];

cout<<"\n\n\nResultant matrix by Subtraction is : \n\n";

for(i=0; i<m; i++)

{

for(j=0; j<n; j++)

{

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

}

cout<<"\n";

}

}

void multiply(int a[80][80],int b[80][80])

{

int c[80][80];

j=0;

int k=0;

cout<<”\n\nResultant of Matrix Multiplication : “;

for(i=0; i<m; i++)

{ cout<<"\n";

for(j=0; j<q; j++)

{

c[i][j]=0;

for(k=0; k<n; k++)

{ c[i][j]=c[i][j]+(a[i][k]\*b[k][j]);

}

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

}

}

}

void multiply1(int b[80][80],int a[80][80])

{

int c[80][80];

j=0;

int k=0;

int sum=0;

for(i=0; i<n; i++)

{

for(k=0; k<n; k++)

{

for(j=0; j<n; j++)

{

sum=sum+b[i][j]\*a[j][k];

}

c[i][k]=sum;

sum=0;

}

}

cout<<”\n\nResultant of Matrix Multiplication : “;

cout<<"\n\n\n\n";

for(i=0; i<p; i++)

{

for(j=0; j<n; j++)

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

cout<<"\n";

}

}

void equivalency ( int a[80][80],int b[80][80])

{int count=0;

for(i=0; i<m; i++)

{

for(j=0; j<n; j++)

{

if(a[i][j]==b[i][j])

++count;

}

}

if(count==(m\*n))

cout<<"\n\nThe matrices are equivalent.";

else

cout<<"\n\nThe matrices are not equivalent.";

}

void upper\_tri(int a[80][80])

{int s;

if(mc==1)

s=m;

else

s=p;

cout<<”\n\nUpper triangle of matrix is : “;

for(i=0; i<s; i++)

{

for(int l=0; l<=(i+1)\*2; l++)

cout<<" ";

for(j=i; j<s; j++)

{

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

}

cout<<"\n";

}

}

void lower\_tri(int a[80][80])

{int z;

if(mc==1)

z=m;

else

z=p;

cout<<”\n\nLower triangle of matrix is : “;

for(i=0; i<z; i++)

{

for(j=0; j<=i; j++)

{

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

}

cout<<"\n";

}

}

void diagonal\_1\_a(int a[80][80])

{int ui;

if(oc==1)

ui=m;

else

ui=p;

if(mc==2)

{

sum=0;

cout<<"\n\nDiagonal Elements are : \n\n";

for(i=0; i<ui; i++)

{

cout<<a[i][ui-(i+1)]<<"\n";

sum+=a[i][ui-(i+1)];

}

cout<<"\n\nSum of diagonal elements is : "<<sum;

}

else if(mc==1)

{ sum=0;

cout<<"\n\nDiagonal Elements are : \n\n";

for(i=0; i<ui; i++)

{

cout<<a[i][i]<<"\n";

sum+=a[i][i];

}

cout<<"\n\nSum of diagonal elements is : "<<sum;

}

}

void row\_sum\_a(int a[80][80])

{int v,u;

if(oc==1)

{

v=m;

u=m;

}

else

{

v=p;

u=q;

}

for(i=0; i<v; i++)

{ sum=0;

for(j=0; j<u; j++)

{

sum=sum+a[i][j];

}

cout<<"\n\nSum of Row "<<i+1<<" = "<<sum;

}

}

void column\_sum\_a(int a[80][80])

{int v,u;

if(oc==1)

{

v=m;

u=m;

}

else

{

v=p;

u=q;

}

for(j=0; j<v; j++)

{ sum=0;

for(i=0; i<u; i++)

{

sum=sum+a[i][j];

}

cout<<"\n\nSum of Column "<<j+1<<" = "<<sum;

}

}

**Output:**

Describe the Size of matrix 1 :

Enter the number of rows : 3

Enter the number of columns : 3

Enter the elements of matrix 1 :

1

2

3

4

5

6

7

8

9

Describe the size of matrix 2 :

Enter the number of rows : 3

Enter the number of columns : 3

Enter the elements of matrix 2 :

3

2

1

1

2

3

2

0

1

-----------------------------------------------------------------------------------------------------------------------------------------------------

Matrix 1 is :

1 2 3

4 5 6

7 8 9

Matrix 2 is :

3 2 1

1 2 3

2 0 1

Choose from the following menu :

1. Add two 2-D arrays

2. Subtract two 2-D arrays

3. Multiply two 2-D arrays

4. Check equivalency of two 2-D matrices.

5. Upper triangular matrix

6. Lower Triangular matrix

7. Display and find sum of diagonal elements

8. Display and find row wise sum of 2-D array

9. Display and find column wise sum of 2-D array

10. Exit

Enter your choice : 1

Resultant matrix by addition is :

4 4 4

5 7 9

9 8 10

Want to choose again => y

Choose from the following menu :

1. Add two 2-D arrays

2. Subtract two 2-D arrays

3. Multiply two 2-D arrays

4. Check equivalency of two 2-D matrices.

5. Upper triangular matrix

6. Lower Triangular matrix

7. Display and find sum of diagonal elements

8. Display and find row wise sum of 2-D array

9. Display and find column wise sum of 2-D array

10. Exit

Enter your choice : 2

1. Subtract Matrix 2 from Matrix 1

2. Subtract Matrix 1 from Matrix 2

Enter your choice : 1

Resultant matrix by Subtraction is :

-2 0 2

3 3 3

5 8 8

Want to choose again => Y

Choose from the following menu :

1. Add two 2-D arrays

2. Subtract two 2-D arrays

3. Multiply two 2-D arrays

4. Check equivalency of two 2-D matrices.

5. Upper triangular matrix

6. Lower Triangular matrix

7. Display and find sum of diagonal elements

8. Display and find row wise sum of 2-D array

9. Display and find column wise sum of 2-D array

10. Exit

Enter your choice : 3

(i). Multiply matrix 1 with matrix 2

(ii). Multiply matrix 2 with matrix 1

Enter your choice : 1

Resultant of matix Multiplication :

11 6 10

29 18 25

47 30 40

Want to choose again =>y

Choose from the following menu :

1. Add two 2-D arrays

2. Subtract two 2-D arrays

3. Multiply two 2-D arrays

4. Check equivalency of two 2-D matrices.

5. Upper triangular matrix

6. Lower Triangular matrix

7. Display and find sum of diagonal elements

8. Display and find row wise sum of 2-D array

9. Display and find column wise sum of 2-D array

10. Exit

Enter your choice : 4

The matrices are not equivalent.

Want to choose again =>Y

Choose from the following menu :

1. Add two 2-D arrays

2. Subtract two 2-D arrays

3. Multiply two 2-D arrays

4. Check equivalency of two 2-D matrices.

5. Upper triangular matrix

6. Lower Triangular matrix

7. Display and find sum of diagonal elements

8. Display and find row wise sum of 2-D array

9. Display and find column wise sum of 2-D array

10. Exit

Enter your choice : 5

(i). Display Upper triangular of matrix 1

(ii).Display Upper triangular of matrix 2

Please enter your choice : 1

Upper triangle of matrix is :

1 2 3

5 6

9

Want to choose again => Y

Choose from the following menu :

1. Add two 2-D arrays

2. Subtract two 2-D arrays

3. Multiply two 2-D arrays

4. Check equivalency of two 2-D matrices.

5. Upper triangular matrix

6. Lower Triangular matrix

7. Display and find sum of diagonal elements

8. Display and find row wise sum of 2-D array

9. Display and find column wise sum of 2-D array

10. Exit

Enter your choice : 6

(i). Display Lower triangular matrix 1

(ii).Display Lower triangular matrix 2

Enter your choice : 2

Lower Triangle of matrix is :

3

1 2

2 0 1

Want to choose again => Y

Choose from the following menu :

1. Add two 2-D arrays

2. Subtract two 2-D arrays

3. Multiply two 2-D arrays

4. Check equivalency of two 2-D matrices.

5. Upper triangular matrix

6. Lower Triangular matrix

7. Display and find sum of diagonal elements

8. Display and find row wise sum of 2-D array

9. Display and find column wise sum of 2-D array

10. Exit

Enter your choice : 7

(i). Diagonals of matrix 1

(ii).Diagonals of matrix 2

Enter your choice : 1

1. Find sum of diagonal (top-left to bottom-right)

2. Find sum of diagonal (bottom-left to top-right)

Enter your choice : 1

Diagonal Elements are :

1

5

9

Sum of diagonal elements is : 15

Want to choose again => y

Choose from the following menu :

1. Add two 2-D arrays

2. Subtract two 2-D arrays

3. Multiply two 2-D arrays

4. Check equivalency of two 2-D matrices.

5. Upper triangular matrix

6. Lower Triangular matrix

7. Display and find sum of diagonal elements

8. Display and find row wise sum of 2-D array

9. Display and find column wise sum of 2-D array

10. Exit

Enter your choice : 8

1. Display and find row wise sum of a matrix 1 :

2. Display and find row wise sum of a matrix 2 :

Enter your choice : 1

Sum of Row 1 = 6

Sum of Row 2 = 15

Sum of Row 3 = 24

Want to choose again => Y

Choose from the following menu :

1. Add two 2-D arrays

2. Subtract two 2-D arrays

3. Multiply two 2-D arrays

4. Check equivalency of two 2-D matrices.

5. Upper triangular matrix

6. Lower Triangular matrix

7. Display and find sum of diagonal elements

8. Display and find row wise sum of 2-D array

9. Display and find column wise sum of 2-D array

10. Exit

Enter your choice : 9

1. Display and find column wise sum of a matrix 1 :

2. Display and find column wise sum of a matrix 2 :

Enter your choice : 2

Sum of Column 1 = 6

Sum of Column 2 = 4

Sum of Column 3 = 5

Want to choose again => N