**/\***

**=====================================================================**

**Assignment : Matrix Operations**

**Name : Gaurav Ghati**

**class : SE 10**

**Batch : F 10**

**====================================================================**

**\*/**

#include<stdio.h>

void add(int[10][10], int[10][10], int, int, int, int);

void diff(int[10][10], int[10][10], int, int, int, int);

void printMatrix(int[10][10], int, int);

void suddelPoint(int[10][10], int, int);

void transpose(int [10][10], int, int);

void mul(int[10][10], int[10][10], int, int, int, int);

void displayMenu();

int i, j, k;

int main(){

int r1, r2, c1, c2;

int a[10][10] , b[10][10];

printf("Enter number of rows and columns for 1st matrix :");

scanf("%d%d", &r1, &c1);

printf("Enter Elements of 1st matrix : ");

for(i=0; i<r1 ; i++){

for(j=0; j<c1; j++){

scanf("%d", &a[i][j]);

}

}

printf("Enter number of rows and columns for 2nd matrix :");

scanf("%d%d", &r2, &c2);

printf("Enter Elements of 2nd matrix : ");

for(i=0; i<r2 ; i++){

for(j=0; j<c2; j++){

scanf("%d", &b[i][j]);

}

}

int choice = 0;

int enter = 1;

displayMenu();

while(enter==1){

scanf("%d", &choice);

switch(choice) {

case 1 : printf("Addition of A and B\n");

add(a,b,r1,c1,r2,c2);

break;

case 2: printf("Subtraction of A and B :\n");

diff(a,b,r1,c1,r2,c2);

break;

case 3: printf("Transpose of A is :\n");

transpose(a,r1,c1);

break;

case 4: printf("For A :\n");

suddelPoint(a,r1,r2);

break;

case 5: printf("Transpose of B is :\n");

transpose(b,r2,c2);

break;

case 6: printf("For B :\n");

suddelPoint(b,r2,c2);

break;

case 7: printf("Multiplication of A and B\n");

mul(a,b,r1,c1,r2,c2);

break;

case 8: displayMenu();

break;

case 9: enter=0;

}

}

}

void displayMenu(){

printf("<<-----------Menu------------->>\n");

printf("1. Addition of A and B\n");

printf("2. Subtraction of A and B\n");

printf("3. Transpose of A\n");

printf("4. saddel points of A\n");

printf("5. Transpose of B\n");

printf("6. saddel points of B\n");

printf("7. Multiplication of A and B\n");

printf("8. Display menu\n");

printf("9. Exit\n");

}

void printMatrix(int a[10][10], int r1, int c1){

for(i=0; i<r1 ; i++){

for(j=0; j<c1; j++){

printf("%d ", a[i][j]);

}

printf("\n");

}

}

void add(int a[10][10], int b[10][10], int r1, int c1, int r2, int c2){

if(r1 != r2 || c1 != c2){

printf("\nThe number of rows or the number of columns are not same for performing addition operation.\n");

return;

}

int c[10][10];

for(int i=0; i<r1 ; i++){

for(int j=0; j<c1; j++){

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

}

}

printMatrix(c, r1, c1);

}

void diff(int a[10][10], int b[10][10], int r1, int c1, int r2, int c2){

if(r1 != r2 || c1 != c2){

printf("\nThe number of rows or the number of columns are not same for performing subtraction operation.\n");

return;

}

int c[10][10];

for(i=0; i<r1 ; i++){

for(j=0; j<c1; j++){

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

}

}

printMatrix(c, r1, c1);

}

int minRow(int a[10][10], int r1, int c1, int row){

int min = a[row][0];

for(int i=0; i<c1; i++){

if(a[row][i] < min){

min = a[row][i];

}

}

return min;

}

int maxRow(int a[10][10], int r1, int c1, int row){

int max = a[row][0];

for(int i=0; i<c1; i++){

if(a[row][i] > max){

max = a[row][i];

}

}

return max;

}

int minCol(int a[10][10], int r1, int c1, int col){

int min = a[0][col];

for(int i=0; i<r1; i++){

if(a[i][col] < min){

min = a[i][col];

}

}

return min;

}

int maxCol(int a[10][10], int r1, int c1, int col){

int max = a[0][col];

for(int i=0; i<r1; i++){

if(a[i][col] > max){

max = a[i][col];

}

}

return max;

}

void suddelPoint(int a[10][10], int r1, int c1){

int rowMin[r1], colMin;

int point = 0;

for(i=0; i<r1; i++){

for(j=0; j<c1; j++){

if( (minRow(a,r1,c1,i) == maxCol(a,r1,c1,j)) || (maxRow(a,r1,c1,i) == minCol(a,r1,c1,j))){

point++;

printf("Saddle point is at (%d, %d) and is %d\n",i , j, a[i][j]);

}

}

}

if(point==0)

printf("There is no suddel point.\n");

}

void transpose(int a[10][10], int r1, int c1){

for(i=0;i<c1;i++){

for(j=0;j<r1;j++){

printf(" %d", a[j][i]);

}

printf("\n");

}

}

void mul(int a[10][10], int b[10][10], int r1, int c1, int r2, int c2){

int c[10][10];

if(c1!=r2){

printf("\nThe col of first matrix is not equal to row of second matrix\n");

} else {

for(i=0;i<r1;i++){

for(j=0;j<c2;j++){\

c[i][j] = 0;

for(k=0;k<r1; k++){

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

}

}

}

printMatrix(c, r1, c2);

}

}

**OUTPUT :**

Enter number of rows and columns for 1st matrix :3 3

Enter Elements of 1st matrix : 1 2 3

4 5 6

7 8 9

Enter number of rows and columns for 2nd matrix :3 3

Enter Elements of 2nd matrix : 9 8 7

6 5 4

3 2 1

<<-----------Menu------------->>

1. Addition of A and B

2. Subtraction of A and B

3. Transpose of A

4. saddel points of A

5. Transpose of B

6. saddel points of B

7. Multiplication of A and B

8. Display menu

9. Exit

ENTER CHOICE (8 For Display Menu) : 1

Addition of A and B

10 10 10

10 10 10

10 10 10

ENTER CHOICE (8 For Display Menu) : 2

Subtraction of A and B :

-8 -6 -4

-2 0 2

4 6 8

ENTER CHOICE (8 For Display Menu) : 3

Transpose of A is :

1 4 7

2 5 8

3 6 9

ENTER CHOICE (8 For Display Menu) : 4

For A :

Saddle point is at (0, 2) and is 3

Saddle point is at (2, 0) and is 7

ENTER CHOICE (8 For Display Menu) : 5

Transpose of B is :

9 6 3

8 5 2

7 4 1

ENTER CHOICE (8 For Display Menu) : 6

For B :

Saddle point is at (0, 2) and is 7

Saddle point is at (2, 0) and is 3

ENTER CHOICE (8 For Display Menu) : 7

Multiplication of A and B

30 24 18

84 69 54

138 114 90

ENTER CHOICE (8 For Display Menu) : 8

<<-----------Menu------------->>

1. Addition of A and B

2. Subtraction of A and B

3. Transpose of A

4. saddel points of A

5. Transpose of B

6. saddel points of B

7. Multiplication of A and B

8. Display menu

9. Exit

ENTER CHOICE (8 For Display Menu) : 9