#include <stdio.h>

#include<stdlib.h>

int \*\*create(int n,int m);

void getv(int n,int m,int \*\*a);

void print(int n,int m,int \*\*a);

void add(int n,int m,int p,int q,int \*\*a,int \*\*b);

void sub(int n,int m,int p,int q,int \*\*a,int \*\*b);

void mul(int n,int m,int p,int q,int \*\*a,int \*\*b);

void trs(int n,int m,int \*\*a);

void sdp(int n,int m,int \*\*a);

int main()

{ int r1,c1,r2,c2;

int t,k;

int \*\*a;

int \*\*b;

printf("Value for row of 1:- ");

scanf("%d",&r1);

printf("Value for column of 1:-");

scanf("%d",&c1);

printf("Value for row of 2:- ");

scanf("%d",&r2);

printf("Value for column of 2:-");

scanf("%d",&c2);

printf(" matrix 1:-\n");

a=create(r1,c1);

getv(r1,c1,a);

print(r1,c1,a);

printf(" matrix 2:-\n");

b=create(r2,c2);

getv(r2,c2,b);

print(r2,c2,b);

do

{ printf(" enter 1 addition of matrices , 2 for subtraction,3 for multiplication,4 for transpose,5 for sdp,0 for exit\n");

printf("Enter choice:\n");

scanf("%d",&k);

switch(k)

{

case 1:

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

case 2:

sub(r1,c1,r2,c2,a,b);break;

case 3:

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

case 4:

printf("enter matrix for transpose:-");

scanf("%d",&t);

if(t==1)

{ trs(r1,c1,a);}

else if(t==2)

{ trs(r2,c2,b);}

else

printf("re-enter");

break;

case 5 : printf("enter matrix for sdp:-");

scanf("%d",&t);

if(t==1)

{ sdp(r1,c1,a);}

else if(t==2)

{ sdp(r2,c2,b);}

else

printf("re-enter");

break;}} while(k!=0);

return 0;}

int \*\*create(int n,int m)

{ int \*\*z; int i;

z=((int\*\*)malloc(n\*sizeof(int\*\*)));

for(i=0;i<m;i++){z[i]=((int\*)malloc(m\*sizeof(int)));}

return(z);}

void getv(int n,int m,int \*\*a)

{ int j,i=0;

printf("Enter values of elements:-");

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

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

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

}}}

void print(int n,int m,int \*\*a)

{ int i,j=0;

printf("\nValue of elements are:-\n");

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

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

{ printf("%d \t",\*(a[i]+j));}

printf("\n");}}

void add(int n,int m,int p,int q,int \*\*a,int \*\*b)

{ int c[10][10];

int i,j;

printf("\nAddition is:-\n");

if(n==p&&m==q)

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

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

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

printf("%d \t",c[i][j]);

}

printf("\n");

}

}

else

printf("Matrices can't be added check row column values\n");}

void sub(int n,int m,int p,int q,int \*\*a,int \*\*b)

{ int c[10][10];

int i,j;

printf("\nSubtraction is:-\n");

if(n==p&&m==q)

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

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

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

printf("%d \t",c[i][j]);

}

printf("\n");

}

}

else

printf("Matrices can't be subtracted check row column values\n");

}

void mul(int n,int m,int p,int q,int \*\*a,int \*\*b)

{ int c[10][10];

int i,j,k;

printf("\nMultiplication is:-\n");

if(m==p)

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

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

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

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

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

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

{ printf("%d \t",c[i][j]);}

printf("\n"); } }

else

{ printf("Matrices can't be multiplied check row column values\n");}}

void trs(int n,int m,int \*\*a)

{ int c[10][10];

int i,j=0;

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

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

{ if(i==j)

{ c[i][j]=a[i][j];}

else

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

} }

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

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

{ printf("%d \t",c[i][j]);}

printf("\n");}

}

void sdp(int n,int m,int \*\*a)

{ int i,j,r,c;

int max,min,max1,min1;

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

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

{ max=a[i][j];

min=a[i][j];

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

{ if(min>a[i][r])

{ min=a[i][r];}}

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

{ if(max<a[c][j])

{ max=a[c][j];}}

if(max==min){ printf("Sdp is %d",max);

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

{for(c=0;c<m;c++)

{ if(a[r][c]==max)

{ printf(" at row %d and column %d \n ",r,c);}}}}

} }

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

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

{ max1=a[i][j];

min1=a[i][j];

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

{ if(max1<a[i][r])

{ max1=a[i][r];}}

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

{ if(min1>a[c][j])

{ min1=a[c][j];}}

if(max1==min1){ printf("\nSdp is %d",max1);

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

{for(c=0;c<m;c++)

{ if(a[i][j]==max1)

{ printf(" at row %d and column %d ",r,c);}}}}

}}}

Value for row of 1:- 2

Value for column of 1:-2

Value for row of 2:- 2

Value for column of 2:-2

matrix 1:-

Enter values of elements:-1

2

3

4

Value of elements are:-

1 2

3 4

matrix 2:-

Enter values of elements:-5

6

7

8

Value of elements are:-

5 6

7 8

enter 1 addition of matrices , 2 for subtraction,3 for multiplication,4 for transpose,5 for sdp,0 for exit

Enter choice:

1

Addition is:-

6 8

10 12

enter 1 addition of matrices , 2 for subtraction,3 for multiplication,4 for transpose,5 for sdp,0 for exit

Enter choice:

2

Subtraction is:-

-4 -4

-4 -4

enter 1 addition of matrices , 2 for subtraction,3 for multiplication,4 for transpose,5 for sdp,0 for exit

Enter choice:

3

Multiplication is:-

14 16

28 32

enter 1 addition of matrices , 2 for subtraction,3 for multiplication,4 for transpose,5 for sdp,0 for exit

Enter choice:

4

enter matrix for transpose:-1

1 2

3 4

enter 1 addition of matrices , 2 for subtraction,3 for multiplication,4 for transpose,5 for sdp,0 for exit

Enter choice:

5

enter matrix for sdp:-1

Sdp is 3 at row 1 and column 0

Sdp is 2 at row 0 and column 1 enter 1 addition of matrices , 2 for subtraction,3 for multiplication,4 for tran

ose,5 for sdp,0 for exit

Enter choice:

0

...Program finished with exit code 0

Press ENTER to exit console.