The 2D array is organized as matrices which can be represented as the collection of rows and columns.

**import** **static** java.lang.System.***out***;

**import** java.util.\*;

**public** **class** Myclass {

**public** **static** **void** main(String[] args) {

**int** i,j;

Scanner sc=**new** Scanner(System.***in***);//this class has nextInt method

***out***.print("Enter numer of row and colomn you want");

|  |  |  |
| --- | --- | --- |
| 0 | 1 | 2 |

**int** r=sc.nextInt();//3//take input (as int)

**int** c=sc.nextInt();//3// take input (as int)

|  |  |  |
| --- | --- | --- |
| 1 | 2 | 3 |
| 4 | 5 | 6 |
| 7 | 8 | 9 |

Heap

|  |  |
| --- | --- |
| 0 |  |
| 1 |  |
| 2 |  |

**int**[][] arr=**new** **int**[r][c];

arr

**for**(i=0;i<r;i++)

2 {

2000

**for**(j=0;j<c;j++)

arr[i][j]=sc.nextInt();

}

**for**(i=0;i<r;i++)

|  |  |  |
| --- | --- | --- |
| 0 | 1 | 2 |
| 1 | 2 | 3 |

{

**for**(j=0;j<c;j++)

arr

|  |
| --- |
| 0=4000 |
| 1=5000 |
| 2=6000 |

***out***.prin***t***(arr[i][j]);

|  |  |  |
| --- | --- | --- |
| 4 | 5 | 6 |

***out***.println();

}

2000

|  |  |  |
| --- | --- | --- |
| 7 | 8 | 9 |

}

}

H E A P

Display data of following index

For(i=0;i<3;i++)

For(j=0;j<3-i;j++)//3 2 arr[i][j]

Data in matrices

|  |  |  |  |
| --- | --- | --- | --- |
|  | 0 | 1 | 2 |
| 0 | 1 | 2 | 3 |
| 1 | 4 | 5 | 6 |
| 2 | 7 | 8 | 9 |

123

00 01 02

10 11

20

45

For(i=0;i<3;i++)

For(j=0;j<3-i;j++)//3 2

Arr[i][j]

Or

For(i=0;i<3;i++)

For(j=0;j<3;j++)//3 2

**If(i+j<=2)**

Arr[i][j]

7

for(i=0;i<3;i++)

<3-0=3 k=3

<3-1=2 k--

<3-2=1

{

for(j=0;j<3-i;j++)

out.print(,arr[i][j]);

out.print("\n");

}

Display data of following index

i>=j

Data in matrices

123

56

9

00 01 02

11 12

22

00

10 11

20 21 22

|  |  |  |  |
| --- | --- | --- | --- |
|  | 0 | 1 | 2 |
| 0 | 1 | 2 | 3 |
| 1 | 4 | 5 | 6 |
| 2 | 7 | 8 | 9 |

1

45

789

3

5 6

7 8 9

For(i=0;i<3;i++)

For(j=0;j<3;j++)//3 2

**If(i>=j)**

Arr[i][j]

for(i=0;i<3;i++)

{

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

out.print(arr[i][j]);//00 //10 11 // 20 21 22

02

11 12

20 21 22

out.print("\n");

}

TASK🡺

Display data of following index

Data in matrices

|  |  |  |  |
| --- | --- | --- | --- |
|  | 0 | 1 | 2 |
| 0 | 1 | 2 | 3 |
| 1 | 4 | 5 | 6 |
| 2 | 7 | 8 | 0 |

123

00 01 02

11 12

22

56

9

for(i=0;i<3;i++)

{

for(j=i ;j<3;j++)

out.print(,arr[i][j]);

out.print("\n");

}

TASK🡺

Display data of following index

Data in matrices

|  |  |  |  |
| --- | --- | --- | --- |
|  | 0 | 1 | 2 |
| 0 | 1 | 2 | 3 |
| 1 | 4 | 5 | 6 |
| 2 | 7 | 8 | 0 |

3

56

02

11 12

20 21 22

789

For(i=0;i<3;i++)

For(j=0;j<3;j++)//3 2

**If(i+j>=2)**

arr[i][j]

else

“ “

J=2 i=0 == j=2-i

1 i=1

0 i=2

for(i=0;i<3;i++)

{

for(j=2-i ;j<3;j++)

***out***.print (arr[i][j]);//02// 11 12🡺j=1//20 21 22

***out***.print ("\n");

}

Display data of following index

Data in matrices

|  |  |  |  |
| --- | --- | --- | --- |
|  | 0 | 1 | 2 |
| 0 | 1 | 2 | 3 |
| 1 | 4 | 5 | 6 |
| 2 | 7 | 8 | 9 |

1

00

11

22

5

9

For(i=0;i<3;i++)

For(j=0;j<3;j++)//3 2

**If(i==j)**

arr[i][j]

else

out.print(“ “)

for(i=0;i<3;i++)

out.print(arr[i][i]);//00 11 22 🡺1 5 9

Display data of following index

Data in matrices

|  |  |  |  |
| --- | --- | --- | --- |
|  | 0 | 1 | 2 |
| 0 | 1 | 2 | 3 |
| 1 | 4 | 5 | 6 |
| 2 | 7 | 8 | 0 |

3

02

11

20

5

7

For(i=0;i<3;i++)

For(j=0;j<3;j++)//3 2

**If(i+j==2)**

arr[i][j]

else

out.print(“ “)

for(i=0;i<3;i++)

02 [ 2-0]

11 [2-1]

2 0 [2-2]

out.print(arr[i][2-i]);

You have data in 3\*3 matrix you have to store row wise sum and column wise sum in to another array.

**import** **static** java.lang.System.***out***;

**import** java.util.\*;

**public** **class** Myclass {

**public** **static** **void** main(String[] args) {

**int** i,j;

|  |  |  |  |
| --- | --- | --- | --- |
|  | 0 | 1 | 2 |
| 0 | 1 | 2 | 3 |
| 1 | 4 | 5 | 6 |
| 2 | 7 | 8 | 9 |

Scanner sc=**new** Scanner(System.***in***);

;

arr

**int** []r=**new** **int**[3];

**int** []c=**new** **int**[3];

2000

**int** sum;

**int**[][] arr=**new** **int**[3][3];

6 0 15 0 24

**for**(i=0;i<3;i++)

{

sum

**for**(j=0;j<3;j++)

arr[i][j]=sc.nextInt();

}

|  |  |  |
| --- | --- | --- |
| 0 | 1 | 2 |
| 6 | 15 | 24 |

//row wise sum

r

**for**(i=0;i<3;i++)

{ sum=0;

8000

**for**(j=0;j<3;j++)

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

r[i]=sum;

|  |  |  |
| --- | --- | --- |
| 0 | 1 | 2 |
| 12 | 15 | 19 |

}

c

//column wise sum

**for**(i=0;i<3;i++)

8000

{ sum=0;

**for**(j=0;j<3;j++)

sum=sum+arr[j][i];//00 10 20| 01 11 21| 02 12 22

c[i]=sum;

}

//print row wise sum

**for**(i=0;i<3;i++)

***out***.println(r[i]);

//print col. wise sum

**for**(i=0;i<3;i++)

***out***.println(c[i]);

}

}

|  |  |  |
| --- | --- | --- |
| 0 | 1 | 2 |

Transpose to another matrices

**import** **static** java.lang.System.***out***;

a

|  |  |  |
| --- | --- | --- |
| 1 | 2 | 3 |
| 4 | 5 | 6 |
| 7 | 8 | 9 |

|  |
| --- |
| 0 |
| 1 |
| 2 |

**import** java.util.\*;

**public** **class** Myclass {

**public** **static** **void** main(String[] args) {

**int**[][] a={{1,2,3},{4,5,6},{7,8,9}};

**int**[][] b=**new** **int**[3][3];

**int** i,j;

|  |  |  |
| --- | --- | --- |
| 0 | 1 | 2 |

**for**(i=0;i<3;i++)

{

b

**for**(j=0;j<3;j++)

|  |
| --- |
| 0 |
| 1 |
| 2 |

|  |  |  |
| --- | --- | --- |
| 1 | 4 | 7 |
| 2 | 5 | 8 |
| 3 | 6 | 9 |

{

b[**j][i**] = a[i][j];

}

}

**for**(i=0;i<3;i++)

{

**for**(j=0;j<3;j++)

{***out***.print(b[i][j]);

}

***out***.print("\n");

}

}

}

01 I j==<i<j

02

10

12 20

21

Transpose original matrix

**import** **static** java.lang.System.***out***;

for(i=0;i<3;i++){

for(j=i+1;j<3;j++) 01 02 12

{ t=a[i][j];

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

a[j][i]=t;

} }

**import** java.util.\*;

**public** **class** Myclass {

**public** **static** **void** main(String[] args) {

**int**[][] a={{1,2,3},{4,5,6},{7,8,9}};

**int**[][] b=**new** **int**[3][3];

|  |  |  |
| --- | --- | --- |
| 0 | 1 | 2 |

**int** i,j;

**int** t;

6

**for**(i=0;i<3;i++)

|  |  |  |
| --- | --- | --- |
| 1 | 2 4 | 3 7 |
| 4 2 | 5 | 6 8 |
| 7 3 | 8 6 | 9 |

{

10

20

21

i>j

|  |
| --- |
| 0 |
| 1 |
| 2 |

**for**(j=0;j<3;j++)//01 02 12

{

**if**(i<j)0 0 0 1 0 2 //1 0 11 12

{

t=a[i][j];01

a[i][j]=a[j][i]; 10 20 21

a[j][i]=t;

}

}

}

**for**(i=0;i<3;i++)

{

**for**(j=0;j<3;j++)

{***out***.print(a[i][j]);

}

***out***.print("\n");

}

}

}

Magic square

|  |  |  |
| --- | --- | --- |
| 0 | 1 | 2 |

#include<stdio.h>

|  |  |  |
| --- | --- | --- |
| 1 | 2 | 3 |
| 4 | 5 | 6 |
| 7 | 8 | 9 |

int main()

|  |
| --- |
| 0 |
| 1 |
| 2 |

{ R C

int arr[3][3];

out.print("enter data in 3\*3 matrix")

for(i=0;i<3;i++)

{

for(j=0;j<3;j++)

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

}

for(i=0;i<3;i++)

{

for(j=0;j<3;j++)

out.print(,arr[i][j]);

|  |  |  |
| --- | --- | --- |
| 0 | 1 | 2 |

out.print("\n");

}

|  |  |  |
| --- | --- | --- |
| 1 | 2 | 3 |
| 4 | 5 |  |
| 7 |  |  |

|  |
| --- |
| 0 |
| 1 |
| 2 |

/\*1 2 3

4 5

7

\*/

for(i=0;i<3;i++)

<3-0=3 k=3

<3-1=2 k--

<3-2=1

{

for(j=0;j<3-i;j++)

out.print(,arr[i][j]);

|  |  |  |
| --- | --- | --- |
| 0 | 1 | 2 |

out.print("\n");

|  |  |  |
| --- | --- | --- |
| 1 |  |  |
| 4 | 5 |  |
| 7 | 8 | 9 |

}

|  |
| --- |
| 0 |
| 1 |
| 2 |

/\*1

4 5

7 8 9

\*/

for(i=0;i<3;i++)

{

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

out.print(,arr[i][j]);//00 //10 11 // 20 21 22

out.print("\n");

}

3

56

789

/\* 3

|  |  |  |
| --- | --- | --- |
| 0 | 1 | 2 |

j

56

|  |  |  |
| --- | --- | --- |
|  |  | 3 |
|  | 5 | 6 |
| 7 | 8 | 9 |

|  |
| --- |
| 0 |
| 1 |
| 2 |

789

i

\*/

for(i=0;i<3;i++)

{

for(j=2-i ;j<3;j++)

{

out.print(,arr[i][j]);

out.print("\n");

}

123

56

9

/\*

|  |  |  |
| --- | --- | --- |
| 0 | 1 | 2 |

123

56

|  |  |  |
| --- | --- | --- |
| 1 | 2 | 3 |
|  | 5 | 6 |
|  |  | 9 |

|  |
| --- |
| 0 |
| 1 |
| 2 |

9

\*/

for(i=0;i<3;i++)

{

for(j=i ;j<3;j++)

out.print(,arr[i][j]);

out.print("\n");

|  |  |  |
| --- | --- | --- |
| 0 | 1 | 2 |

|  |  |  |
| --- | --- | --- |
| 1 |  |  |
|  | 5 |  |
|  |  | 9 |

|  |
| --- |
| 0 |
| 1 |
| 2 |

}

159

/\* 1

5

9\*/

for(i=0;i<3;i++)

out.print(,arr[i][i]);

/\*

|  |  |  |
| --- | --- | --- |
| 0 | 1 | 2 |

3

02 [ 2-0]

11 [2-1]

2 0 [2-2]

5

357

|  |  |  |
| --- | --- | --- |
|  |  | 3 |
|  | 5 |  |
| 7 |  |  |

|  |
| --- |
| 0 |
| 1 |
| 2 |

7

\*/

for(i=0;i<3;i++)

out.print(,arr[i][2-i]);

|  |  |  |
| --- | --- | --- |
| 1 | 2 | 3 |
| 4 | 5 | 6 |
| 7 | 8 | 9 |

|  |  |  |
| --- | --- | --- |
| 7 | 4 | 1 |
| 8 | 5 | 2 |
| 9 | 6 | 3 |

Rotate matrix clock wise 90.

|  |  |  |
| --- | --- | --- |
| 0 | 1 | 2 |

|  |  |  |
| --- | --- | --- |
| 1 | 2 | 3 |
| 4 | 5 | 6 |
| 7 | 8 | 9 |

|  |
| --- |
| 0 |
| 1 |
| 2 |

4. Column wise sum

#include<stdio.h>

int main()

{

int a[3][3]={1,2,3,4,5,6,7,8,9};

int i,j,sum=0;

int s[3];

a

for(i=0;i<3;i++)

{

sum=0;

for(j=0;j<3;j++)

j i

00 0 1 02

10 11 12

20 21 22

s

|  |  |  |
| --- | --- | --- |
| 12 | 15 | 18 |

{

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

}

s[i]=sum;

0 1 5 12 0 2 7 15 0 3 9 18

}

sum

for(i=0;i<3;i++)

out.print("%d\n",s[i]);

getch();

}

Transpose in another Matrix

#include<stdio.h>

int main()

{

int a[3][3]={1,2,3,4,5,6,7,8,9};

int b[3][3];

a

int i,j,temp;

|  |  |  |
| --- | --- | --- |
| 0 | 1 | 2 |

|  |  |  |
| --- | --- | --- |
| 1 | 2 | 3 |
| 4 | 5 | 6 |
| 7 | 8 | 9 |

|  |
| --- |
| 0 |
| 1 |
| 2 |

for(i=0;i<3;i++)

{

for(j=0;j<3;j++)

{

b[j][i] = a[i][j]

}

}

b

for(i=0;i<3;i++)

|  |  |  |
| --- | --- | --- |
| 0 | 1 | 2 |

|  |  |  |
| --- | --- | --- |
| 1 | 4 | 7 |
| 2 | 5 | 8 |
| 3 | 6 | 9 |

|  |
| --- |
| 0 |
| 1 |
| 2 |

{

for(j=0;j<3;j++)

{

out.print("%d ",b[i][j]);

}

out.print("\n");

}

|  |  |  |  |
| --- | --- | --- | --- |
| 34 | 34 | 34 | 34 |

getch();

}

|  |  |  |  |
| --- | --- | --- | --- |
| 16 | 3 | 2 | 13 |
| 5 | 10 | 11 | 8 |
| 9 | 6 | 7 | 12 |
| 4 | 15 | 14 | 1 |

|  |
| --- |
| 34 |
| 34 |
| 34 |
| 34 |

|  |  |  |  |
| --- | --- | --- | --- |
| 0 | 1 | 2 | 3 |

|  |
| --- |
| 0 |
| 1 |
| 2 |
| 3 |

Majic square

int i,j,sum,r[4],c[4],sum1,d1,d2,flag=0;

int arr [4][4] =

{16,3,2,13,5,10,11,8,9,6,7,12,4,15,14,1};

d1=d2=0;

for(i=0;i<4;i++)

{

sum=0;

|  |  |  |  |
| --- | --- | --- | --- |
| 34 | 34 | 34 | 34 |

sum1=0;

for(j=0;j<4;j++)

{ sum=sum+arr[i][j];

Sum1

sum

sum1=sum1+arr[j][i];

}

34

34

r[i]=sum;

c[i]=sum1;

}

|  |  |  |  |
| --- | --- | --- | --- |
| 34 | 34 | 34 | 34 |

for(i=0;i<4;i++)

r

{ d1=d1+arr[i][i];

d2=d2+arr[i][3-i];

}

if(d1==d2)

c

|  |  |  |  |
| --- | --- | --- | --- |
| 34 | 34 | 34 | 34 |

{ for(i=0;i<4;i++)

{

if(r[i]==c[i]&&r[i]==d1)

{flag=1;

}

else

{flag=0;

D2

D1

break;}

}

if(flag==0)

out.print("not a magic square");

34

34

else

out.print("It is a magic square");

r

|  |  |  |  |
| --- | --- | --- | --- |
| 34 | 34 | 34 | 34 |

c

|  |  |  |  |
| --- | --- | --- | --- |
| 34 | 34 | 34 | 34 |

Q🡺Accept data in 3\*3 matrix in two set and do the sum of it and store it in a 3rd matrix

|  |  |  |
| --- | --- | --- |
| 1 | 2 | 3 |
| 4 | 5 | 6 |
| 7 | 8 | 9 |

|  |  |  |
| --- | --- | --- |
| 10 | 11 | 12 |
| 13 | 14 | 15 |
| 16 | 17 | 18 |

|  |  |  |
| --- | --- | --- |
| 11 | 13 | 15 |
| 17 | 19 | 23 |
| 23 | 25 | 31 |

Q🡺 Accept data in 3\*3 matrix in two set and do the multiplication of 2 matrix;

10+11+12🡺33

00\*00 +00\*01+00\*02=

|  |  |  |
| --- | --- | --- |
| 33 | 13 | 15 |
| 17 | 19 | 23 |
| 23 | 25 | 31 |