**Assignment Number:02**

**Subject:Data Structure And Algorithms**

**Name: Shrirang Mhalgi**

**Roll NO.:222006**

**Class:S.E.**

**Division:B**

**Batch:B1**

* **PROBLEM STATEMENT:-**

An m x n matrix is said to have a saddle point if some entry a[i][j] is the smallest value in row i and the largest value in j. Write C/ C++ function that determines the location of a saddle point if one exists.

**CODE**

#include<iostream>

#include<limits>

using namespace std;

class matr

{

private: int mat[10][10],i,j,row,col,min,max,col\_index,row\_index; //data memebers

public:

matr()

{

i=j=row=col=row\_index=col\_index=0;

min=max=0;

}

void getdata();

void show ();

void saddlepoint();

};

void matr::getdata()

{

cout<<"\n Enter How many Rows in matrix: ";

for (;;)

{

if (cin >> row)

{

break;

}

else

{

cout << "Please enter a valid integer" << endl;

cin.clear();

cin.ignore(numeric\_limits<streamsize>::max(), '\n');

}

}

cout<<"\n Enter How many cols in matrix: ";

for (;;)

{

if (cin >> col)

{

break;

}

else

{

cout << "Please enter a valid integer" << endl;

cin.clear();

cin.ignore(numeric\_limits<streamsize>::max(), '\n');

}

}

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

{

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

{

cout<<"\nEnter matrix mat["<<i<<"]["<<j<<"] element : ";

cin>>mat[i][j];

}

}

}

void matr::show ()

{

cout<<"\nThe Elements inside the matrix are as follows...\n\n";

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

{

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

{

cout<<"\t"<<mat[i][j];

}

cout<<"\n";

}

cout<<"\n";

}

void matr::saddlepoint()

{

int big[5],small[5];

for(i=0;i<5;i++)

{

big[i]=small[i]=0;

}

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

{

small[i]=mat[i][0];

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

{

if(mat[i][j]<=small[i])

{

small[i]=mat[i][j];

}

}

}

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

{

big[j]=mat[0][j];

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

{

if(mat[i][j]>=big[j])

{

big[j]=mat[i][j];

}

}

}

max=small[0];

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

{

if(small[i]>=max)

{

max=small[i];

row\_index = i;

}

}

min=big[0];

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

{

if(big[j]<=min)

{

min=big[j];

col\_index = j;

}

}

if(min==max)

{

cout<<"\nSaddle point is "<<min<<endl;

cout<<"At Position:- "<<row\_index+1<<","<<col\_index+1<<endl;

}

else

{

cout<<"\nSaddle point does not exist in the Matrix!\n\n";

}

}

int main()

{

int ch,c;

matr s1;

do

{

cout<<"1.accept , show matrix"<<"2.saddle point"<<endl<<"0.exit"<<endl;

cin>>ch;

switch(ch)

{

case 1:

s1.getdata();

s1.show ();

break;

case 2:

s1.saddlepoint();

break;

default:

cout<<"!!!INVALID!!!"<<endl;

break;

}

cout<<"press any no. except zero"<<endl;

cin>>c;

}while(c!=0);

return 0;

}

/\*

OUTPUT-:

1.accept , show matrix

2.saddle point

0.exit

1

Enter How many Rows in matrix: 3

Enter How many cols in matrix: 3

Enter matrix mat[0][0] element : 1

Enter matrix mat[0][1] element : 2

Enter matrix mat[0][2] element : 3

Enter matrix mat[1][0] element : 4

Enter matrix mat[1][1] element : 5

Enter matrix mat[1][2] element : 6

Enter matrix mat[2][0] element : 7

Enter matrix mat[2][1] element : 8

Enter matrix mat[2][2] element : 9

The Elements inside the matrix are as follows...

1 2 3

4 5 6

7 8 9

press any no. except zero

1

1.accept , show matrix

2.saddle point

0.exit

2

Saddle point is 7

At Position:- 3,1

press any no. except zero

1

1.accept , show matrix

2.saddle point

0.exit

1

Enter How many Rows in matrix: 3

Enter How many cols in matrix: 3

Enter matrix mat[0][0] element : 1

Enter matrix mat[0][1] element : 0

Enter matrix mat[0][2] element : 0

Enter matrix mat[1][0] element : 0

Enter matrix mat[1][1] element : 1

Enter matrix mat[1][2] element : 0

Enter matrix mat[2][0] element : 0

Enter matrix mat[2][1] element : 0

Enter matrix mat[2][2] element : 1

The Elements inside the matrix are as follows...

1 0 0

0 1 0

0 0 1

press any no. except zero

1

1.accept , show matrix

2.saddle point

0.exit

2

Saddle point does not exist in the Matrix!

press any no. except zero

0

\*/