Assignment No.:	2
Title:	To find out the saddle point.
Subject:	Data Structures Laboratory.
Class:	S.Y. 13 (BTech) C.S.E
Roll No.:	2215055(MITU21BTIT0010).
Assessment (Marks):	
Signature and Date of Assessment:	

Experiment No. 2

* Experiment Title :- Implement C++ program for finding out the saddle point in a matrix.

* Objectives :-
a) To understand the 2D Matrix in different applications.
b) To understand the concept of saddle point.

* Problem Statement :- A matrix of m x n order is said to have a saddle point, if some entry at a[i][j] is the smallest value in row i and the largest value in j. Write C++ function that determines the location of a saddle point if it exists.

* Outcomes :-
1) Understanding the use of 2D matrix.
2) Understanding the use of saddle point in various real world application.
g++ compiler on Ubuntu 14.04 (64 bits)

* Saddle point :- A saddle point of a matrix is an element which is both the largest element in its column and the smallest element in its row.

* → Saddle point coordinates in a matrix

Maximum element column-wise ↓

1	2	3
4	5	6
7	8	9

→ Minimum element row-wise.

* Algorithm (To find saddle point)

1) Start
2) Define a matrix and its respective row and column numbers.
3) Read no of rows & columns (size of matrix) to create a matrix.
4) Read matrix A as per specified size.
5) Use 2 for loops, outer for loop is used to read row elements & inner for loop to read column elements.
6) Find the minimum element of the current row and store the column index of the minimum element.
7) Find the maximum element of the current column and store the row index of the maximum element.

9) Check if the row minimum element is also maximum in its column.
9) Use the stored column index here.
10) If it is same, then saddle point is present else continues till the end of the matrix.
11) Finally, display the saddle point element if present.
12) Stop.

* Conclusion:- Thus, we implemented concept of saddle point in 2D matrix.

* Questions:-

1) What is the saddle point?

→ A saddle point of a matrix is an element which is both the largest element in its column and the smallest element in its row.

2) What are the applications of saddle point?

→ The application of the saddle point method for the evaluation of the probability density function of the decision variable at the receiver of a pre-amplified system in the presence of intrachannel crosstalk is investigated.

Process returned 0 (0x0) execution time : 30.513

s Press any key to continue.

```
/**

ASSIGNMENT NO:- 2

problem statement

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

Author: Anurag Haldey.

Date: 25/08/2022.

**/


#include<iostream>

using namespace std;


class saddlemat

{

  int mat[10][10],i,j,row,col,min,max;

  public:

  saddlemat()

  {

    i=j=row=col=0;

    min=max=0;

  }
```

```
  void getdata();
  void showmat();
  void saddlepoint();
};

void saddlemat::getdata()
{
   cout<<"\n enter how many rows in matrix: ";
   cin>>row;
   cout<<"\n enter how many columns in matrix: ";
   cin>>col;
   for(i=0;i<row;i++)
   {
      for(j=0;j<col;j++)
      {
         cout<<"\n enter matrix mat["<<i<<"]["<<j<<"]element: ";
         cin>>mat[i][j];
      }
   }
}

void saddlemat::showmat()
{
   cout<<"\n the 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 saddlemat::saddlepoint()
{
    int greatest[5],smallest[5];
    for(i=0;i<5;i++)
    {
        greatest[i]=smallest[i]=0;
    }
    for(i=0;i<row;i++)
    {
        smallest[i]=mat[i][0];
        for(j=0;j<col;j++)
        {
            if(mat[i][j]<=smallest[i])
            {
                smallest[i]=mat[i][j];
                cout<<"\n smallest"<<"["<<i<<"]"<<smallest[i];
```

```
      }
    }
}
for(j=0;j<col;j++)
{
   greatest[j]=mat[0][j];
   for(i=0;i<row;i++)
   {
      if(mat[i][j]>=greatest[j])
      {
         greatest[j]=mat[i][j];
         cout<<"\n
      greatest"<<"["<<j<<"]"<<greatest[j]; }
    }
 }
 max=smallest[0];
 for(i=0;i<row;i++)
 {
    if(smallest[i]>=max)
    {
       max=smallest[i];
    }
 }
 min=greatest[0];
 for(j=0;j<col;j++)
 {
```

```
    if(greatest[j]<=min)
    {
       min=greatest[j];
    }
  }
  if(min==max)
  {
     cout<<"\n Saddle Point is present in this array\n";
     cout<<"SaddlePoint is : "<<min;
   }
   else
   {
      cout<<"\n SaddlePoint is not present in this
    matrix.\n\n"; }
}
int main()
{
    saddlemat s1;
    s1.getdata();
    s1.showmat();
    s1.saddlepoint();
    return 0;
}
```

Output:

enter how many rows in matrix: 3
enter how many columns in matrix:

3 enter matrix mat[0][0]element: 1

enter matrix mat[0][1]element: 3

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

enter matrix mat[1][0]element: 5

enter matrix mat[1][1]element: 4

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

enter matrix mat[2][0]element: 8

enter matrix mat[2][1]element: 9

enter matrix mat[2][2]element: 7

the elements inside the matrix are as follows..

1 3 2

5 4 6

8 9 7

smallest[0]1

smallest[1]5

smallest[1]4

smallest[2]8

smallest[2]7

greatest[0]1

greatest[0]5

greatest[0]8

greatest[1]3

greatest[1]4

greatest[1]9

greatest[2]2

greatest[2]6

greatest[2]7

Saddle Point is present in this array

SaddlePoint is : 7

Process returned 0 (0x0) execution time : 7.450

s Press any key to continue.

/**

ASSIGNMENT NO-: 3

problem statement

A set S={1,3, a, s, t, i} represent alphanumeric characters. Write a program in C++ to generate all possible passwords of length.

Author: Anurag Haldey.

Date: 12/09/2022.

**/