**Name: Session:**

**Programming II**

**2D Arrays**

**Lab Exercise 4/29/2020**

In this lab you will create an application that uses a 2D array.

Theater Seating. Write a program that can be used to sell tickets for performances. The auditorium has 15 rows of seats with 30 seats in each row. The program should display a screen that shows which seats are available and which are taken. For example:

Seats

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30

Row 1 \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \*

Row 2 \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \*

Row 3 \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \*

Row 4 \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \*

Row 5 \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* X \* \* \* \* \* \* \* \*

Row 6 \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \*

Row 7 \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \*

Row 8 \* \* \* \* \* \* X \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \*

Row 9 \* \* \* \* \* \* \* \* \* \* \* \* X \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \*

Row 10 \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \*

Row 11 \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* X \* \* \* \* \* \* \* \* \* \*

Row 12 \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \*

Row 13 \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \*

Row 14 \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \*

Row 15 \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \*

\* = open seat X = seat taken

Allow the program user to input which seat they would like to reserve. If the seat is taken, they should get a message denying them the seat. If the seat is reserved, the status of the seating should be updated.

Note: In the working application I replaced the untaken seat symbol with a - and the taken seat symbol with a !. This is done to prevent problems with a property called kerning (an X symbol is wider than a \* symbol) which distorts the display.

1. Add the following 2D array declaration as a global variable.

char [,] seats = new char[15, 30];

1. Add the following code to the Form1\_Load event handler.

string message = "";

for (int r = 1; r <= 15; r++)

message += "Row " + r + Environment.NewLine;

lblRow.Text = message;

for (int row = 0; row < 15; row++)

for (int col = 0; col < 30; col++)

seats[row, col] = '-';

display();

1. Add the following code to the display() function

string message = "";

for (int row = 0; row < 15; row++)

{

for (int col = 0; col < 30; col++)

message += seats[row, col] + " ";

message += Environment.NewLine;

}

lblChart.Text = message;

1. Add the following code to the btnReserve\_Click event handler.

int row, col;

//Get row and column from textboxes

row = Convert.ToInt32(txtRow.Text);

col = Convert.ToInt32(txtCol.Text);

//Check to see if seat already reserved

//If already taken display message box and leave function

if (seats[row - 1, col - 1] == '!')

{

MessageBox.Show("Seat Already taken");

return;

}

//Assign reserved character to array

seats[row - 1, col - 1] = '!';

//Display the array

display();

//Reset the textboxes and put focus on first textbox

txtCol.Text = "";

txtRow.Text = "";

txtRow.Focus();

1. Test your application

**When you have completed your application, print a screenshot of your running application and turn in.**