**Project (**char Arrays)

**Tic Tac Toe**

You have to make N x N Board game where n will be declare and initialize at the start of main.

N can be any positive integer. So make your logical generic for any positive integer>=3

**Players**: 2 players **X** and **O**

**Setup**

Draw a board, creating either N x N grid of squares on console.

Here are the examples:

**Playing on a 3x3 Board**

* X always goes first.
* Players alternate placing Xs and Os on the board until either (a) one player has three in a row, horizontally, vertically or diagonally; or (b) all nine squares are filled.
* If a player is able to draw three Xs or three Os in a row, that player wins.
* If all nine squares are filled and neither player has three in a row, the game is a draw.
* Also handle wrong inputs/ already filled boxes

**Playing on a 4x4 Board**

* X always goes first.
* Players alternate placing Xs and Os on the board until either (a) one player has four in a row, horizontally, vertically or diagonally; or (b) all 16 squares are filled.
* If a player is able to draw four Xs or four Os in a row, that player wins.
* If all 16 squares are filled and neither player has four in a row, the game is a draw.
* Also handle wrong inputs/ already filled boxes

**CODE:**

Make 1D char array of size (n\*n)

For example for 3x3 boards your array will have 9 elements.

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

For user it will look like this.

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

Player X your turn:

Please enter box number: 5

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

Player O your turn:

Please enter box number: 9

|  |  |  |
| --- | --- | --- |
| 1 | 2 | 3 |
| 4 | X | 6 |
| 7 | 8 | O |

It’s up to you. You can also ask row and column number from user.

You have to make functions according to your need. All functions should be generic.

Some of basic functions can be…

PlayerTurn(player name) //player name = O or X

PrintBoard()

Bool IsWin(player name)

Bool IsDraw()

Bool IsValidMove()

Your all logic should be in functions so divide your code into appropriate functions.