

Tic-Tac-Toe Game

For this exercise, you will create a Tic-Tac-Toe game for two players. An optional enhancement would be to create a Computer opponent.

Program Requirements

While this program is an introduction to lists, it will also be a quick review of functions and input validation. Since we have not explicitly covered two dimensional lists, you should use a list of nine elements, one for each space on the board

The functions you need to implement are:

- 1) displayInstructions
 - a. displays the instructions
 - b. does not return anything
- 2) showBoard
 - a. pass in board
 - b. display current state
 - c. does not return anything
- 3) getMove
 - a. pass in the board
 - b. get and validate a move(in range, unoccupied square)
 - c. update the board
- 4) checkWin
 - a. pass in board
 - b. return true if win
- 5) checkDraw
 - a. pass in board
 - b. return true if draw

Your program should ask the user if they want to play another game and repeat if the response is y.

Look at the pseudocode program below to help you get started.

Example Execution

Here is a example of how your program might look when it is running:

This is a game of Tic-Tac-Toe. You will select an empty location and enter its index to select a move. The first player will be X and the second will be O.

The board is:

[1][2][3]

[4][5][6]

[7][8][9]

X, what is your move: 3

The board is:

[1][2][X]

[4][5][6]

[7][8][9]

Y, what is your move: 3

That is an illegal move:

Y, what is your move: 10

That is an illegal move:

Y, what is your move: 2

The board is:

[1][O][X]

[4][5][6]

[7][8][9]

Pseudocode of Game Loop

Your main program should look something like this: It should call the functions to do the work.

Main Program

Repeat Play Loop until no longer wants to play

Output Instructions

initialize Board

Initialize Player to X

Play Until Win or Draw

Show Board

Get Move

Check for Win

Check for Draw

If not Win or Draw

Swap Player

Say whether win or draw

Ask if want to play another game

Win is the same character horizontally, vertically, or diagonally. Draw is 9 moves without win or all spaces filled with X or O

Optional Computer Opponent

Instead of playing another person, you might want to create a computer opponent.

An easy computer player would do a random move. A smarter computer would see if it could win in one move, or block in one move, or pick some location to move to.