Homework #3

Complete By: Sunday, October 21th @ 11:59pm
Submission: submitted via Gradescope

Tic Tac Toe

Write a modular Tic Tac Toe program in Python. Your design should factor in the possibility of extending the application in several ways. To name some possibilities of ways we might augment the program by replacing modules:

Model
- Resizable Board
- n-players
- Battleship
- Scrabble
- Allowing for the duplication of boards for an AI to speculatively explore
- The ability to save and replay the history of a game

View
- ASCII – different board designs/spacing
- HTML
- Tkinter or other Graphical Display

Controller
- Enter Location as a single value
- Use mouse to click location
- Use mouse or keyboard to select buttons in a web page
- Arrows to highlight locations + enter/space to select a location
- Accept commands via packets over the network
- AI player replacing a human player

You do not need to implement any of these, these are possibilities for future development you should be aware of as you design your application. For this assignment only the basic game of tic tac toe between two players taking turns on the same machine with a basic console display representing the state of the board.

An example of running the program is as follows:

```
Type the row and column in which you wish to move.
+---+---+---+
+   |   |   +
+---+---+---+
+   |   |   +
+---+---+---+
+   |   |   +
+---+---+---+
Turn 1: Player 1 (X), choose your move: 2 2
```

```
+---+---+---+
+   |   |   +
+---+---+---+
+   | X |   +
+---+---+---+
+   |   |   +
+---+---+---+
Turn 2: Player 2 (O), choose your move: 1 1

+---+---+---+
+ O |   |   +
+---+---+---+
+   | X |   +
+---+---+---+
+   |   |   +
+---+---+---+
Turn 3: Player 1 (X), choose your move: 1 2

+---+---+---+
+ O | X |   +
+---+---+---+
+   | X |   +
+---+---+---+
+   |   |   +
+---+---+---+
Turn 4: Player 2 (O), choose your move: 1 0
Invalid move, outside board, try again: 1 2
Invalid move, position already taken, try again: 3 2

+---+---+---+
+ O | X |   +
+---+---+---+
+   | X |   +
+---+---+---+
+   | O |   +
+---+---+---+
Turn 5: Player 1 (X), choose your move: 1 3

+---+---+---+
+ O | X | X +
+---+---+---+
+   | X |   +
+---+---+---+
+   | O |   +
+---+---+---+
Turn 6: Player 2 (O), choose your move: 3 1
```

```
+---+---+---+
+ O | X | X +
+---+---+---+
+   | X |   +
+---+---+---+
+ O | O |   +
+---+---+---+
Turn 7: Player 1 (X), choose your move: 3 3

+---+---+---+
+ O | X | X +
+---+---+---+
+   | X |   +
+---+---+---+
+ O | O | X +
+---+---+---+
Turn 8: Player 2 (O), choose your move: 2 1

+---+---+---+
+ O | X | X +
+---+---+---+
+ O | X |   +
+---+---+---+
+ O | O | X +
+---+---+---+
Player 2 Wins!  Play again? (y/n): y

+---+---+---+
+   |   |   +
+---+---+---+
+   |   |   +
+---+---+---+
+   |   |   +
+---+---+---+
Turn 1: Player 1 (X), choose your move: 2 2

+---+---+---+
+   |   |   +
+---+---+---+
+   | X |   +
+---+---+---+
+   |   |   +
+---+---+---+
Turn 2: Player 2 (O), choose your move: 1 1

+---+---+---+
+ O |   |   +
+---+---+---+
+   | X |   +
+---+---+---+
+   |   |   +
+---+---+---+
Turn 3: Player 1 (X), choose your move: 1 2
```

```
+---+---+---+
+ O | X |   +
+---+---+---+
+   | X |   +
+---+---+---+
+   |   |   +
+---+---+---+
Turn 4: Player 2 (O), choose your move: 1 0
Invalid move, outside board, try again: 1 2
Invalid move, position already taken, try again: 3 2

+---+---+---+
+ O | X |   +
+---+---+---+
+   | X |   +
+---+---+---+
+   | O |   +
+---+---+---+
Turn 5: Player 1 (X), choose your move: 3 1

+---+---+---+
+ O | X |   +
+---+---+---+
+   | X |   +
+---+---+---+
+ X | O |   +
+---+---+---+
Turn 6: Player 2 (O), choose your move: 1 3

+---+---+---+
+ O | X | O +
+---+---+---+
+   | X |   +
+---+---+---+
+ X | O |   +
+---+---+---+
Turn 7: Player 1 (X), choose your move: 2 3

+---+---+---+
+ O | X | O +
+---+---+---+
+   | X | X +
+---+---+---+
+ X | O |   +
+---+---+---+
Turn 8: Player 2 (O), choose your move: 2 1
```

```
+---+---+---+
+ O | X | O +
+---+---+---+
+ O | X | X +
+---+---+---+
+ X | O |   +
+---+---+---+
Turn 9: Player 1 (X), choose your move: 3 3

+---+---+---+
+ O | X | O +
+---+---+---+
+ O | X | X +
+---+---+---+
+ X | O | X +
+---+---+---+
Tie Game!  Play again? (y/n): n
Goodbye!
```

Up to 50% of the grade will be based on design. Here are some suggestions of features of design we will be looking for. Your should design your solution in such a way to prepare for future iterations of the game changing out the input method, graphical display, or even rule set for the game. Add comments to your modular decomposition to highlight which pieces of your modules are private and which define the API that the modules you would swap out would need to implement in order for the application to continue to function. Practice tight cohesion and loose coupling when designing your modules and be sure to define the contact points where modules are joined together (the interface for each module).

This application will be a console application in either Python, "compiled" and run at the command line. The name of the file containing your "main" function should be {your netid}HW03.py.

When finished, submit a zip containing all your source file(s) in a folder called {your netid} at the base directory of the zip (put your files into a folder, then zip that folder) to Gradescope.