CS 1160 – Introduction To Computer Programming

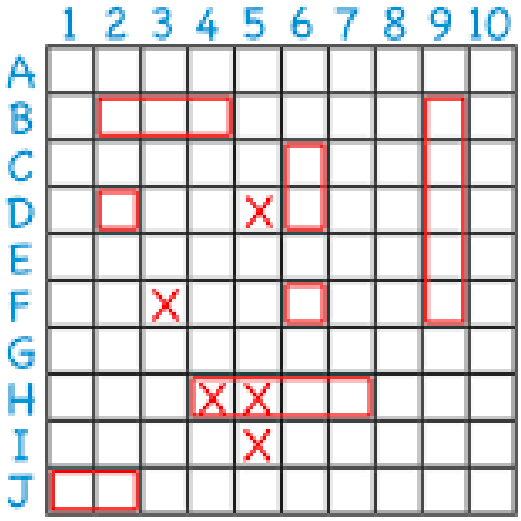
Lab 8 – Battleship

Learning Objectives

- Learn how to make large and complex programs

Overview

You must remake the classic game Battleship in Python. The game of Battleship is played by two players placing “ships” on a grid hidden from the other player. Once all ships are placed, the two players make guesses as to which places on the grid the other player has put ships. If they correctly guess the grid location, the ship will have been attacked. If all the grid locations of the ship are guessed correctly, the battleship is “sunk”. When all of a player’s ships are sunk, the player loses and the other player wins. An example of one player’s grid after some attacks from the other player are shown:



For your implementation you do not have to ask the user to place their battleships. You may hard-code your battleship placement. The battleships that the players can have are: one 5 unit ship, one 4 unit ship, one 3 unit ship, two 2 unit ships, two 1 unit ships, just like the above diagram shows. The grid must be a 10 x 10 grid.

The Program

The program must ask for two player’s names and use these names when asking a player to make a move. The program must loop, displaying the previous guesses made, asking for a row and a column to make an attack, displaying whether or not the attack was a hit or a miss, until one player wins. A player will win when the other player does not have any battleships remaining. You are required to at least use a main function. You are required to check if the column and row entered by the user is a valid number (1 through 10). A trimmed down sample output is shown below:

```
Enter name for player 1: Max
Enter name for player 2: Jack
Max's Turn!
Max's Previous Guesses:
   1 2 3 4 5 6 7 8 9 10
----------------------
1 |□ □ □ □ □ □ □ □ □ □
2 |□ □ □ □ □ □ □ □ □ □
3 |□ □ □ □ □ □ □ □ □ □
4 |□ □ □ □ □ □ □ □ □ □
5 |□ □ □ □ □ □ □ □ □ □
6 |□ □ □ □ □ □ □ □ □ □
7 |□ □ □ □ □ □ □ □ □ □
8 |□ □ □ □ □ □ □ □ □ □
9 |□ □ □ □ □ □ □ □ □ □
10|□ □ □ □ □ □ □ □ □ □
Guess a row: 1
Guess a column: 1
Good Shot! You hit your opponent's battleship!
Jack's Turn!
Jack's Previous Guesses:
   1 2 3 4 5 6 7 8 9 10
----------------------
1 |□ □ □ □ □ □ □ □ □ □
2 |□ □ □ □ □ □ □ □ □ □
3 |□ □ □ □ □ □ □ □ □ □
4 |□ □ □ □ □ □ □ □ □ □
5 |□ □ □ □ □ □ □ □ □ □
6 |□ □ □ □ □ □ □ □ □ □
7 |□ □ □ □ □ □ □ □ □ □
8 |□ □ □ □ □ □ □ □ □ □
9 |□ □ □ □ □ □ □ □ □ □
10|□ □ □ □ □ □ □ □ □ □
Guess a row: 7
Guess a column: 3
You missed your opponent's battleship!
```

Multiple turns later…

```
      Jack's Turn!
      Jack's Previous Guesses:
         1 2 3 4 5 6 7 8 9 10
      ----------------------
      1 |- □ □ □ □ □ □ □ X □
      2 |□ □ □ □ □ □ □ □ □ □
      3 |□ □ □ - □ □ □ □ □ □
      4 |□ □ □ □ □ □ □ - □ □
      5 |□ □ □ - □ □ □ - □ □
      6 |□ □ □ X X X □ □ □ □
      7 |□ □ - □ □ □ □ □ □ □
      8 |□ □ □ □ □ □ □ □ □ □
      9 |□ □ □ □ □ □ □ - □ □
      10|□ □ □ □ □ □ □ □ □ □
      Guess a row: 5
      Guess a column: 4
      You already guessed that coordinate!
      Guess a row: 5
      Guess a column: 3
      You missed your opponent's battleship!
```

Multiple turns later…

```
      Jack's Turn!
      Jack's Previous Guesses:
         1 2 3 4 5 6 7 8 9 10
      ----------------------
      1 |- □ □ □ □ □ □ □ X □
      2 |□ □ □ - □ □ □ □ □ □
      3 |□ □ □ - X □ □ □ □ □
      4 |□ □ □ - - □ □ - □ □
      5 |□ □ - - □ □ □ - □ □
      6 |□ □ □ X X X □ □ □ □
      7 |□ □ - □ □ □ □ □ □ □
      8 |□ □ □ □ □ □ □ □ □ □
      9 |□ □ □ □ □ □ □ - - □
      10|□ □ □ □ □ □ □ □ □ □
      Guess a row: 98
      Guess a column: 8
      Your input was not in the range of the grid!
      Guess a row: 9
      Guess a column: 8
      You already guessed that coordinate!
      Guess a row: 8
      Guess a column: 9
      You missed your opponent's battleship!
```

```
Multiple turns later...

      Jack's Turn!
      Jack's Previous Guesses:
         1 2 3 4 5 6 7 8 9 10
      ----------------------
      1 |- □ □ □ □ □ □ □ X □
      2 |□ □ □ - □ □ □ □ □ □
      3 |□ □ □ - X □ □ □ □ □
      4 |□ □ □ - - □ □ - □ □
      5 |□ □ - - □ □ □ - □ □
      6 |□ □ □ X X X □ □ □ □
      7 |□ □ - □ □ □ □ □ □ □
      8 |□ □ □ □ □ □ □ □ - □
      9 |□ □ □ □ □ □ □ □ - - □
      10|□ □ □ □ □ □ □ □ □ -
      Guess a row: 10
      Guess a column: 9
      You missed your opponent's battleship!
      Max's Turn!
      Max's Previous Guesses:
         1 2 3 4 5 6 7 8 9 10
      ----------------------
      1 |X □ □ □ □ □ □ □ □ X
      2 |X □ □ □ X □ □ □ □ □
      3 |X □ □ □ X - □ □ □ □
      4 |X □ □ □ X □ - □ □ X
      5 |X □ □ □ - □ □ □ □ □
      6 |□ □ X X X X □ □ □ □
      7 |□ □ □ □ □ □ □ □ X X
      8 |□ □ □ □ □ □ □ □ □ □
      9 |□ □ □ □ □ □ □ □ X □
      10|□ □ □ □ □ □ □ □ □ □
      Guess a row: 9
      Guess a column: 8
      Good Shot! You hit your opponent's battleship!
      Max Wins!
```

How to Submit

Save your .py Python program with your code and submit it to the drop box in Pilot.

Grading

This lab is worth 3.000 points, distributed as follows:

Task	Points
Successfully displayed grid with guesses each turn	1.000
Successfully gave player 1 and player 2 turns until a player won	1.000
Successfully took player input using input validation and without crashing	1.000
Total	3.000