CptS 111 – Fall 2024

Project Proposal
Due: Friday, 11.15.24, before 11:59 pm

Instructions

Fill out this form as completely as you can. It will be used by your TA to approve your project. They may ask you to revise it to ensure it meets all the requirements for the project or if they think your project is too ambitious.

1. Names of team members

   Logan Harris
   Zachary Cornthwaite

2. Project title

   Battleship in Python

3. What project will do

   Creates the popular 2 player game of battleship. Will ask the players where they will place their ships on a grid. Will switch turns between players. Will tell the user if they hit or miss. Will visualize the grid, per player, using matplotlib, and update for hits and misses. Whoever sinks the other players ships first will win.

4. Project tasks; each task represents a single function (minimum of 6 functions plus `main()`)

   main() = will start the game once called. Will prompt the users for their username (default values of player1 and player2) and do a RNG for whoever starts first (maybe will flip a coin?). Will call each function. With this said, main will call place_ships, and will have a looping function for take_turn(), status_update(), turn_time(), display_board, and will constantly update final_scoreboard() until the game is over. Will always check if ends by querying status_update file and will break the entire loop of the game once a player wins, and will call the final_scoreboard() function and will print the winner.
   place_ships() = will prompt whatever player won the coinflip to place their ships. 5 ships: carrier (5 space), battleship (4 spaces), crusier (3 spaces), submarine (3 spaces), destroyer (2 spaces). Then it will prompt the next player. Will create a 10x10 (A-J, 1-10) grid prompt to place the ships. Ships can be placed horizontally and vertically only. Each player inputs and where their ships are will be written to a file.

take_turn() = will prompt whatever player won the coinflip to enter a coordinate. Will check whether the other player has a ship tile on that coordinate. Will then return a appropiate "hit" or "miss".
status_update(): each person turn it will check where they hit or miss, will update the file that was created in place_ships(). Will check once the ships are sunk, and will return the message of "You sunk my {battle_ship_type}!"
turn_time(): will set a time per each players turn. Will be 30 seconds. If the player's clock reaches zero, it will pass their turn.
display_board(): this is a matplotlib function that will take the data in status_update() table (the file we are writing to) and will replot the board each turn depending on whose turn it is and the information that they are privy to (where they have missed and where they have hit). Will swap per turn per player.
final_scoreboard(): will keep track of all stats of the game. Some ideas are, who missed the most, how many turns who taken in the game, total game time, which player had the longest turn on average.

5. Plan for testing your code (used closed-box texting and include edge cases if appropriate)

Save your Word document as a **pdf** file and submit it via Canvas before 11:59 pm, Friday, 11.15.24.