CS 314 Final Project Proposal

Battleship

Craig Sirota and Dov Kassai

Introduction

Based on the classic guessing game, we would like to recreate the game using Python. Through the command line, players would be able to connect with their friends and play with them. We would provide the user with an authentic experience, very similar to if they were playing the game in person with their friends. Along with the game play, we would also feature a chat, allowing players to communicate with each other while playing.

Networking

Using the User Datagram Protocol, users would be able to connect and play with their friends. When they first run the game, they will be prompted with:

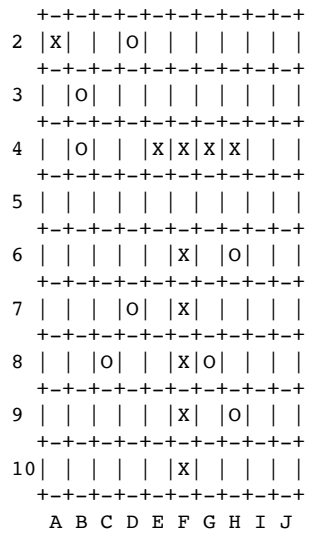
Host Game (1)
Join Game (2)

If they select to host a game, their IP address will be printed onto the screen. They would then share that address with their friend, who would select join game. Selecting join game would prompt them to input an IP address. Both programs would wait for confirmations from the other before continuing. While waiting for confirmations, we would periodically print a loading message, to assure to the user that the program has not crashed, but is in fact just waiting.

Game Play

Prior to the start of the game, the users would both be prompted to place their ships on the board. Once both players have placed their ships on their boards, the host computer would randomly select one of the players to go first. Player A will enter coordinates in the standard form, [A-J][1-10]. This will send a message to Player B's computer, requesting the state of that location (hit/miss). Player B's machine will alert Player A if the strike was a hit or miss. Player B will print a message saying that Player A has missed or hit {ship name} at {location}. Both players will also be alerted if there is a hit which results in the sinking of a ship. After the turn, both players will have the their board, and their strike map of the opponents board, printed out. On the player's board, numbers would correspond to the ship in that location, X's would mark hits, and O's would mark misses. On the strike history map, the same format would stand, the only difference would be that the unit ships would not be shown. The following is an example of what a strike map board might look like:

```
   A B C D E F G H I J
  +-+-+-+-+-+-+-+-+-+-+
1 |X| | | | | | | | | |
```



Organization

The program would contain a single player class, which would contain 5 instances of a ship class, which would hold the name, size, and location of each ship. This class would also contain a method that takes a location as a parameter and outputs whether that location was a hit or a miss. The player class would also contain two 10x10 two-dimensional lists, representing the two boards each user will have access too.