Cheree LaPierre and Shayna Snyder

CS172-1

Final Project Specification

13 December 2016

Battleship-Inspired Text Adventure Game

In battleship there are two boards—a board where you can see where your own ships are placed, and a board for guessing the locations of your opponent’s ships. Our program needs to display both of these two boards throughout the game. We will prompt the user to input coordinates that indicate where they are placing their ships. Using do-while loops, switch statements, and if statements, we need to ensure that the ships are within the board and do not overlap each other. This input will then be used to generate the first board—the one that displays the user’s own boats. Since we are making this a single-player game, the user will be playing against the “computer” or more accurately, the random number generators that we will use to place the computer’s ships and guess where the user’s ships are. The other board needs to hold, but not show, the computer’s ships and will show the players hits and misses. If the player or computer is able to sink all of the others ships, then they have won. We will have a counter for both players to keep track of how many ships each player has sunk. We will store the rules of the game in a text file that we can access through file input and ouput statements.

UML Diagram:

|  |
| --- |
| Player |
| -num: int |
| +createBoard(array[5][5]: int): void |
| +empty: int const |
| +occupied: int const |
| +missed: int const |
| +hit: int const |
| +columns, rows, repeat: int |
| +letterLocation[5]: char |
| +sunkenShips(name: string): void |
| +shipNames: string |

|  |
| --- |
| Human |
| -humanBoard[5][5], x, y, xhuman[5] yhuman[5], sunkPlayerShips = 0: int |
| +Human(string) |
| +getName(string): string |
| +printBoard(array[5][5]: int): void |
| +checkCompGuesses(xGuessC[25]: int, yGuessC[25]: int): void |
| +getSunkPlayerShips(): int |
| +getxGuessC(): int |
| +getyGuessC(): int |

|  |
| --- |
| Computer |
| -compBoard[5][5], x, y, xcomputer[5], ycopmuter[5], sunkCompShips = 0: int |
| +printBoard(array[5][5]: int): void |
| +placeShips(array[5][5]: int): void |
| +makeGuessH(array[5][5]: int): void |
| +checkPlayerGuesses(xGuessH[25]: int, yGuessH[25]: int): void |
| +getSunkCompShips(): int |
| +getxGuessH(): int |
| +getyGuessH(): int |