Project 2

CIS-17A (45434)

**Battleship**

Due Date:

July 17, 2018

Author:

Jorge Rubi

**Introduction:**

Battleship is a game where the player takes turns attempting to guess the location of their opponent’s hidden “ships” which are scattered throughout a 10x10 gridded game board, typically labeled using alphabetical letters along the Y-axis and numbers along the X-axis. The objective of the game is to be the first to find the location of all your opponent’s battleships. Once you manage to do so, you win!

**How to play:**

In this C++ version of the game, the user is the one facing off against the computer as they attempt to guess the location of the computer’s ships. The game starts off by introducing the player to the objective of the game as well as giving them a couple instructions on how they should input their guesses. Additionally, the program provides descriptions for what certain symbols represent throughout the game (ie: a “+” is an unexplored location on the board, an “\*” is a location which has been guessed but was found to have no ships aka a “miss”, and an “X” marks an explored location which happened to have a ship aka a “hit”!). The program will then present the game board to the user and will constantly update it as the user attempts to guess various locations and eventually beat the game. To conclude, the user must enter their guesses in a “B4”,”C5”,”F7” manner and they must acquire a total of 17 “X”’s or “hits” to beat the game! It’s that simple.

There are a total of

**Summary:**

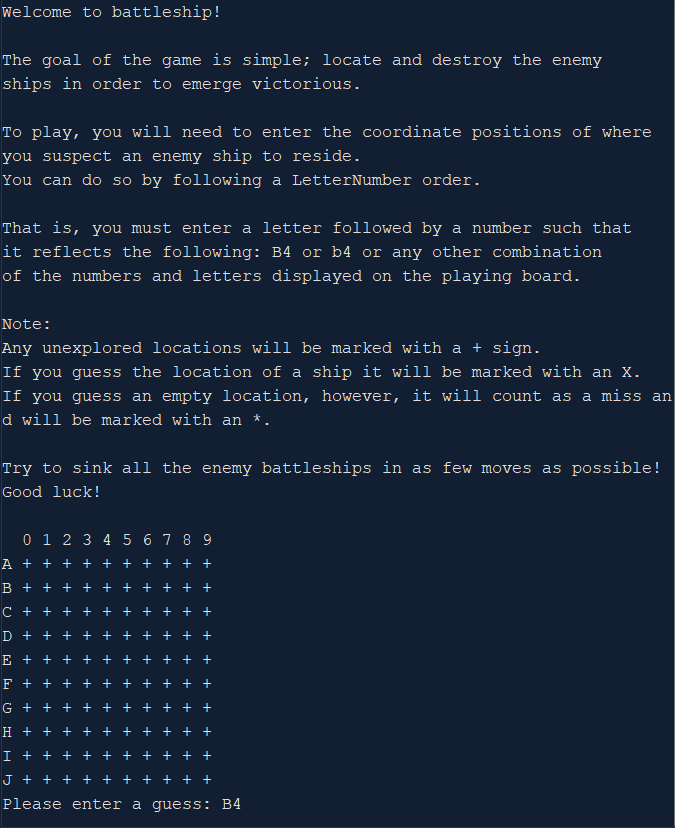
As I made this program I wanted to completely upgrade and convert my first project into something much more functional and advanced. I was able to do so by implementing loads more features into my project as well as functions and multidimensional arrays which functioned a lot more efficiently and effectively.

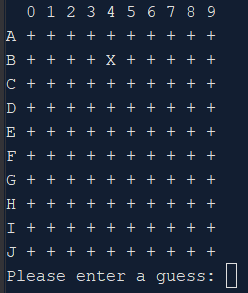
**Project Size:** about 1400 lines

**Description:**

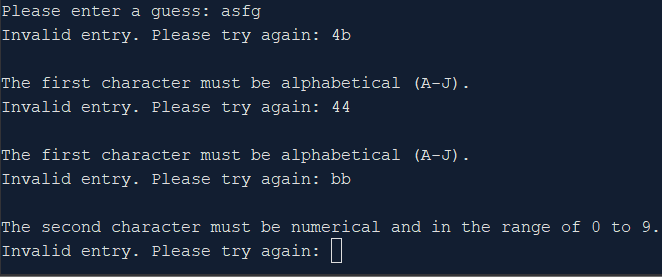
**Sample Input/Output:**

Input: B4 (enter)



Output updates accordingly:  
 

Invalid input yields error prompts to the user:



**Variables:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Type** | **Variable Name** | **Description** | **Location** |
| Board | arr | Holds game board values and properties | main(), init(), shipGen(), run(), print(), update() |
|  | arrray | Initializes game board to a blank state | bordGen() |
| Vars | ships | Array of structures to hold ship properties | shipGen() |
| Integer | size | Holds the game board size | bordGen(), print() |
|  | sizeTemp | Stores size of overlap, if there happens to be any during generation | shipGen() |
|  | xVal | Stores random x value for generation | shipGen() |
|  | yVal | Stores random y value for generation | shipGen() |
|  | count | Used to undo ship placement in case of overlap | shipGen() |
|  | moves | Counts the number of moves taken by the user | run() |
| Char | guess | Pointer to hold the user's guess for the enemy ship position | run() |
|  | temp | C-string of processed user input | Input() |
| bool | gen | Ensures proper ship generation | shipGen() |
|  | valid | Ensure proper input validation | Input(), endGame(), |
| string | letters | Holds string of letters for output on the Y axis | print() |
|  | str | Holds raw user input prior to validation and processing | Input() |
|  | name | Holds user’s initials for highscore file | endGame() |
| fstream | file | Variable used for file I/O | endGame() |

**Concepts:**

The majority of new concepts applied in this project involved structures. Getting used to the syntax and incorporating two-dimensional arrays, arrays of structures, and similar ideas took a bit of getting used to. Upon completing this project, however, I felt a lot more comfortable with structures; as well as classes, despite not incorporating them into this project. The majority of the newfound structure syntax can be found in the shipGen function. Additionally, some newfound file I/O syntax can be found within the endGame function.

**References:**

1. Cplusplus the online resource
2. IBM Knowledge Center
3. Gaddis 9th Edition
4. CIS-5 Project 2 (For algorithm/order ideas – no code was borrowed)

**Program:**