**Battleship game**

Sharvita Paithankar

108172438

23 October 2017

**Program Description:**

This program allows the user to play Battleship against the computer. Before play begins, each player secretly arranges their ships on their primary grid. Each ship occupies a certain number of consecutive squares on the grid. The game is played in rounds. In each round, each player takes a turn to fire a torpedo at a target square in the opponent's grid. The opponent then indicates whether the shot was a hit or a miss. When all of the coordinates of a ship have been hit, the ship is sunk, and the ship's owner announces that they have sunk their ship. When all of one player’s ships are sunk, the other player wins the game.

**Overall Software Architecture:**

The game is played on two grids, one for each player. The grids are typically square and in our case, is 10 by 10.The following classes were used to design this battleship program :Grid, Player User, Computer, and Water Vehicle. The user is asked to input where to put their ships in the grid. Once the user inputs a name of the file, the information sent to grid class and water vehicle objects are made to place on grid. If a ship input is invalid, it is ignored and not placed on grid. For the computer, the ships are generated randomly in the Computer.cpp class and placed on its grid. The user is now asked to shoot torpedoes. If it is valid, the following coordinate is marked on user’s grid. The user will be notified if the ship was either miss or a hit. Player class has virtual function like attackShip(), getShipInfo(), and setShip() because each player attacks and shoots differently and virtual function allowed it to make this program easier to read, WaterVehicle() class has isSunk and isHit function which checks if ships are hit

and sunk. These functions are called everytime a user shoots a torpedo. The process is repeated until either the user decides to quit or we have a winner.

**Input Requirements:**

The user will be asked to input a filename for the program to read in the location of the ship(starting position) and its orientation.

The user will also be asked where they want to shoot. If they do not input valid inputs, they will lose their chance to shoot and the computer shoots.

**Output Requirements:**

Miss : O

Hit : X

Ship : S(submarine), B(battleship), C(Cruiser), K (carrier), d(Destroyer)

Water : “ “ (Blank)

**Problem Solution Discussion :**

I was having trouble with allowing the user’s and computer’s class to remember where they shot and missed. After the user shot torpedos, the program used to lose data and did not print out the present hits and misses for both computer’s and player’s grid. After examining the program and debugging, I realised that when a Player return a Grid(there is a getGrid() function in Player.cpp, which return a Grid), a copy of a grid is returned and not the grid has all the information. This issue was fixed by returning the Grid by reference.

**Data Structures:**

A vector was used in the grid class to created a board for the game. It is an integer vector and each number represents something in the game(0 = water, 1 = hit, 2 = miss, 3 = Submarine, 4 = Cruiser, 5 = Destroyer, 6 = Carrier, 7 = Battleship). A certain number is placed when a ship, or if the players hit or miss on the grid. Inheritance will be used in this coding assignment. I think this is useful because virtual functions can be made for the ships since they have a “is a” relationship with Water Vehicle Class. The grid class has “has a” relationship which can be used to placed ships on the grid.

**User interface Scheme :**

The user is shown how the grid looks at the very beginning of the game. They are then asked to “Enter the name of the file” to input all the ship information. They are then shown their grid and the ships that have been placed.

They are informed “It’s your turn to shoot the torpedo!”

And asked to “Enter a letter between A-J for column. “ and “Enter a number between 1-10 for row”

It will either be a “Hit” or a “Miss”.

“Computer has shot its torpedo” will be printed after computer generates random numbers to shoot user’s grid. It then prints the user’s grid to show if computer has “hit the ship” or “Computer missed. It asks the user “Do want to continue playing? Enter 1 for yes and 2 for no” and represents the process of asking the column and row.

**Status of program:**

What works :

1. Asks user to input file
2. Checks if ships are valid and are able to be placed on grid
3. Asks the user for column and row and as long as they are correct, shoots the computer’s grid.
4. Generated random number for computer to shoot torpedo, doesn’t shoot in the same place.
5. Asks user if they want to quit game.
6. Prints out the user’s grid and hits and misses on computer’s grid

What does not work :

1. Does not check correctly if the column and row are entered properly, goes into an infinite loop if the user enters wrong column
2. Does not inform the user correctly which battleship has been hit. Just says “one of your ships has been hit”
3. Have not tested if the hasWon() function works and if it correctly informs which player has won

**Status of program on CSE GRID:**

Does not compile on cse grid and give the following error :

main.cpp:(.text+0x9d): undefined reference to `Grid::Grid()'

main.cpp:(.text+0x9d): undefined reference to `Grid::printGrid()'

main.cpp:(.text+0xc4): undefined reference to `User::User()'

main.cpp:(.text+0xd3): undefined reference to `Computer::Computer()'

main.cpp:(.text+0xfe): undefined reference to `User::getShipsInfo()'

main.cpp:(.text+0x138): undefined reference to `Player::getGrid()'

main.cpp:(.text+0x140): undefined reference to `Grid::printGrid()'

main.cpp:(.text+0x15e): undefined reference to `Computer::getShipsInfo()'

main.cpp:(.text+0x17c): undefined reference to `Player::getGrid()'

main.cpp:(.text+0x19d): undefined reference to `Player::getGrid()'

main.cpp:(.text+0x1da): undefined reference to `Player::getGrid()'

main.cpp:(.text+0x1ef): undefined reference to `User::attackShips(Grid&)'

main.cpp:(.text+0x1fe): undefined reference to `Player::getGrid()'

main.cpp:(.text+0x210): undefined reference to `Player::getGrid()'

main.cpp:(.text+0x21b): undefined reference to `Grid::printUserHitsOnGrid(Grid&)'

main.cpp:(.text+0x246): undefined reference to `Player::getGrid()'

main.cpp:(.text+0x25b): undefined reference to `Computer::attackShips(Grid&)'

main.cpp:(.text+0x26a): undefined reference to `Player::hasWon()'

main.cpp:(.text+0x2a2): undefined reference to `Player::hasWon()'

The following illustrates all the classes and functions used in the program.

