ITWM5113 – SOFTWARE DESIGN AND DEVELOPMENT

ASSIGNMENT 1

Lecturer Name: DR. SIMON LAU BOUNG YEW

Prepared By:

STUDENT NAME	MATRIC NO.
Patricia A/P Sitinathan	MC210914598

Project Description

Module 1 Project - Battle Ships

This project will help you get more familiar with arrays. You will be recreating the game of battleships. A player will place 5 of their ships on a 10 by 10 grid. The computer player will deploy five ships on the same grid. Once the game starts the player and computer take turns, trying to sink each other's ships by guessing the coordinates to "attack". The game ends when either the player or computer has no ships left.

Step 1 – Create the ocean map

The ocean map is represented by a 10 by 10 grid of different characters. The grid is managed by a two-dimensional array. You will use this 2D array to save where the user and computer decide to place their ships, as well as when someone tries to attack a location and misses. At the start of the game the array will be empty and as the game is played you will change what is stored at each index of the array accordingly.

Output:

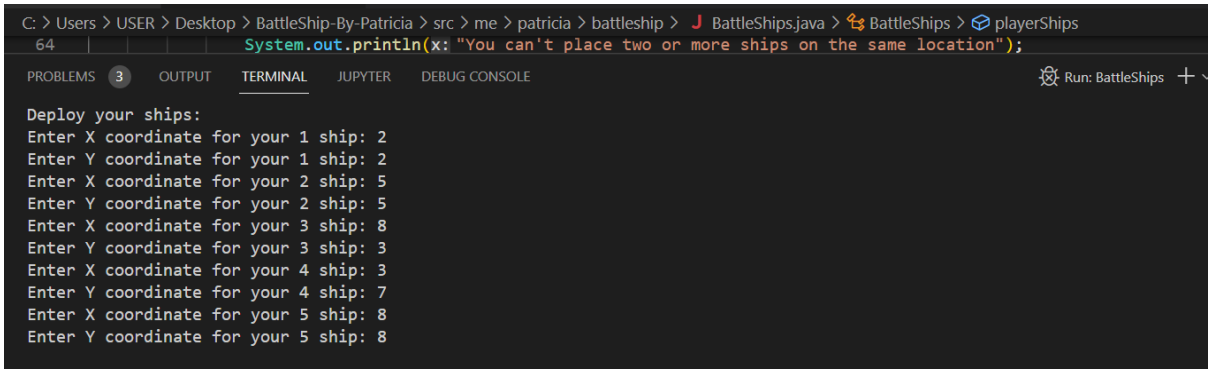
```
PS C:\Users\USER>  & 'C:\Program Files\Amazon Corretto\jdk11.0.15_9\bin\java.exe' '-cp' 'C:\Users\USER\AppData\Local\Temp\vscodesws_7
218e\jdt_ws\jdt.ls-java-project\bin' 'me.patricia.battleship.BattleShips'
**** Welcome to Battle Ships game ****
Right now, sea is empty

  0123456789
0|          |0
1|          |1
2|          |2
3|          |3
4|          |4
5|          |5
6|          |6
7|          |7
8|          |8
9|          |9
  0123456789
```

Step 2 – Deploy player's ships

Once you have your ocean map, you'll need to ask the user where they would like to place their ships. The player should deploy 5 ships. A ship will be stored in a single index of the array as a special character. To place the user's ships they need to tell you the coordinates of where the ship should be placed and you need to update the ocean map to reflect their choices. Remember you'll need to use a Scanner to allow the user to enter in input.

```
import java.util.Scanner; // you must import Scanner to use it

public class BattleShipsGame {

public static void main(String[] args) {

Scanner input = new Scanner(System.in); //This line creates a Scanner for you to use

// ...

/*

Here is some sample code where you ask the user to enter in the coordinates for where to place a ship

*/

System.out.print("Enter X coordinate for your ship: ");

int x = input.nextInt();

System.out.print("Enter Y coordinate for your ship: ");

int y = input.nextInt();

}

}
```

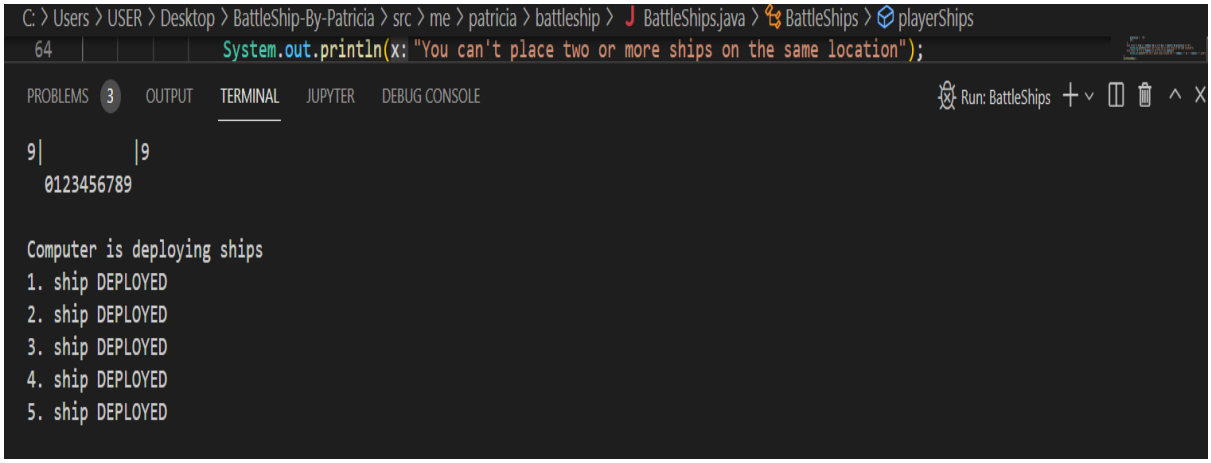
Output :



```
   0123456789
 0|          |0
 1|          |1
 2|  @       |2
 3|      @   |3
 4|          |4
 5|    @     |5
 6|          |6
 7|          |7
 8|  @   @   |8
 9|          |9
   0123456789
```

Step 3 – Deploy computer's ships

Output :



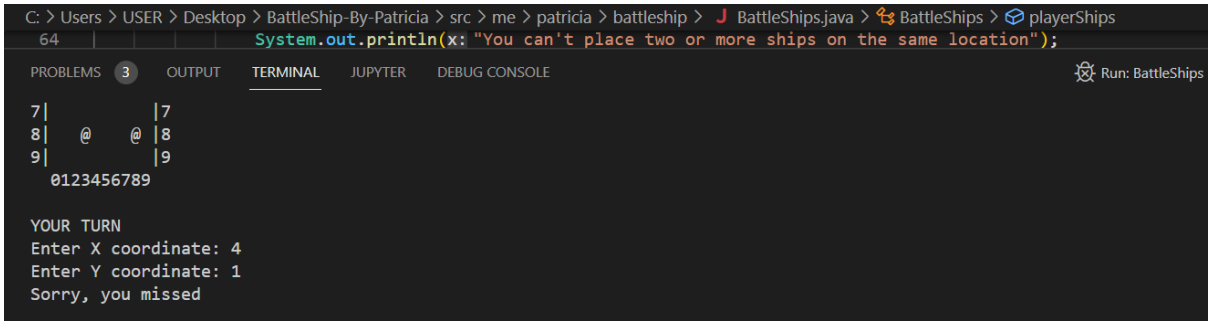
Step 4 – Battle

Player's Turn

Once the player and computer have placed their ships it's time to start the battle! During the battle, the player and computer will take turns guessing X and Y coordinates of the opponent's ships. Every coordinate guessed should be marked so they players know not to guess there again.

When the player enters X and Y coordinates you should check if those coordinates are valid within the Ocean Map and haven't been guessed by the user yet, keep re-prompting until the user enters a valid guess. Once the guess is valid your program needs to evaluate the result of the move.

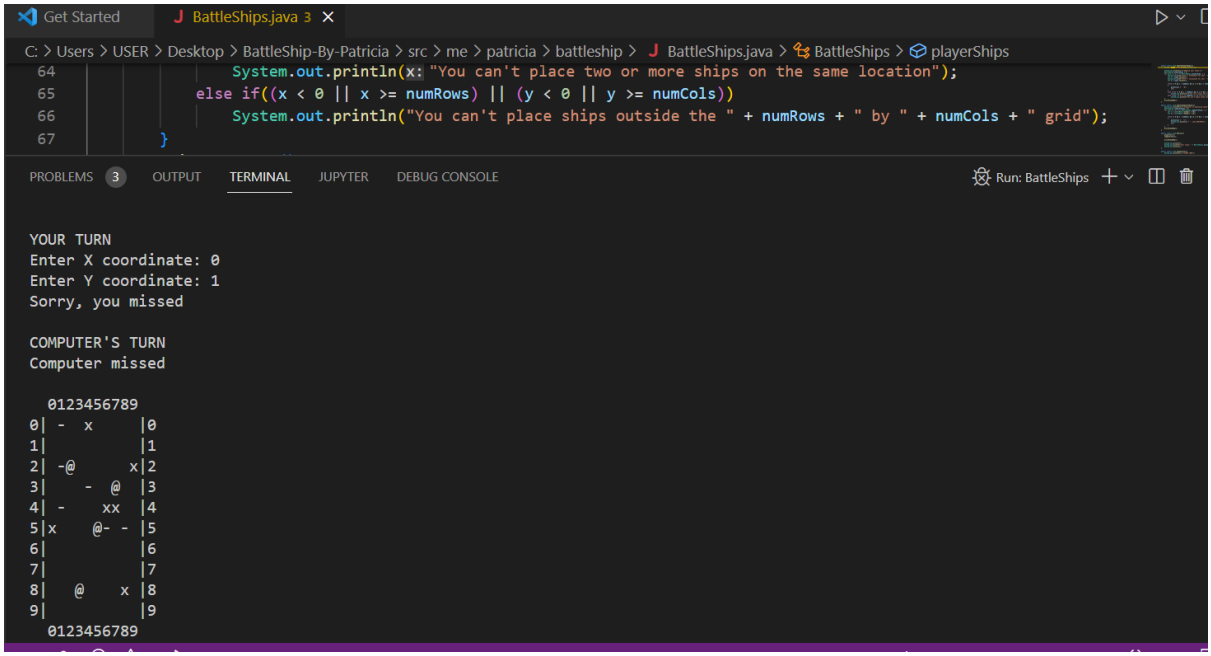
Output :



Computer's Turn

After the player guesses a coordinate it's the computer's turn to guess. The computer's attack should be two randomly generated coordinates. You will need to keep generating random numbers until you get a valid guess, meaning a location that is within the bounds of the board and the computer hasn't already guessed. Once the computer makes a valid guess, you want to print a little update to the user:

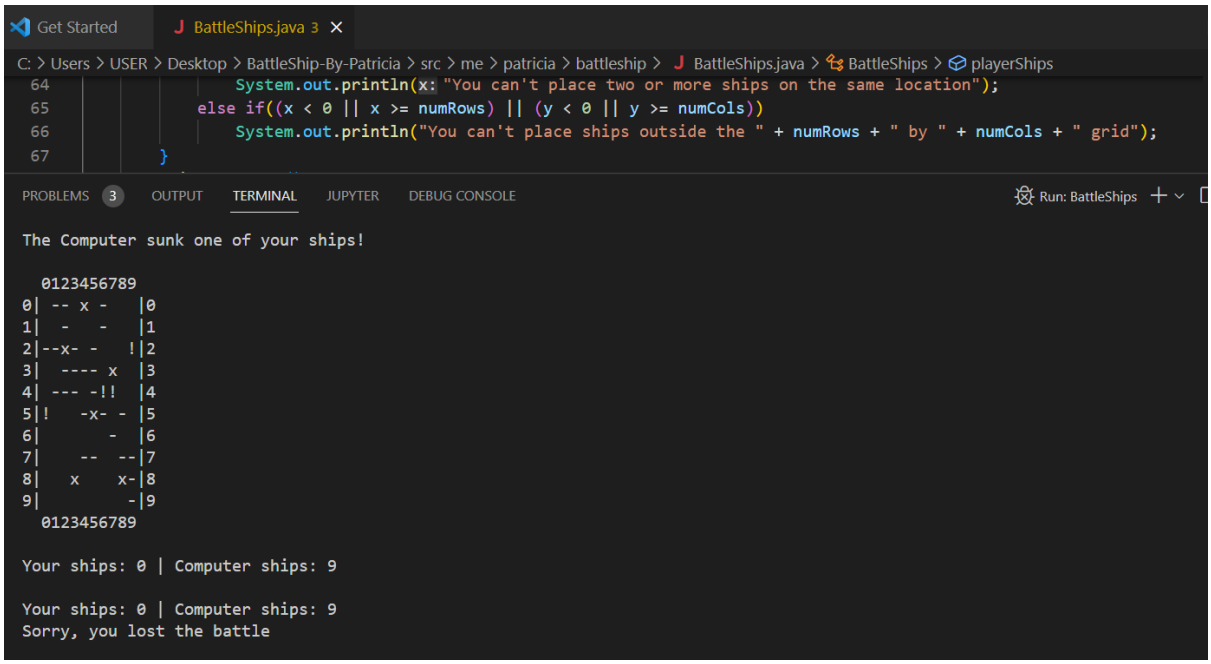
Output :


Get Started
BattleShips.java 3
C: > Users > USER > Desktop > BattleShip-By-Patricia > src > me > patricia > battleship > BattleShips.java > BattleShips > playerShips
64 System.out.println(x: "You can't place two or more ships on the same location");
65 else if((x < 0 || x >= numRows) || (y < 0 || y >= numCols))
66 System.out.println("You can't place ships outside the " + numRows + " by " + numCols + " grid");
67 }
PROBLEMS 3
OUTPUT
TERMINAL
JUPYTER
DEBUG CONSOLE
Run: BattleShips
YOUR TURN
Enter X coordinate: 0
Enter Y coordinate: 1
Sorry, you missed
COMPUTER'S TURN
Computer missed
0123456789
0| - x |0
1| |1
2| -@ x|2
3| - @ |3
4| - xx |4
5|x @- - |5
6| |6
7| |7
8| @ x |8
9| |9
0123456789


Get Started
BattleShips.java 3
C: > Users > USER > Desktop > BattleShip-By-Patricia > src > me > patricia > battleship > BattleShips.java > BattleShips > playerShips
64 System.out.println(x: "You can't place two or more ships on the same location");
65 else if((x < 0 || x >= numRows) || (y < 0 || y >= numCols))
66 System.out.println("You can't place ships outside the " + numRows + " by " + numCols + " grid");
67 }
PROBLEMS 3
OUTPUT
TERMINAL
JUPYTER
DEBUG CONSOLE
Run: BattleShips
8| @ x |8
9| |9
0123456789
Your ships: 4 | Computer ships: 6
YOUR TURN
Enter X coordinate: 5
Enter Y coordinate: 0
Boom! You sunk the ship!
COMPUTER'S TURN
The Computer sunk one of its own ships

Step 5 – Game over

When the user and computer are done guessing, display the current state of the ocean map and score.



One Drive link : BattleShip-By-Patricia.rar

Github : https://github.com/Patricia2828/Assignment-