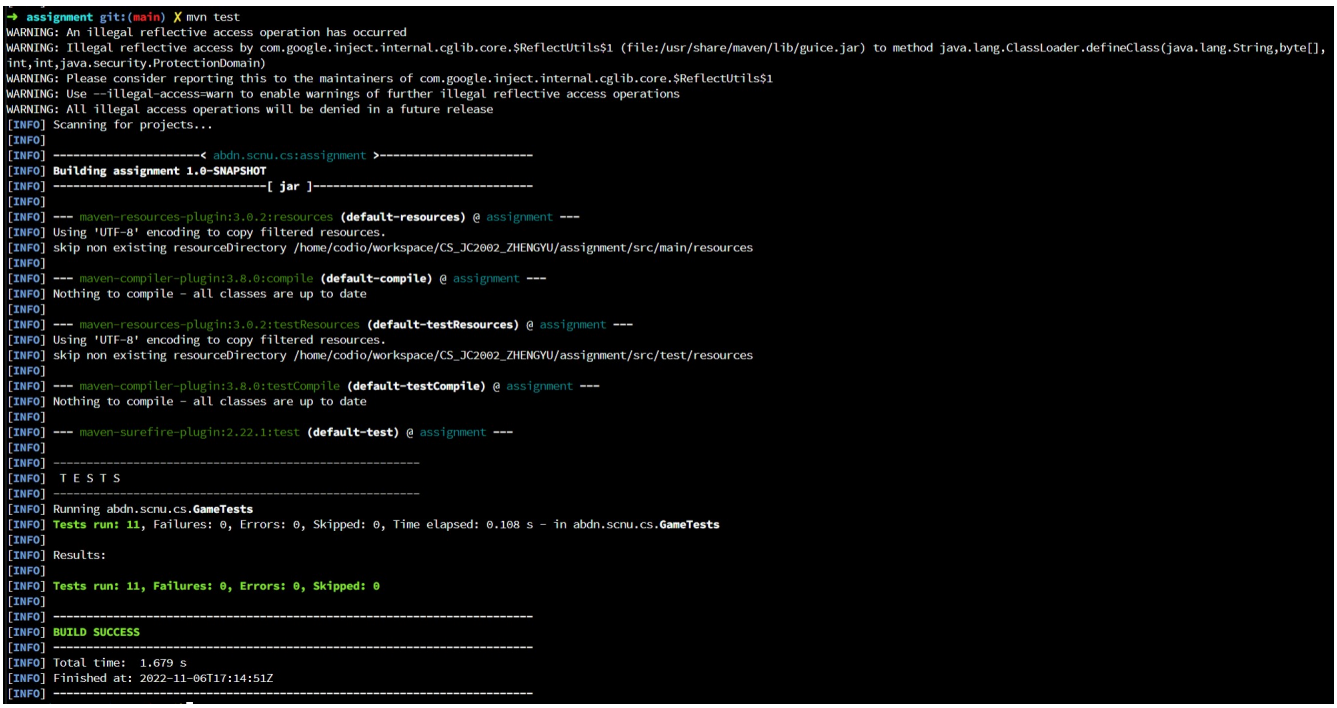
Report

Github link

JavaAssignment

Task

test result:



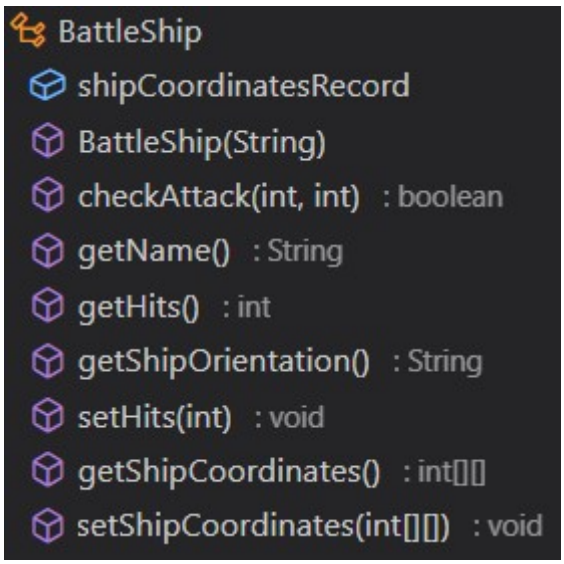
package result:

```
➜ assignment git:(main) mvn package
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by com.google.inject.internal.cglib.core.$ReflectUtils$1 (file:/usr/share/maven/lib/guice.jar) to method java.lang.ClassLoader.defineClass(java.lang.String,byte[],
int,int,java.security.ProtectionDomain)
WARNING: Please consider reporting this to the maintainers of com.google.inject.internal.cglib.core.$ReflectUtils$1
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
[INFO] Scanning for projects...
[INFO]
[INFO] ----------------------< abdn.scnu.cs:assignment >-----------------------
[INFO] Building assignment 1.0-SNAPSHOT
[INFO] --------------------------------[ jar ]---------------------------------
[INFO]
[INFO] --- maven-resources-plugin:3.0.2:resources (default-resources) @ assignment ---
[INFO] Using 'UTF-8' encoding to copy filtered resources.
[INFO] skip non existing resourceDirectory /home/codio/workspace/CS_JC2002_ZHENGYU/assignment/src/main/resources
[INFO]
[INFO] --- maven-compiler-plugin:3.8.0:compile (default-compile) @ assignment ---
[INFO] Changes detected - recompiling the module!
[INFO] Compiling 10 source files to /home/codio/workspace/CS_JC2002_ZHENGYU/assignment/target/classes
[INFO]
[INFO] --- maven-resources-plugin:3.0.2:testResources (default-testResources) @ assignment ---
[INFO] Using 'UTF-8' encoding to copy filtered resources.
[INFO] skip non existing resourceDirectory /home/codio/workspace/CS_JC2002_ZHENGYU/assignment/src/test/resources
[INFO]
[INFO] --- maven-compiler-plugin:3.8.0:testCompile (default-testCompile) @ assignment ---
[INFO] Changes detected - recompiling the module!
[INFO] Compiling 1 source file to /home/codio/workspace/CS_JC2002_ZHENGYU/assignment/target/test-classes
[INFO]
[INFO] --- maven-surefire-plugin:2.22.1:test (default-test) @ assignment ---
[INFO]
[INFO] -------------------------------------------------------
[INFO]  T E S T S
[INFO] -------------------------------------------------------
[INFO] Running abdn.scnu.cs.GameTests
[INFO] Tests run: 11, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.111 s - in abdn.scnu.cs.GameTests
[INFO]
[INFO] Results:
[INFO]
[INFO] Tests run: 11, Failures: 0, Errors: 0, Skipped: 0
[INFO]
[INFO]
[INFO] --- maven-jar-plugin:3.0.2:jar (default-jar) @ assignment ---
[INFO] Building jar: /home/codio/workspace/CS_JC2002_ZHENGYU/assignment/target/assignment-1.0-SNAPSHOT.jar
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  3.943 s
[INFO] Finished at: 2022-11-06T17:11:34Z
[INFO] ------------------------------------------------------------------------
```

Task 1

Staus: **completed**

Outlining: create `BattleShip.java` and extends `AbstractBattleShip` class. In constructor, use Random object to defind the `shipOrientation` and init other params. Write the member functions which defind in `AbstractBattleShip`. Finish the `checkAttack` method. Check all possible situation. Defind a array `shipCoordinatesRecord` to record the coordinate which has been hit.



Task 2

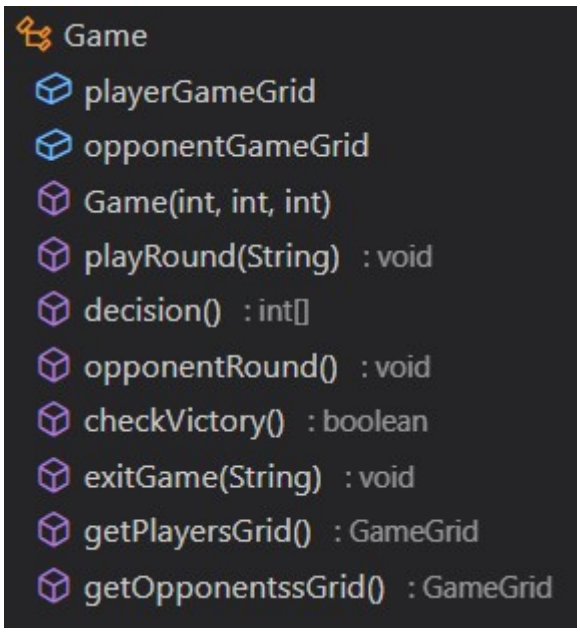
Staus: **completed**

Outlining: Create `GameGrid.java` anad extends `AbstarctGameGrid`. Create the `initializeGrid()`. Finish the `generateShips()`. Finish the `placeShip()`. Create `PlayerGameGrid.java` and `OpponentGameGrid.java`, extends `GameGrid`. Defind `printGrid()` in them.

Task 3

Staus: **completed**

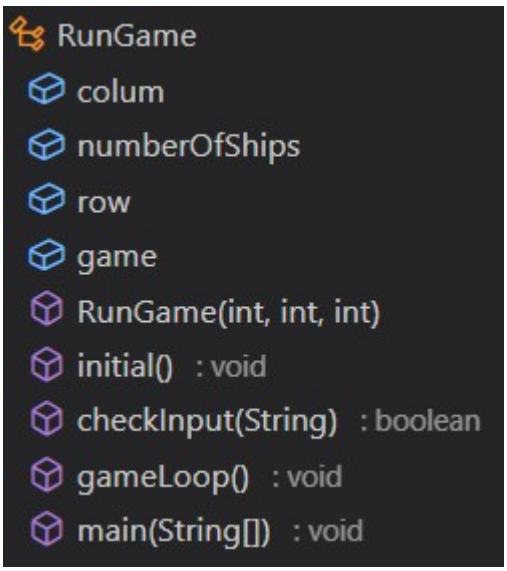
Outlining: Create `Game.java` and implement `GameControls`.create player's and opponent's grid and implement getter methods of them. Implement `exitGame()` method. Implement `checkVictory()`. Implement `playRound()` to make robot smart. In this case, oppenent decision depends on the probability which calculate from "X" coordinates. If all coordinates are less than or equal to 0, its decision depends on the probability which calculate from the number of unknown coordinates. Unfortunately, the fact is I will lose to the robot in most cases. Beyond that, it can be more smart when use RL(Reinforcement Learning) to make decision. If use RL, I prefer to use DQN(Deep Q-Network) that use neural network Q(s,a;w) to approximate Q* (s,a) and make the value-based decision. The environment is easily to create but RL is a little hard to applicate by java, which is esily to implement by C++ or python. Thus, I prefer to use probability algorithm instead of that.



Task 4

Staus: **completed**

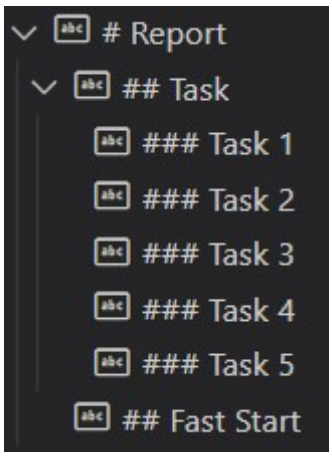
Outlining: Create `RunGame.java` and start it by create a `Game` object. Use params to get the height, hight and the number of ships. Use while loop function to make sure rounds are loop. Use regex `^[0-9]+,[0-9]+$` to match corret input. I create an `IOException` class to catch the exception and throw it. Use `exitGame()` function which in game object to check the `exit`.



Task 5

Staus: **completed**

Outlining: Conclude the tasks status and desribe each with a short words.



Fast Start

Switch to the JAVA project root directory `assignment`

- package the project

```
mvn package
```

- run the jar with you params which are **height**, **width** and **number of ships**

```
java -jar .\target\assignment-1.0-SNAPSHOT.jar 5 5 3
```

Out of Task

In my opinion, the human-computer interaction interface is helpful because it is too hard to calculate the coordinates by eyes and hands when the width and height are both large. For this reason, I will improve it by adding GUI and dynamic refresh pattern. In addition, I will apply RL on opponent decision if I have enough time.