**Project 1: Build a system – Team's choice of language and platform**

**Overview**

- Let's make Battleship!
- Battleship is a two-player game.
- Both players secretly place 1 to 5 ships on a 10x10 grid.
- Taking turns, each player announces where on the opponent's grid they wish to fire.
- The opponent must announce whether or not one of the ships was hit.
- The first player to sink all the opponent's ships wins.

**Requirements**

1. Game Setup
   a. Board size
      i. 10x10
      ii. The columns are denoted by letters (A-J)
      iii. The rows are denoted by number (1-10)
   b. Number of ships (per player)
      i. Given by user
      ii. Minimum of 1 and a maximum of 5
   c. Types of ships
      i. This will be based on the number of ships chosen
      ii. If a total of 1 ship is chosen, then each player gets a single 1x1 ship
      iii. If a total of 2 ships is chosen, then each player gets a 1x1 and 1x2 ship
      iv. This continues up to 5, where each player with a 1x1, 1x2, 1x3, 1x4 and a 1x5 ship
   d. Ship placement
      i. After the number of ships is chosen, players need to be able to secretly orient and place the ships on their board
2. Playing the Game
   a. Taking turns, the users pick a space on the opponent's board to "fire" at
   b. They must then be informed if the shot was a "hit" or a "miss"
   c. The player's view should be updated to reflect this
   d. After each shot, it is the other players turn
3. Destroying a ship
   a. Once a ship has been hit in every space it occupies, it is sunk

b. For example, if the 1x3 ship occupies spaces B3, B4, B5

c. Once the opponent has shot those three spaces, that ship is sunk

4. Player's view

a. A player should have full view of their board and where their ships are placed

b. Show how many times each ship has been hit

c. A player should have a board to track all shots they've fired and whether they were misses or hits

5. Game End

a. Once a player has sunk all the opponent's ships, they immediately win

## Language and Platform

- Team's choice.
- Avoiding team conflict!

## Requirements

- Create your GitHub account if you do not have one.
- Learn how to use git.
- Collaborate with your teammate to contribute to the project on GitHub repository.
- **This project is 25% of total points.** You must report your weekly progress in each week during the lab time to me and the weekly demo is 10% of the total points. The final report (including GitHub repository) is 15% of the total points.

## Weekly Demo (10%)

- You will demo on a device of your own choice during the lab time.
- You can present with a PowerPoint, using GitHub repository, or directly demo the game to show your contribution.
- I might test your application. Crashes, memory leaks or other things that you also hate in bad software will be met with a penalty.
- Your product should be intuitive to use and your code should be easily extensible and divided into logical components.
- You should be prepared to answer questions from me.
- The grades will be given according to your individual contribution and answers to the questions.

## Project Report (15%)

- Write a report to summarize your project.
- Include your GitHub repository link in your report.
- For GitHub repository, you should include necessary documentation (e.g. README). The codes should include necessary comments (e.g. name of program contained in the file, brief description of the program, creation date, etc). You should provide comments explaining each line of code or a group line of codes.
- The grades will be given according to the clarity of the report and the quality of the GitHub repository.

**Deadline**

The deadline for project 1 is February 25 at 11:59 pm. Late submission will be punished. 1 day late 10% reduction, 2 days late 20% reduction, ..., until there is no point left.