**D E S I G N D E C I S I O N S**

Most of the design I decided to go with was decided solely because I thought it would be easiest. Some of it turned out to not be and I tweaked it slightly, others worked well. Some of the design ideas I took from you during lecture.

**I M P L E M E N T A T I O N P R O C E S S**

Implementation went pretty straight forward from the design due to the use of methods & your ideas you gave us. Only thing I didn’t take from you was having a placeShip in the ship class – it seemed useless.

**P R O B L E M S**

Changes I made were adding turns while placing ships. I originally hard coded it as 5 ships from player 1 and then 5 ships from player 2 because it’s pre-game setup. I didn’t see that as part of the game, so no turns were coded. I added turns and a bit of freedom with type of ship to be placed because of the design restraints you placed upon us. I also had to go back and add an output for when a ship is destroyed.

I was unable to implement *correctly* the endgame screen when a player wins. I could not figure out my error for the life of me – I spent at least 8 hours on the completed code trying to solve *why* the players could not win immediately. The results were not consistent to top it off – sometimes it’d work perfect, sometimes it’d take an additional turn from the other player to win and sometimes the other player would win on that additional turn.

Sadly, another error I could not solve was players being able to attack the same location more than once – the if-statement never executed. I attempted many variations of code in that area and all nearby (methods it calls) areas and was not successful. I decided it wasn’t as important as players not being able to win, so I left it.

**R E S U L T S**

I strengthened my understanding of classes and pointers with this project. I also learned that I am not very good at debugging code when the part that looks like the trouble-source isn’t actually the problem; especially since it was coded incrementally, and the error stemmed when adding said error correction. For the above, I went through my entire code, line by line, trying to see what could cause players not winning (since it’s coded w/ return 0; at the end of the if (endgame) statement).

I’ve attached a typescript for the test results. However, I only did one playthrough (with one ship and modified Gameover methods). I coded and tested for capitalization of ships during placement as well as all error correction for placement of ships (out of bounds, overlapping). I coded and tested for hits, misses and overlapping attacks (was not successful) and coded and tested for end game scenarios and was able to end the game, but not immediately and sometimes not for the correct winning player. In the typescript, it displays hits and misses and only one variant of ship placement – however that was tested thoroughly and ship placement should never result in an error.