INTRODUCTION TO COMPUTER SCIENCE
FINAL PROJECT PROPOSAL
HUNGRY SHARK GAME

Group Members:
Varvara Snitko (vgs6617)
Himanshi Lalwani (hl3937)

Description:
This game is a variation of Hungry Shark Evolution. A hungry shark is in search of fish. There are lots of fish in the sea which the shark can eat but there are also jellyfish which can poison the shark and hence, are to be avoided by the shark. If time allows, we will also add thornyhead fish which are to be avoided too.

We will be creating two difficulty modes: easy and difficult. In easy mode, the shark is supposed to eat 10 fish to win. The shark also has three lives. Therefore, if it bumps into jellyfish thrice, the game ends there. In difficult mode, the shark is supposed to eat 20 fish. The number of fish and jellyfish appearing will be relatively less and more respectively. The number of lives remain the same.

The main screen will have a start button. The user gets to choose a level on the next screen using mouse interaction. Shark is managed by the arrow keys. The game is based on collision detection. Every time the shark eats a fish or collides with the jellyfish, a sound effect is executed. We are considering having a background music too which will be played on loop. The number of fish eaten is being updated in the upper left corner and the number of lives remaining is being updated in the upper right corner. At the end of the game, 'You win/lose' is displayed and the user is given instructions to restart the game.

Demo:
You can watch demo of the game application here: https://youtu.be/GOUMxLOGP3I

Features/Distribution of Work:

- ➔ Creating the main screen with the start button - Himanshi
- ➔ Instructions of the game and modes on the next screen - Himanshi
- ➔ Movement of Shark - Himanshi
- ➔ Adding background - Himanshi

➔ Displaying the number of fish eaten - Himanshi
➔ Creating fish and jellyfish classes - Himanshi & Varvara
➔ Collision Detection - Varvara
➔ Displaying the number of lives remaining - Varvara
➔ Checking for game over and displaying the result - Varvara
➔ Creating the restart option on the result screen (sending the program to the second screen) - Varvara
➔ Adding sound effects - Varvara