Contributors

Khail Lyons(Part 1, Part 2, Overall setup for repository and framework library, Complete back end programmer)

Sebastian Meredith(Part 3, Part 4, Part 5, Creating and Designing the game, Documenting and commented the code, Completing any of the recording/documentation requirements.)

Name of Framework Library and Link to GitHub

Final Project

https://github.com/Khalil-Lyons/FinalProject.git

Overview of Framework

Our framework is primarily focused on the Input aspect of our game as it provides a lot of functionally for the game, based on the way we designed. Our input strives to be as easy for the user to interact with as possible as well as along with the commenting makes it very easy for programmers to follow along with how we end up processing the information that gets written to the output. As well as we tried to condense our code as best as possible in order to make sure that it works almost instantaneously after the User has put in their input.

Our framework consists of two files that make the game run,

1. The header TicTacToe.h
2. The .cpp file that contains the 3 areas(Input,Logic,Display) TicTacToe.cpp

The header is very straight forward with a few predefinitions for the game including Void Board which is used to create the board, The char Square which creates the array of numbers that are used in the logic and input of the game. Finally, the game state integer which basically just triggers the end of the game based on if a player won or the game was a draw

The Main is broken down into 3 key components(Input, Logic, and Display)

The first part Input which was the main focus tries to be as clear and concise to the user as possible, Although it doesn't seem like a whole lot on the front end it works well with the other aspects of the program and uses them when necessary in order to ensure the user always knows what they are doing and how that is affecting the game.

The Second part Logic is basically just checking to see if 1. The User is playing by the game rules provided and 2. Checking for rows of 3 in every collum during each player move so that when players do enter a draw or win state the game imminently ends and coverts to the appropriate screen.

The Display is mostly just a text-based part of the program that uses the lines in order to make a game board that fits the game of Tic Tac Toe. It is rebuilt on every move in order to always show the players the most current version of the game as well as giving them an opportunity to see the moves that happened in the past.

Future of the framework

I think the focus on Input definitely shows how you can really work to make something very clear to the user and how many things can be unclear. In the past especially with playing other game, I always thought myself to be stupid when I didn't understand a clear functionally of the game, but ultimately it is also the responsibility of the creator to make sure that the functions of their game are able to be understood, and that even if the user is inputting some wrong for instance, like not a number for the tic tac toe game, that the User knows that they made that mistake or didn't understand how the function was supposed to be used. Instead of expecting the User to completely understand what was going on. Therefore through this project, I have seen the importance of having a clear and concise system that is formatted for the User, and not someone who has made or played the game a thousand times over