Jacob Dohoney-Pilling, Brett Hoskins

CS371-001

Professor Simone Silvestri

11/21/23

Pong Project Report

Background

The goal of this project is to create a working server that can handle two clients. The clients connect to the server, and both control two paddles from the game pong. The game updates in real time, moving the paddles, ball, and keeping track of score.

Design

Before we started, we knew we'd have to design and implement code for both client side and server side interaction. The client should send paddle locations, while receiving messages about the ball, score, as well as the opponents paddle locations. We also decided the server should use sockets and threads to effectively communicate with two clients at the same time.

Implementation

We have a playGame() function on the client side which handles both player and ball movements as well as score updates. It initializes a socket connection with the server, allowing it to receive updates from the other client. As far as the server goes, it's implemented in a way that initiates socket connections to listen to incoming client connections, waiting for both to connect before the game can start. The server ensures that information can safely be relayed between clients, in real time, so that you can smoothly run pong.

Challenges

We had great difficulty connecting to one another, which caused an issue where we weren't sure what needed to be fixed because we couldn't even start to test the actual game. We ensured our IP and ports were correct which ensured we could test the game. We ran into a quick error where one paddle moved much quicker than the other, but noticed it happened to the client on the machine hosting the server. This was a simple optimization fix.

Lessons Learned

What did you not know before starting this project that you now do? Before starting this project, neither of us had any idea on how to actually host a server that clients could connect to. It was immensely satisfying not only to do this, but to produce a game that worked properly.

Conclusions

We have implemented the server AND client for a pong game playable between two players. It was a great synchronization problem that resulted in an implementation we are both proud of.