

**Name:** Raaka Mukhopadhyay

**Andrew ID:** rmukhopa

### **Project Description**

My term project is Air Hockey. The most basic description of the game is two opponents trying to score a puck into the other's goal. However, my implementation of the game will contain additional features to increase the difficulty of the game and make the game more complex, at the user's discretion.

### **Competitive Analysis**

There are many other versions of Air Hockey; for example, I've seen a previous 15-112 project that was Air Hockey, and I've also seen different implementations of Air Hockey on other websites. Overall, Air Hockey is comparable to any two player game where the objective is to hit an object into the opponent's side, making Air Hockey comparable to a wide range of games, including Pong (which we have discussed in 15-112).

My project will be very similar to these aforementioned games at its most basic level. However, similar to the aforementioned previous 15-112 project, my project will implement additional features. Specifically, I hope to make customized levels and allow user discretion to customize the board and other game features.

### **Structural Plan**

The finalized project will consist of two files: the `cmu_112_graphics` module and the term project itself.

The most important functions are `timerFired`, `appStarted`, and `computerPlays`. In `appStarted`, the initial values needed to set up the board, controllers, and puck are stored. In `timerFired`, the functions `movePuck` (which moves the puck if it collides with a controller) and `movePuckLegally` (which checks if the puck hits the side of the boards) are called and used to properly move the puck around the board; additionally, `timerFired` calls the functions to allow the computer to play, check if a goal is scored, and implement extra features are called. The purpose of the function `computerPlays` is to tell the computer controller how and when to move the puck.

### **Algorithmic**

The most complex part of my project will be the functions that will control the computer controller with the extra features and levels. I plan on first making sure that the AI is able to play a "regular" game with the user. This is done by having a lag of the computer's controller following the puck, but the controller is able to follow and hit the puck nonetheless. Then, once that is completely implemented, I will begin to alter the function or make a new function that will make sure that the controller can maneuver around pucks and bouncers, which are additional features of the game. Finally, I will alter the function or make a new function to help implement levels of the game to make the game more challenging and interesting for the user.

### **Timeline Plan**

By TP2, I hope to have the (1) functions that control the computer controller working properly for the extra features and levels. I also hope to (2) redesign the board to more accurately depict an actual air hockey table. I also hope to (3) smooth the random bugs that sometimes occur with the puck. I

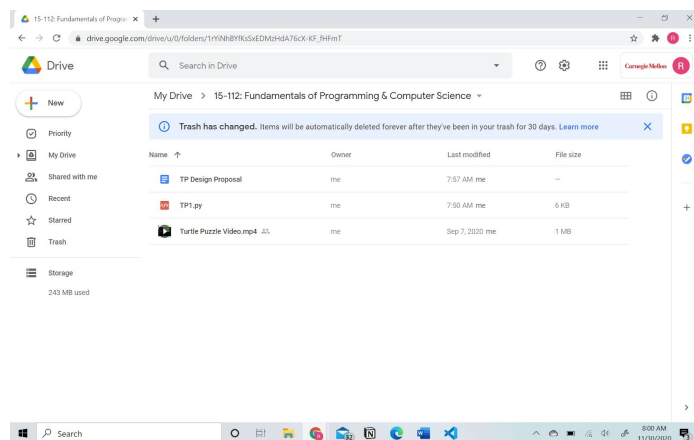
will be working on all these components until their completion, but for more specification, I hope to complete (3) by Wednesday, (2) by Thursday, and (1) by Saturday.

By TP3, I hope to have (1) properly designed the buttons and user interface for choosing the features and levels. I also (2) hope to implement a few extra features that the user can design/customize for a more interesting and interactive experience. I will be working on all these components until their completion, but for more specification, I hope to complete (1) by Sunday and (2) by Monday.

Of course, I also hope to complete all the other TP3 deliverables on time and will be completing them as I am able to do so.

## Version Control Plan

I am using Google Drive as my version control. Below, I have put a screenshot of my current TP1 file uploaded into Google Drive for safekeeping.



## Module List

I am using `cmu_112_graphics` and other Python built-in libraries (math, random, etc.) that we have used in class. I am not using any additional modules.

## TP2 Update

My project will now include a maze obstacle course feature (separate from the normal air hockey game) to add algorithmic complexity. As of now, there will be no AI generated levels for the regular air hockey game.

## TP3 Update

The game is now divided into two modes: regular air hockey and the maze mode.

The maze mode includes a randomly generated maze with three different levels of difficulty (determined by the number of “walls” in the maze). If the user wants to generate a new maze, they can select that option. The maze mode also includes a custom generated maze, which the user can create themselves and play with. The win condition for the maze mode is “scoring” the puck into the goal under 20 seconds, while the goal moves to a random location every 5 seconds.

The regular air hockey mode has three options: regular air hockey, pitt mode, and bouncer mode. In the regular air hockey mode, the user can play air hockey as usual. In pitt mode and bouncer mode, the user can choose where to place either the pitt or the bouncer and play using the obstacle to make the game more interesting.