112 Ping Pong Proposal

Name: Allen Luo
Andrew ID: yuhualuo

Project Description:

The name of my project is 112 Ping Pong, which is a game that simulates real ping pong games. Players will use the mouse to control a ping pong bat and compete with A.I. athletes, trying to win the game.

Competitive Analysis:

Because the rule of my game derive from real ping pong competitions, so the rules will be quite similar to all other ping pong games online. In specific, players need to serve the ball above the net and land on their opponent's half table. Also, the ping pong ball cannot bounce on the same side of the table for more than one time.

There are also many differences between my game and other games and projects online. First, my ping pong game is built based on 3D coordinates, in which both the bat and the ball have their own x, y, and z coordinate to show their movements. While many online projects only construct their code based on 2D coordinate systems. In addition, my game is like a physical simulator of ping pong games, and I used many physical formulas to calculate the dynamic motions. Also, I consider factors like gravity, air resistance, energy loss during collisions to make my game more real. These physical factors are not thoroughly considered in most published online ping pong games, which is another advantage of my game.

Structural Plan:

My finalized project will have two main classes: 3DObject and myBat. The class 3Dobject has two sub-classes called ball and opponentBat. Since those things are frequently used in my code, so I make them into objects and write classes and methods for them. For file management, I will use a folder called "Image" to store all the photos that used in this game.

Algorithm Plan:

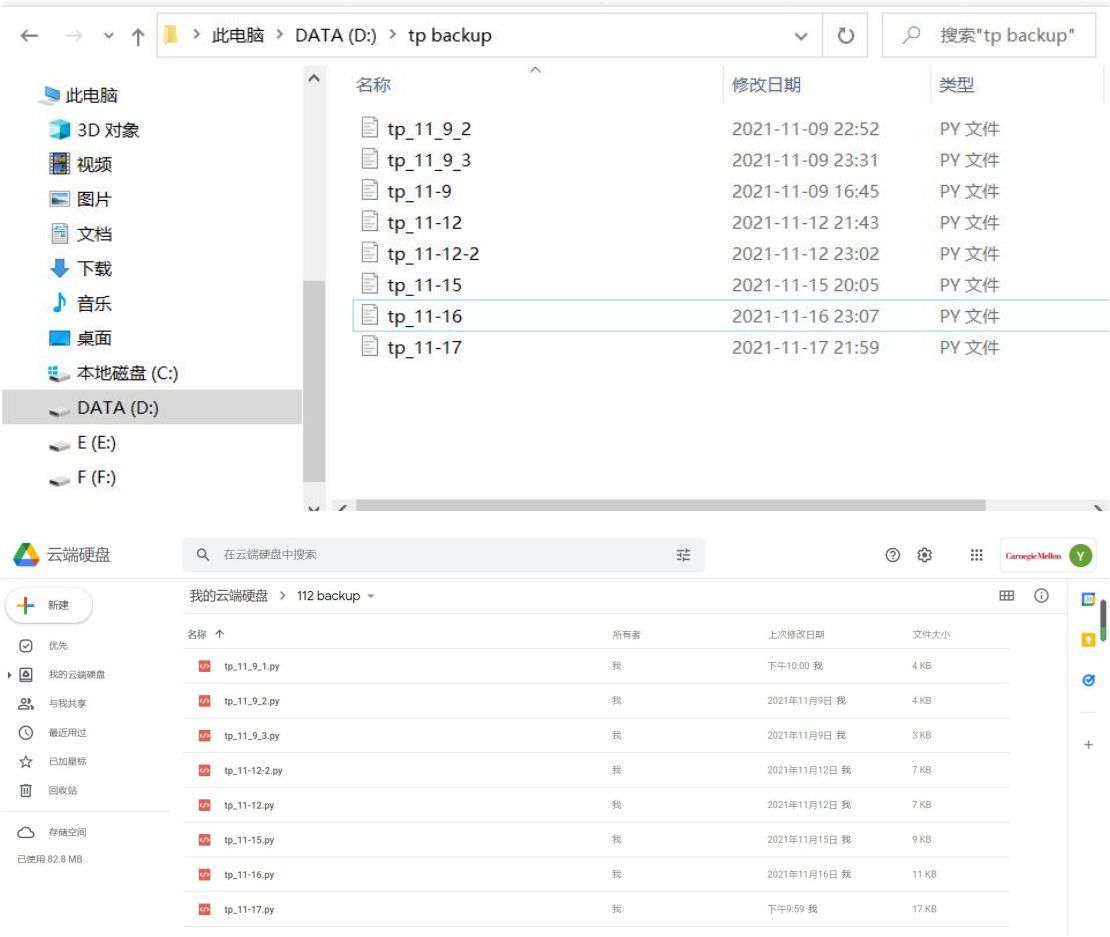
Three parts of my part are really complex. The first part is 3D graphics. To project the ball's 3D coordinate into the 2D canvas, I use the matrix dot product to achieve this conversion. I also use matrix multiplication to rotate the 3D coordinate system, making it fit people's perspectives. Aside from using the matrix, I also simulate the physical motions of the ball using many physical formals. I write many functions to calculate the horizontal movement, vertical movement, and forward movement to make the game more realistic. At last, there is a game A.I. in my ping pong game, and it can calculate the falling point of the ball and control the opponent's bat correspondingly.

Timeline Plan:

By TP0, I will finish all the coordinate systems and the basic control system of the bat and the ball. By TP1, I will apply physics laws and calculations to compute the specific motion of the ball. Also, I will finish the basic rules in the ping pong game. By TP2, I will add the A.I. opponent to my game, and I will also work on more physics calculations of the ball and the bat. By TP3, I will add more interesting features like the rotation of the ball, different modes of game, and other features.

Version Control Plan:

I store backup files on both my computer (another disk) and the online google drive. The following is the screenshots of the backup files.



Module List:

1. 112 graphics
2. Numpy
3. Math

TP2 Update:

Use one more module: random, and others remain the same.

TP3 Update:
No change of design is made.