Project Description:

This project is named Crazy Tanks which pays tribute to the flash game Tank Trouble 2. Player can either play with game AI or play with another player. The goal is to destroy your opponent’s tank within the maze. But unlike usual tanks’ bullets, the bullets in Crazy Tanks will bounce around when they hit the maze war, making it difficult to aim and possible hurting the shooter itself. There will also be powerups and levels to make the game more complex and interesting.

Player1: WSAD move, SPACE shoot

Player2: ↑↓←→ move and L shoot

Press “r” restart

Competitive Analysis:

This project is similar to the game Tank Trouble 2. The two have the same basic control and tank movement, as well as bullets bouncing feature. Each time one kills an opponent tank, one’s score will increase by one.

This project will be different in that player will be able to unlock new tanks with new features as they score higher.

New features like faster speed, more power bullets, cooler skin color, smaller tank size.

Structural Plan:

GameObject(Object):

(from the 112 website, a class that all objects in the game inherent, it can move, rotate and initialize objects)

Tank(GameObject):

(a class that store info for a tank, functions like moving the tank while checking whether it is colliding with a wall, and drawing explosion when tank dies)

Tank2(GameObject):

(a class that store info for another tank, 2nd player or AI player)

Map(GameObject):

(a class that can initialize maps)

Bullets(GameObject):

(a class that store info about bullets, have functions collide to do the bouncing effect, and kill itself automatically after certain period of time)

PygameGame(object)

(from the 112 website, a basic run function for pygame)

Game(PygameGame):

(the main function for the game, modify functions like init(self), redrawAll(self, screen), keyPressed(self, code, mod) and timerFired(self, dt) to play the game )

Algorithmic Plan:

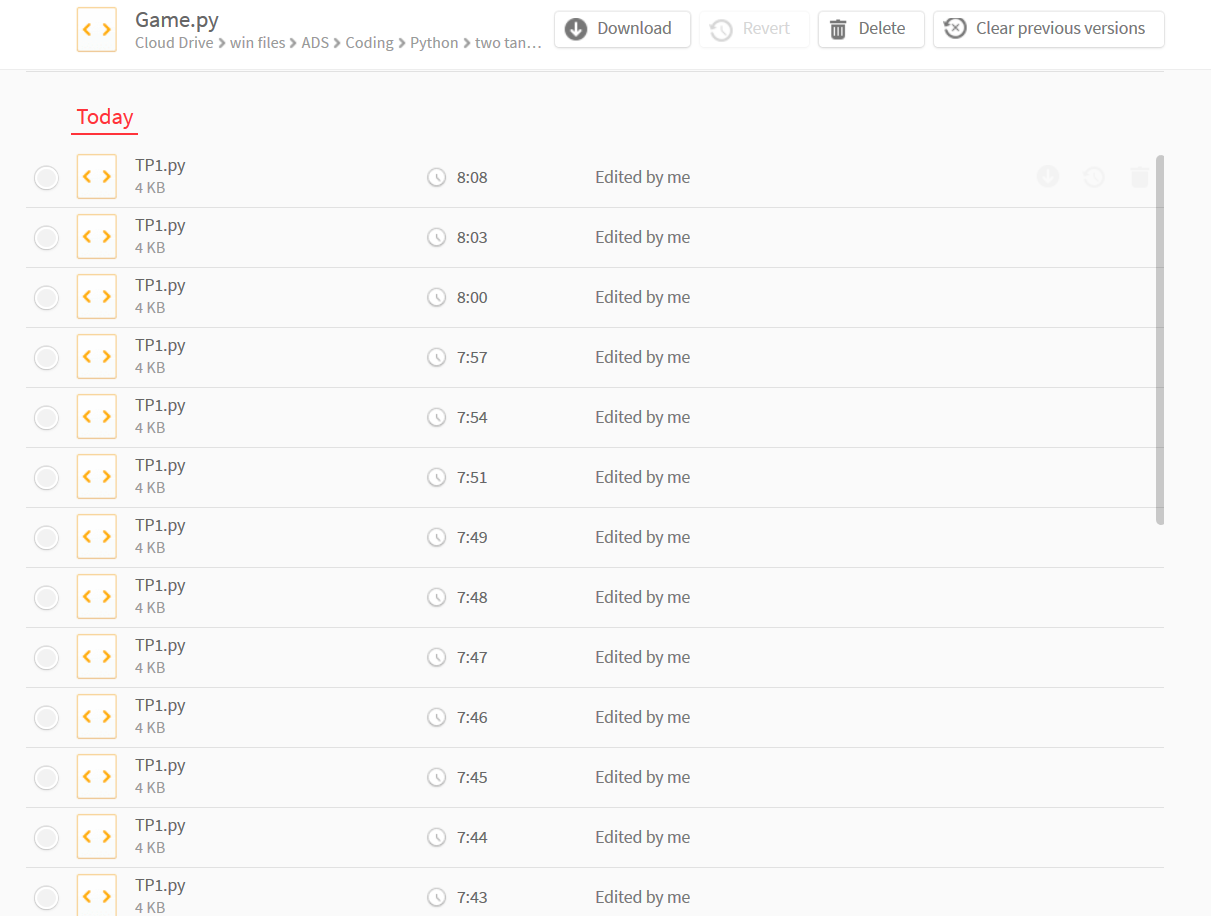
The trickiest part of the project is the game AI in the single player mode. I will approach this problem by reviewing the minmax algorithm covered in the optional lecture and try to design a workable game AI.

Timeline Plan:

I plan to finish the two players mode in TP1 and finish user interface and score tracking and upgrading in MVP. I will focus on Game AI and BGM in additional features.

Version Control Plan:

I am using a cloud drive to backup my codes.



Module List:

Pygame

**TP2 Update:**

Structural Plan:

(each class is an individual file)

AITank(GameObject)

This class is the AI tank in the single player mode. All the calculating algorisms is in this class

TestBullets(GameObject)

This class is the invisible bullets that AI tank will constantly shoot out, to test which way to go. Sort of like bats use ultrasonic wave to test location.

New folder: music

I added a background music to the game

New folder GameObjects:

I put Map, AITank, TestBullets and all the other class that inherent GameObject into this folder.

**TP3 Update:**

Structural Plan:

(each class is an individual file)

EasterMode()

This class generate the easter game. It is triggered by shooting the title “crazy tanks” 5 times. It will increase difficulty every 1000 points the player get and change background color every 100 points. Change in difficulty includes, bigger size and faster sending bullet frequency.

AITank(GameObject)

I improved tank AI a lot since TP2. Right now it has two modes: istracking and is avoiding wall mode. When AI tank is tracking, it will change its direction toward the player’s tank and move forward. As soon as it hit a wall, it will switch to avoiding wall mode. In this mode, the tank will send out two testing bullet the left and right side. Using the time the bullets take to get back, AI tank will determine which side has more space and turn that side. After it turned, it will constantly send out test bullets to the side that is next to the wall. When the time it takes to get back changes, this will indicate there is an opening in the wall. Thus the tank will turn to that direction and switch back to is tracking mode.

It will also shoot out 8 test bullets constantly, to 8 different direction. If any bullets hit the player’s tank, the AI will send a real bullet.

Tank3(GameObject):

This class create a tank object that is controlled through mouse. It will be used in threePlayerMode and singlePlayerMode

ThreePlayerMode():

I added the three player mode that allow the third player to use mouse the compete with the other two.

RankScreen()

This class allows user to input their name at the end of the game, and show the ranking of the player. It works in both usual game mode and Easter egg game mode.

I also added adverse powerups like Biggersize to each tank, along with the positive powerups like fasterSpeed, smallerSize

New folder: music/effects

I added sound effects for shoot bullets and tank explosion

I also added sound effects for powerups