

- **Design Proposal**

- **Project Proposal**

- **Project Description:**

- My project is a game named advanced minesweeper. It will be a more complicated minesweeper. In this game, hidden monsters will be mines. When the explorer detonates where they hide, they would appear and the explorer would lose one life. All monsters can move and some of them can even shoot. The explorer needs to find the way to exit the maze and collect coins as much as possible.

- **Competitive Analysis:**

- General minesweeper: The game is played by revealing squares of the grid by clicking or otherwise indicating each square. If a square containing a mine is revealed, the player loses the game. If no mine is revealed, a digit is instead displayed in the square, indicating how many adjacent squares contain mines; if no mines are adjacent, the square becomes blank, and all adjacent squares will be recursively revealed. The player uses this information to deduce the contents of other squares, and may either safely reveal each square or mark the square as containing a mine.

- **Structural Plan:**

- Object Monster

- attribute location

- attribute direction

- attribute speed

- function move

- Object ShootingMonster inherits from Monster

- function shoot (Monster)

- Object Explorer

- attribute name

- attribute location

- function shoot

- Map

- **Algorithmic Plan:**

- First: generate a map at random

generate hidden monster under the floor at random

- Use random library to randomly place them
- Number of monsters should be determined by field and level of game
- Distinguish monsters based on whether they can shoot or not

generate coins randomly

Second: determine when they appear and where they can move

- Monsters don't appear until the explorer detonates them
- Use timerfired function to control movement of monsters
- Use two methods in two classes for shooting between the explorer and shooting monster

Third: clues appear if no monsters are detonated

- Use some algorithm to calculate the number of hidden monsters around each floor
- Use some algorithm to make sure the maze legal
- Finish the tools part
- Try to use some algorithm to let monsters chase the explorer

- **Timeline Plan**

04/15-04/21: finish the above first step and half second step

04/21-04/28: finish the rest second step and the third step

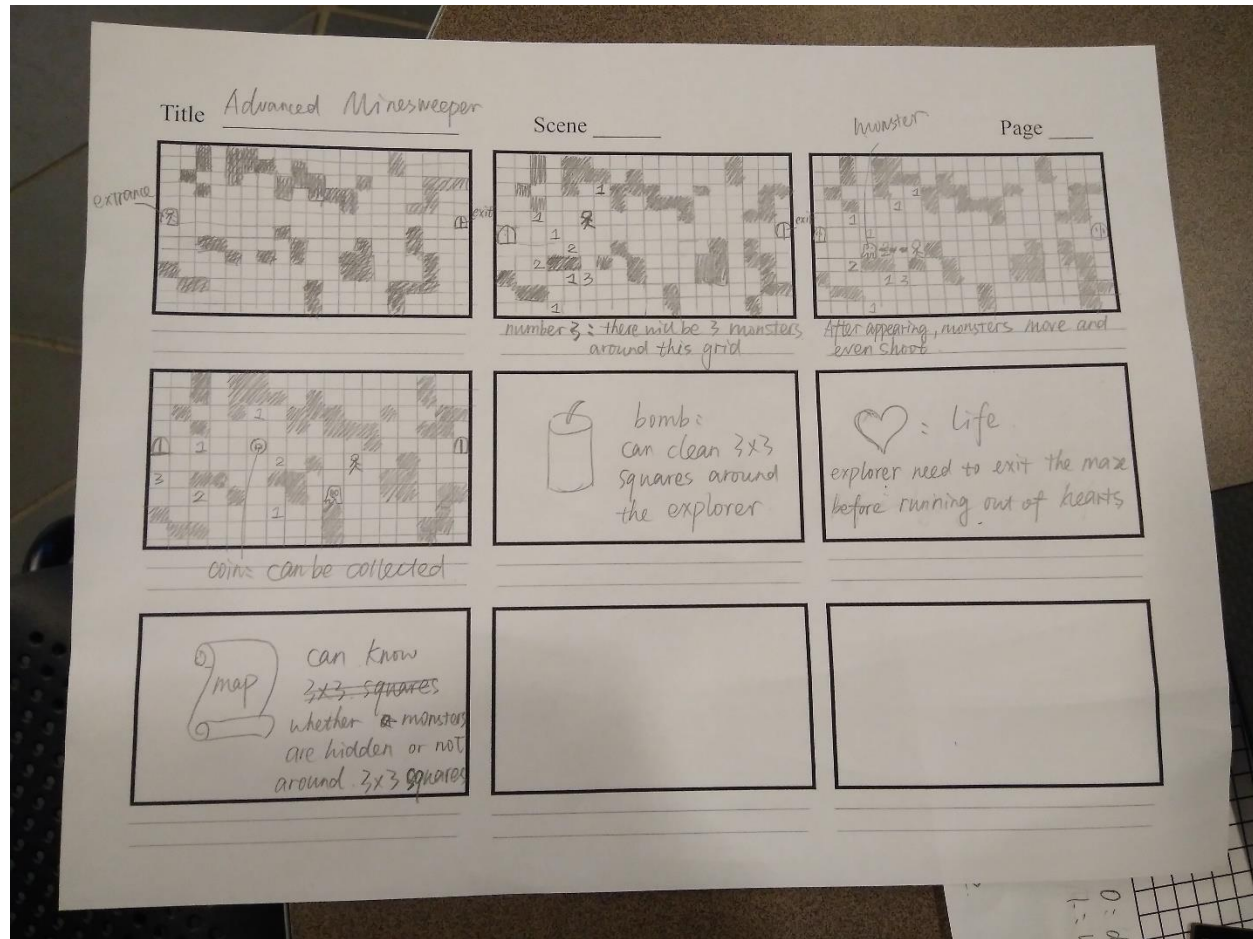
- **Version Control Plan:**

Upload my python file into google drive each two days

- **Module List:**

pygame(just for game music) and tkinter

- **Storyboard**



- **Preliminary Code:**

See python file – “minesweeper\_tp3.py”