Name: Kyle Chen

Design Proposal [20 pts]

Project Description [2.5 pts]: The name of the term project and a short description of what it will be.

Tomb of the Mask: Pacman with a twist; you're a tomb raider that needs to escape with your riches! Avoid traps and monsters to make it out of the maze alive.

Competitive Analysis [2.5 pts]: A 1-2 paragraph analysis of similar projects you've seen online, and how your project will be similar or different to those.

This is an actual arcade game that can now be played on mobile devices and computers as well. Rather than making this game an exact replica of what can be found there, I'm going to add my own player sprite designs, enemy types, ways for the players to interact with the maze, etc. So the concept of the game will be similar to what can be seen online, but with my own twists. I was also thinking about a way to implement a competitive mode for this, where the player can try to outlast the AI (which would just require some pathfinding and automatic movement) but this will be something I do if I have extra time.

Structural Plan [2.5 pts]: A structural plan for how the finalized project will be organized in different functions, files and/or objects.

I currently have a main py file where all the helper functions and classes are (and thus where the actual game will be played), but I will also include my tester files that helped me code up the more algorithmically complex parts of my code.

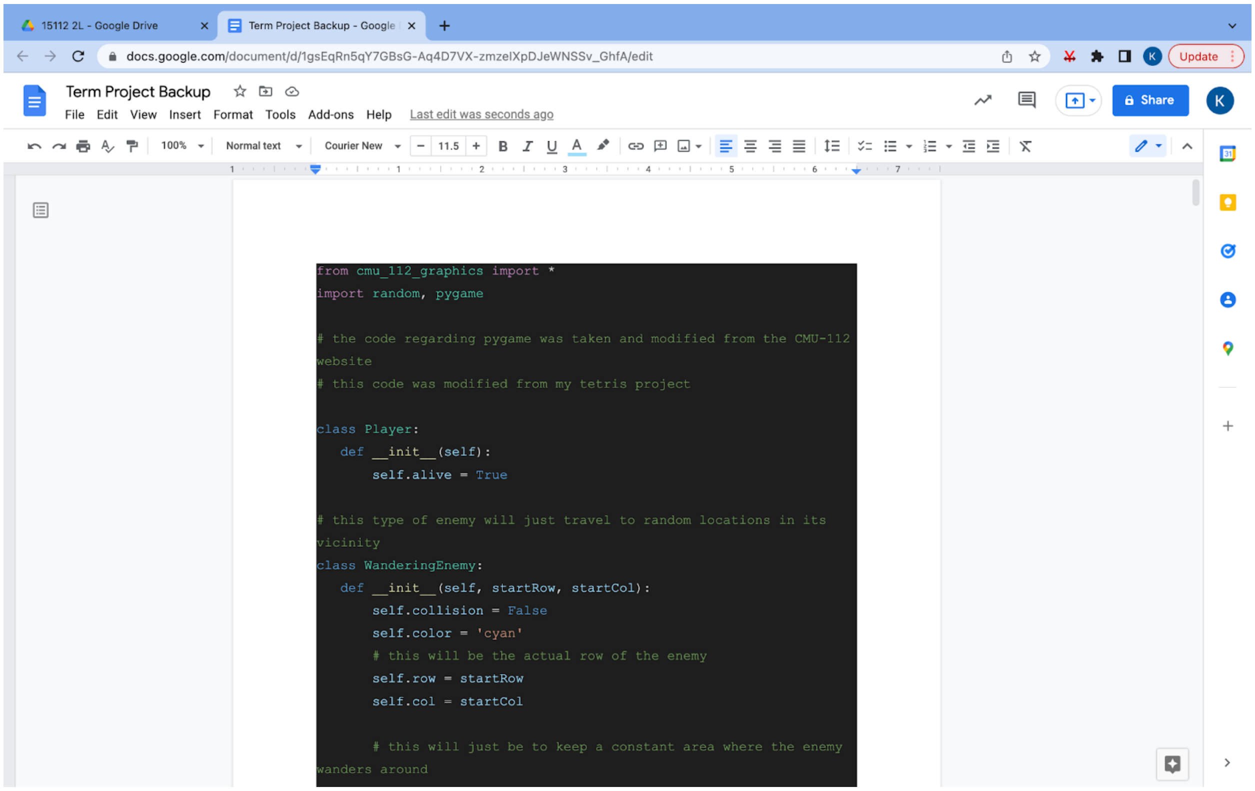
Algorithmic Plan [2.5 pts]: A detailed algorithmic plan for how you will approach the trickiest part of the project. Be sure to clearly highlight which part(s) of your project are algorithmically most complex, and include details of the algorithm(s) you are using in those cases.

As mentioned above, I created two separate tester files for pathfinding and maze generation, since these two are the trickiest parts of my project. I got stuck on maze generation so I finished pathfinding instead, so I'm taking each one step by step and really breaking down the process.

Timeline Plan [2.5 pts]: A timeline for when you intend to complete the major features of the project. I'll finish the major features before MVP (this Friday, Apr 22, 2022)

Version Control Plan [1.5 pts]: A short description **and image** demonstrating how you are using version control to back up your code. Notes:
You must back up your code somehow!!!
Your backups must not be on your computer (ideally, store them in the cloud)

I just copy pasted them into a google document on my google drive. This will be updated daily.


15112 2L - Google Drive
Term Project Backup - Google
docs.google.com/document/d/1gsEqRn5qY7GBsG-Aq4D7VX-zmzelXpDJeWNSSv_GhfA/edit
Update
Term Project Backup
File Edit View Insert Format Tools Add-ons Help
Last edit was seconds ago
Share
100%
Normal text
Courier New
11.5

```
from cmu_112_graphics import *
import random, pygame

# the code regarding pygame was taken and modified from the CMU-112
website
# this code was modified from my tetris project

class Player:
   def __init__(self):
       self.alive = True

# this type of enemy will just travel to random locations in its
vicinity
class WanderingEnemy:
   def __init__(self, startRow, startCol):
       self.collision = False
       self.color = 'cyan'
       # this will be the actual row of the enemy
       self.row = startRow
       self.col = startCol

       # this will just be to keep a constant area where the enemy
wanders around
```

Module List [1 pts]: A list of all external modules/hardware/technologies you are planning to use in your project. Note that any such modules must be approved by a tech demo. If you are not planning to use any additional modules, that's okay, just say so!

I'm only using pygame as a way to add sound effects and background music (passed the tech demo!).

Storyboard [5 pts]
Generate a storyboard that demonstrates how a user would interact with your finished project. Your storyboard should have at least six panels, and at least three of those should demonstrate features within the project. You may scan or take a picture of your storyboard and include it in the directory as the file storyboard.png (other acceptable filetypes include .gif, .jpg, and .pdf).

- Buying things from the store (clicking)
- Left, right, up, etc. to control the characters
- Pause button
- Different modes to choose from

- Include an **updated design document** with all your design document information from TP1, and a new section called 'TP2 Update' with any changes you have made since TP1. If you haven't made any changes, still add this section in, but just say "No changes made" or something similar. If you do not have this update, **you will be penalized.**

TP2 Update

TP using DFS:

- Infinite maze generation while sidescrolling using DFS
- Lose condition (getting touched by the of the lava) implemented

TP using Kruskal’s

- Able to get Kruskal's Algorithm to work
- Generated the entire maze and working on sidescrolling
- Updated pathfinding algorithm to suit Kruskal’s

TP3 Update

- Lose conditions have been implemented with the Kruskal’s iteration of term project
    - Pathfinding enemies
    - Wandering enemies (go up and down a row or col)
    - Added sprite animation for the enemies
    - Lava moving up the maze to keep you going up
- Infinite maze sidescrolling and generation, and difficulty increases as you solve more mazes (but it caps at a certain level so that it doesn’t become impossible to solve)
- Shop has been implemented with working powerups
    - Different costs for each power up
    - Is able to check if you have enough currency to buy, and if bought then it will not allow you to buy another time
- Powerups:
    - Shield - Protects player from death once
    - Magnet - Collects all coins and points within a 3x3 array around the player
    - Coin Addict - Turns all points into coins for one round
    - Score Multiplier - Adds 5 for each point you get
- Made the layout and design of the home screen and game look nicer and close to the actual game
    - Able to restart game using “r”
    - Implemented a “Best Score” variable
    - Is able to keep track of best score and current coins when restarting game