Gerald Kim

15112 Term Project

**Project Description**:

The project is currently named Portal Runner. It is a side-scrolling platformer taking place in the Portal City, a steampunk world relying on portal technology. The user plays as a fugitive who stole a valuable device from the government - one that grants the holder the power of teleportation. To win the game, the user must traverse 5 randomly generated levels of the Portal City while avoiding or defeating the city guards with the help of the stolen machine and other portals and power ups.

**Competitive Analysis:**

Games like Geometry Dash and Super Mario Bros/Super Mario Maker are similar to mine. Both games are side scrolling games, but the side scrolling aspect of my game is more similar to Super Mario Bros because Geometry Dash utilizes a continuously rightwards moving camera, making the user unable to control the horizontal movement of their character. Super Mario Maker and Portal Runner will give control of the horizontal movement to the user. Also, the existence of enemies in my game, Portal Runner, makes that aspect similar to Super Mario which also contains enemies. However, the main difficulty of my game will be platforming and less focused on enemies, which is similar to Geometry Dash which focuses only on the platforming. The main difference between my game and those games will be the procedurally generated levels, while the other games have level builders, which I may implement after MVP.

**Structural Plan:**

I will have a main file that runs all other files. I’ll also most likely have a separate file for the character class, which will hold all relevant variables for the playable character. Within my main file, I will have several different functions. These functions will include checking if a player is on a surface, if a player has collected items for a powerup, enemy movement, checking if a player has collided with an enemy, and game over states. My pathfinding algorithm and randomly generating levels code will take their own separate files which will then be imported to the main file for use.

**Algorithmic Plan:**

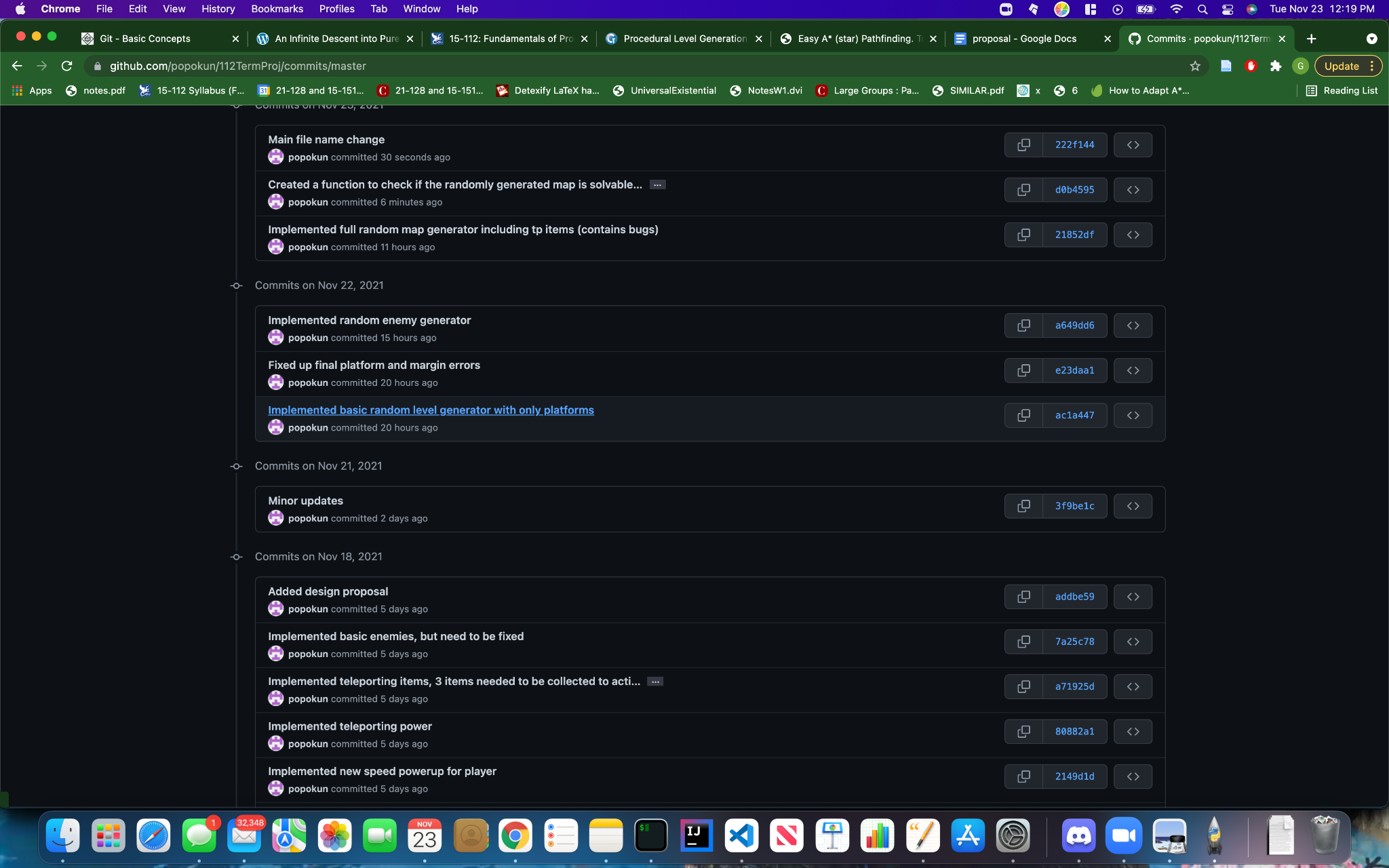
The most algorithmically complex part of my project is randomly generating levels and if they can be solved. For the randomly generated levels, I plan on creating a grid and randomly generating coordinates for all obstacles, items and enemies. These coordinates will be on the grid. To see if the randomly generated level is solvable, I plan on utilizing a pathfinding algorithm. One important factor I’ll need is the maximum jump distance of the character to see if I can get from one platform to another. If all the platforms are connected (we can find this by using the pathfinding algorithm), then the player will be able to interact with the level.

**Timeline Plan:**

I plan on implementing a rudimentary version (contains bugs) of the basic gameplay by TP1, i.e. items, enemies, and physics. I will implement a rudimentary version of the randomly generated levels that are beatable by TP2. By TP3, all bugs in the basic gameplay and the randomly generated levels will be fixed, UI will be improved to match the aesthetic of Portal City.

**Version Control Plan:**

I will use git to backup my code on Github.



**Module List:**

I do not plan on using any additional modules

**TP2 Update**

**Design Changes:**

Instead of completing 5 levels, the user now needs to complete only 2 levels to beat the game. To pass these levels, the user now must collect the three items to obtain the teleportation power.

**TP3 Update**

**Design Changes:**

In addition to collecting the three items, the player must traverse all platforms to beat the level.