## 

## TP-3 Update

The design changes were largely on the path I was on before. The enemies are more developed because they are now able to weaken you depending on how many defense cards are in your draw deck. I have a shop room in which you can buy cards, upgrades to your characters and upgrades to your cards. My treasure room also gives you random item from the shop but for free. I can also retain the high score with the txt files for each difficulty. The map now generates recursively depending on what room is above it. The choice of rooms we have for each difficulty also changes. In addition to this, once a room is cleared, all other paths that are non-adjacent are deleted from the game (and non-adjacent to the next, recursively). You are also able to select an attack card and pick the target you would like to do the attack to. The graphics are significantly more polished, and the game scaling works for the health of enemies, amount of gold gained, amount of damage, and how good the upgrades are.

## TP-2 Update

The design changes were largely through practically of how easy things would be to create. The main thing I did different than my plan is I made the deck that you play out of similar to the deck view for the discard pile and draw pile. You just click on the card, and it will do its action. I will probably plan to only make one character, considering I find it more worth it to continue to try and develop the shop, treasure room, and scenarios. I have a win game and lose game method that works appropriately.

## TP-1 Update

Currently no large design changes so far. Some of the emblems will look different from what I drew in my storyboard.

## Proposal

Term Project: Algorithmic Ascension

The project is called Algorithmic Ascension, which will be a Slay the Spire remake with 15-112 concepts. Slay the Spire is a single-player roguelike deck building game, where the player encounters several enemies and situations. As the player progresses through the spire, they gain cards that can be used to build their deck. The goal is to beat the final boss at the top of the spire. Algorithmic Ascension will not capture all the ideas of Slay the Spire, but will capture the main combat method and win condition of Slay the Spire. There will be different characters, and throughout the game you will be allowed to upgrade your character. You will also gain cards throughout the game. These cards can be used with any character, but each character will have different attributes that occur when you use the card. For every unique game, the map structure will be randomized with a few common attributes, depending on the difficulty chosen. The map will have either recovery sites, enemy rooms, unknown rooms, shops, or treasure chests. Unknown locations either have any of these things, or a situation in which you must make a decision.

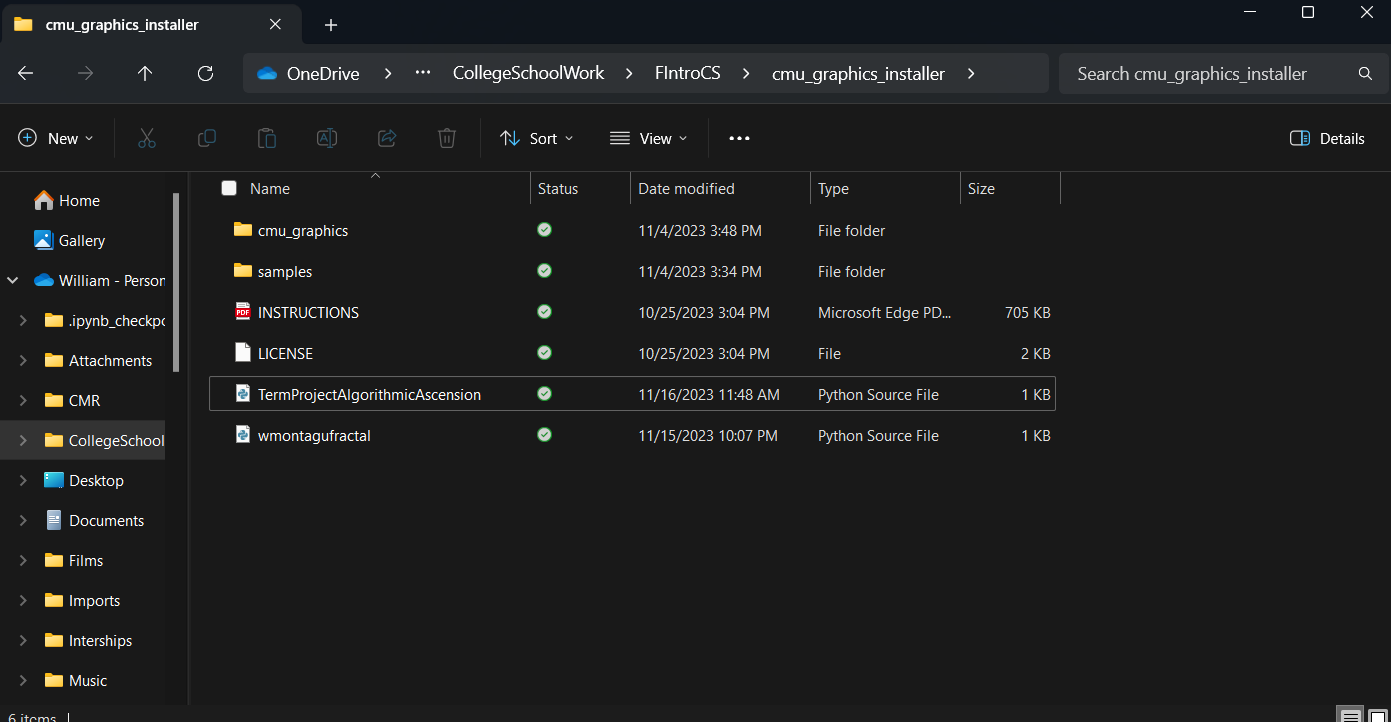
The combat method and win condition of Algorithmic Ascension can be seen similarly in many games such as Slay the Spire, Pirates Outlaw and One Step From Eden. The main similarity is the genre, while the differences between the games are more distinct. My game differs from Slay the Spire, through the way characters are upgraded throughout the game, rather than gaining items which empower them. In addition, Slay the Spire has specific cards for each character, rather than retaining the same cards for different characters. With Pirates Outlaw, the game's combat system requires you to recharge your mana with the cards, rather than receiving full mana each attacking turn, which I plan not to do. One Step From Eden differs through the combat system, which is more fast-paced, with each character having its own basic attack, but having the same set of cards (like Algorithmic Ascension).

Structurally and functionally, I believe the project would contain several classes and functions within the class. The main would mostly pertain to traversing the map. The classes would include but not limited to, Enemy, Character, Deck, Card. Each enemy would have another class, which would classify what type of Enemy it is. The character class would have methods to play cards, and would also include the character attributes. These attributes would be chosen according to which character the player chooses. The Deck would have methods to add and remove cards, and shuffle the cards. It would also have another variable that would pertain to the discard pile in combat, which would be empty at the beginning of every combat. Throughout the combat, if the deck is empty, we would shuffle the discard pile and make the discard pile the new deck.

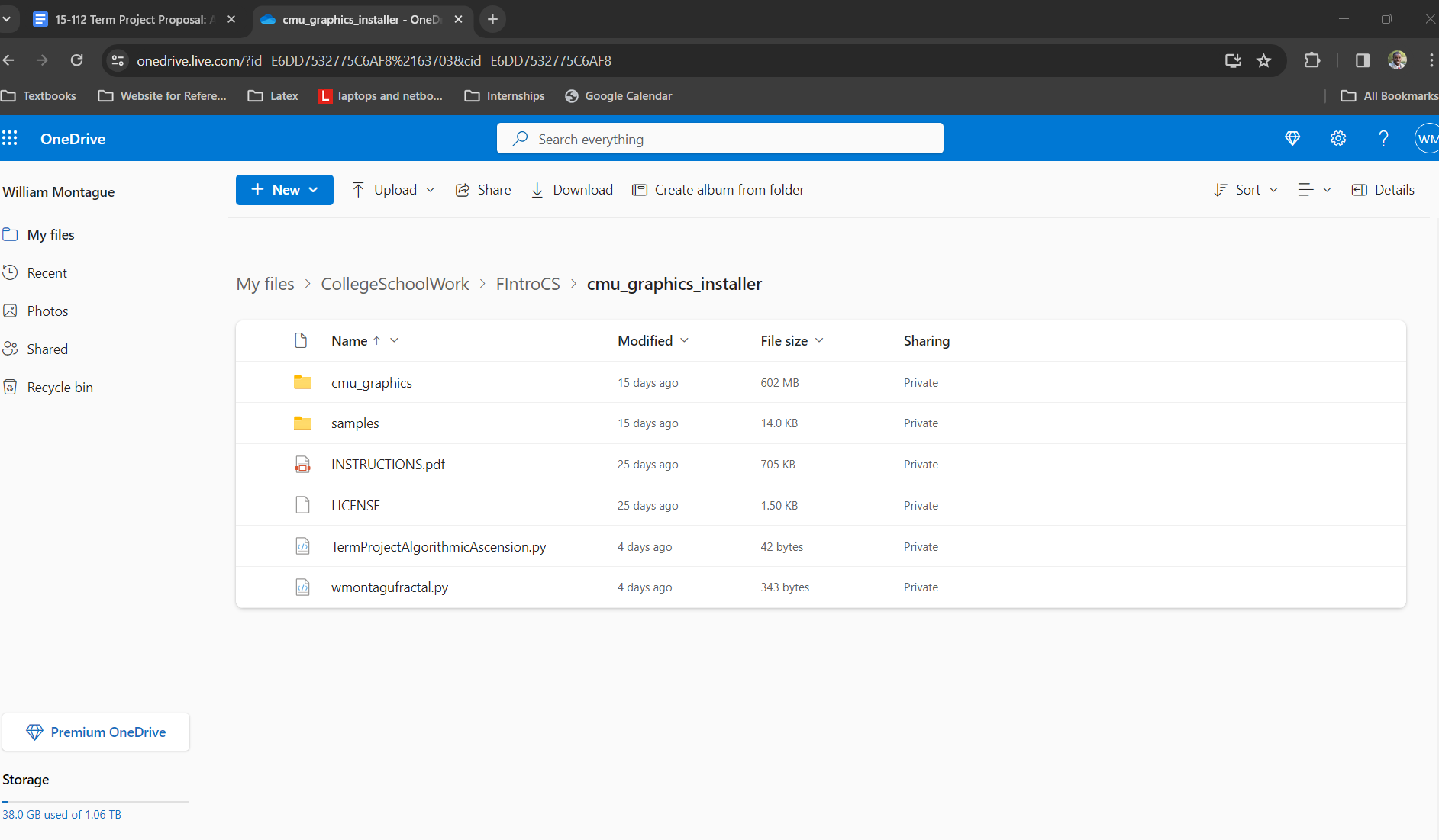
The trickiest part of the project would probably be the map, and randomizing for different difficulty levels. It is tricky because the map is visually hard to relate to the code, but also because the order of different aspects matter a lot. I plan to have one map, where if you reach the top and beat the boss, you win. I would need to know how to randomize such that each easy map is as easy as the other one, because your health would not reset between combats. This means that I would need a recovery site, treasure chest, and shop to occur on a similar level on each easy map, and the rest can be randomized to either enemy rooms or unknown rooms. I will get around this by allocating time to work on the map of the game, and asking for help in animations to represent each of the objects on the screen as part of the map.

This project is due on December 6th at 5pm. TP1 is due on Monday, November 27th at 5pm and TP2 is due Friday December 1. Through TP0-TP1, I will work on the main functions and classes that are required for the game (Deck, Cards, Characters, etc.), and do preliminary animations. I plan to dedicate one and a half hours a day over thanksgiving break (totaling 7.5 hours), to do this. Through TP1-TP2, will be my main animation phase, in which I continue to work 1.5 hours a day (totalling 10.5 hours). Through TP2-TP3, I will finish animations and compile the map with aspects of combat and finalize the project with feedback.

I plan to use my personal OneDrive for version control. I can simply save the project to the OneDrive folder I keep my school work in, and when I have internet access, the latest save will be uploaded to my OneDrive. My OneDrive is safe with Two Factor Authentication, and I have never experienced problems with saving on it.



I can also access this folder on the web through the OneDrive site.



I am not planning to use any additional modules.