

## **Final Project Reflections – Stones, Cardboard, Cutters**

### **Design Description:**

There are seven .cpp files that are needed for this game.

1. The main function
2. The dynArr container
3. The Tile parent class
4. The PlayerTile child class
5. The EnvironmentTile child class
6. The DeathTile child class
7. The Map game class

#### The Main function:

The main function will serve a simple purpose. The actual game functions will be implemented in the Tile and Map classes. The main function will simply serve as a driver function.

#### The dynArr container:

The dynamic array container will be used to implement a bag for the player. It will allow for various items to be stored.

#### The Tile parent class:

The Tile class is an abstract parent class that just serves as a template for its children classes. It represents the different kind of rooms that will be available to be explored by the user. It will store information about its four neighbors (above, below, left, and right), have a run() function to perform the room's actions, and a couple other functions to get some of the details about the room.

#### The PlayerTile child class:

The PlayerTile class will represent the boss rooms in the map. The user will have to play rock, paper, scissors against a boss in this kind of room. The user must beat all of the bosses in the map before the time runs out in order to win the game. The user will use the items they have in their bags in order to fight these bosses.

#### The EnvironmentTile child class:

The EnvironmentTile class will represent the supply rooms in the map. In these rooms, the user will have an opportunity to pick up one of three items (rock, paper, scissors) to be used in boss battles. However, the user will only be able to hold three items at a time and must drop one item to pick up a new one.

#### The DeathTile child class:

The DeathTile class will represent rooms in the map that gives the user a high chance of instantly losing the game. If they happen to wander into one of these rooms, they will have to try to guess a random number between 1-10 in order to stay alive. This is more like a trap room.

### The Map game class:

The map class holds a grid of these rooms. It will also have functions to run the actual game and provide a menu for the user to be able to navigate the rooms.

### **Problem Resolution:**

Surprisingly, this was the one assignment that I encountered the least amount of bugs and obstacles in. Throughout the whole process, other than working out very minor syntax bugs, I didn't have any trouble with any major bugs. I believe that the major reason for this is that there wasn't really any new material in this assignment. I had a lot of experience struggling through the different concepts in past assignments, but I always made sure I understood each concept before the respective assignments were due. This helped me to smoothly produce this game.