3D Maze Escape

**Authors**: Ameya Gandhi, Abbas Dohadwala, Aayush Lakhotia

**Revision**: 4/25/21

**Introduction**:

For our project, we are planning on creating a 3-Dimensional maze game with interactions with the player. Our program generates a 3-Dimensional maze with various puzzles and puts you in a 1st person view of the maze. You have to wander through the maze and try and find the exit to the maze. We wrote this program based on the challenges we had with the Maze challenge in the recursion in 2-Dimensional arrays lab. However, we wanted to put our own spin on it, and decided to create a 3-Dimensional maze as it would be more challenging and we would learn how to render blocks in the third dimension. You enter a maze-like place where if you don’t escape in a certain time then the whole world will fall into chaos. The main rules are you can’t see the whole maze from a birds eye view, there is a time limit to solve the maze, you cannot go through the walls, and you have to get around traps and puzzles that may send you all the way back. The goal is to escape and save the world because for some reason the evil mage has his power source stuck inside the maze. When you escape the maze, the power source is destroyed and the mage loses his powers. The players should be around high school age to use the program as the memory skills and logic required will have to be strong to successfully complete the puzzle. There will be a way to exit the current session in the case that you want to reset the maze after you change the difficulty or some other option.

**Instructions**:

[Explain how to use the program. This needs to be specific:

Which keyboard keys will do what?

The space bar will move you the direction you are looking. If you need to turn, use your mouse to face that direction.

Where will you need to click? Keys Clicked will be the spacebar and the escape key (esc on top left).

Will you have menus that need to be navigated? What will they look like?

There will be a menu for the difficulty of the maze and it will look like a regular menu of someone who enters a game and wants to choose the difficulty on a GUI with two levels. There will also be another menu that can be accessed mid game to reset.

Do actions need to be taken in a certain order?

Actions do need to be taken in a certain order as you need to respond to the necessary prompts like the difficulty and perhaps the mode before you start.

**Features List (THE ONLY SECTION THAT CANNOT CHANGE LATER)**:

**Must-have Features**:

* A feature we need is a menu screen and a home screen. The home screen will have the options between the multiple levels. There will also be a menu screen ingame that will take you to the home screen.
* The maze will be viewed in first person. There will be a renderer that updates the player’s view of the 3D maze.
* Traps in the maze which will send the user to the start of the maze without the timer resetting.

* There will be multiple mazes of different difficulties. The user will be able to choose which maze they want to do.
* There will be a timer that displays the amount of time you have left to complete the maze.

**Want-to-have Features**:

* A score system which is based on the amount of time you finish with. It will be like a standard scores system with three stars being the best.
* There will be a puzzle at the end of the game to exit the maze. There will be a riddle that we found off the internet that will need to be answered to exit the game.
* An optional map for a single layer that you can toggle on and off at the menu screen.
* The maze could be randomly generated. This way, each level will change every time so if you fail the first time it will be slightly different.
* A multiplayer mode that can be played with another player in a race type situation.

* Interactions with the maze that will need to be made in the correct order to solve the maze.

**Stretch Features**:

* Graphics optimization so that the program isn’t laggy.
* A Leaderboard system which records all scores across the program users and displays them in a ranking of the top 10.
* Credits screen that displays the high scores and the names of the creators.

**Class List**:

Menu - GUI for the menu screen

Main - the main java class

Traps - Creates the various traps throughout the maze

LogicGates - The places where user has to solve logic puzzles

Maze - keeps track of the maze is a 2D version

Player - All controls of the player

Renderer - To update the player’s perspective as they move

**Credits**:

* Abbas Dohadwala: Rendering of the maze, generation of the maze
* Aayush Lakhotia: Create the traps, logic puzzles
* Ameya Gandhi: README features list, help Abbas render the maze, work on controls for the user
* We will use processing.js for the 3D part, using eclipse to write the program.