**Programming Project Report**

Student Blake Williams

Student 010974718

**Problem Statement:**

The Goal of this project is to take in a inputted .txt file containing information about a maze structure. Then use various Open GL commands to draw, position, and render textures based off the read in .txt file. The only user inputs will be allowing them to translate and rotate the maze to get a better view of the display.

**Design:**

To do this First the maze.txt has the read in the needed values to properly render the maze. Then the textures for each of the cubes must be rendered in, then propery sized and placed into the right location. To do this an array will be created to store xy values and then used to identify what texture needs to be rendered where.

**Implementation:**

To implement this, first the readFile() function was created which reads in a filename. Uses ifstream to open the file. Then the first 2 rows values are read into separate individual variables. These contain the information on how big the maze is, and where the player cube starts. Then 2 for loops get each value and store it into an array called maze[i][j]. So we now have a xy coordinate plane to map the single key values to dictate the texture and position used In the rendering. After the maze.txt is read in, the next modification was to the display() function. Here the maze is physically rendered. This is done by another nested for loop which the i values represent y coordinates and the j values represent x coordinates. Then inside this for loop 4 if statements handle the positioning of the blocks by checking the value of maze[i][j] matches the inputted key from the array. If it does then the if statement executes, rendering the cube at that position. The way the cube is rendered in the maze is by first calling the glTexImage2D() function which defines the texture used in the render, which is the last inputted value. Then the block() function is called and passed in xyz values based off the i and j of the nested for loops. These combined to create the render of the maze.

**Testing:**

To test this program first an initial render was made to make sure cubes were rendering properly. Then skeleton methods were made to detect the 3 main texture types (b = Brick , r = Rock ,w = Wood). Then starting with brick the method for rendering brick was implemented, run, and checked to see if it properly rendered the right brick cubes in the right spots. This was repeated for the remaining texture types. Then for the player cube, a skeleton method was created to test if the player starting values were able to be recognized. Once the player starting values were recognized by the if() statement, then began the trial and error of scaling down the player cube. It was discovered that for the xmin,ymin,zmin you could do (xbase+offset, ybase+offset, offset) and for xmax,ymax,zmax(xbase+offset / size, ybase+offset/size,offset/size) to scale the cube down to the right size. Then another texture was created and rendered on top of the new cube. Once this was completed the project was completed. Everything worked as anticipated with only minor changes made. A screenshot of a video game

Description automatically generatedA screenshot of a video game

Description automatically generated

**Conclusions:**

The result of the project was success as the maze was properly read in and rendered. Next time I will look at different methods of rendering the cubes using OpenGL calls. This project took 7hrs to fully complete including the report.