CS324 Coursework Assignment

Krzysztof Janusiewicz

January 16, 2017

1 Introduction

2 Compiling and Running the Program

In order to compile the program on linux, simply navigate to the main directory and use the makefile by running.

```
make
```

To run the program:

```
./maze
```

3 Using the Program

To turn the camera left and right, use the 'a' and 'd' keys respectively. To move forward and backward, use the 'w' and 's' keys.

4 Design

4.1 Tree Structure and Recursive Functions

A tree-like system of objects of the class game_object has been used to represent the graphical elements of the maze. Each game_object object has a pointer to its parent, and an std::vector¡game_object¿ containing its children, as well as physical information such as position, rotation and scale. Whenever some amount of time passes and objects need to be updated in regards to their position, velocity and other such qualities, the update() method can be called on the root game_object. This will then recursively update each of its children, and the same will happen for other methods such as display().

4.2 Graphical Component Inheritance

Each game_object also has a game_component object. This class is inherited by multiple others, namely graphics_object, textured_graphics_object and light_object. In the future, the implementation could be extended to include a camera_object, which would allow for the camera to move together with other objects and graphical components such as 3D models and lights. The nature of the inheritance means that each game_object within the tree can have a light, 3D model or the camera attached to it. In the future, game_object could have its game_component member changed to a std::vector¡game_component¿ for ease of adding multiple components to one object.

5 Features of the Program

6 OpenGL Features Used

6.1