FUNDIMENTALS OF Games and Graphical system development

Creating A 3D scene

Table of Contents

[Key Words 1](#_Toc39145226)

[Description and Instructions 2](#_Toc39145227)

[UML Tree 3](#_Toc39145228)

[Critical Reflection 4](#_Toc39145229)

# Key Words

C++

3D Scene

Coding

Tutorial

# Description and Instructions

In the 3D scene I have created is cubes that fly towards the screen, if I were to turn this into a game it would be a concept for a 3D version of the classic 1979 Asteroids game. However, this is a 3D scene with some user input.

In this scene you can move the camera and rotate the cubes the instructions to do so are as follows.

To Move Cubes;

A key – Rotates the cubes left

D Key – Rotates the cubes right

To Move Camera;

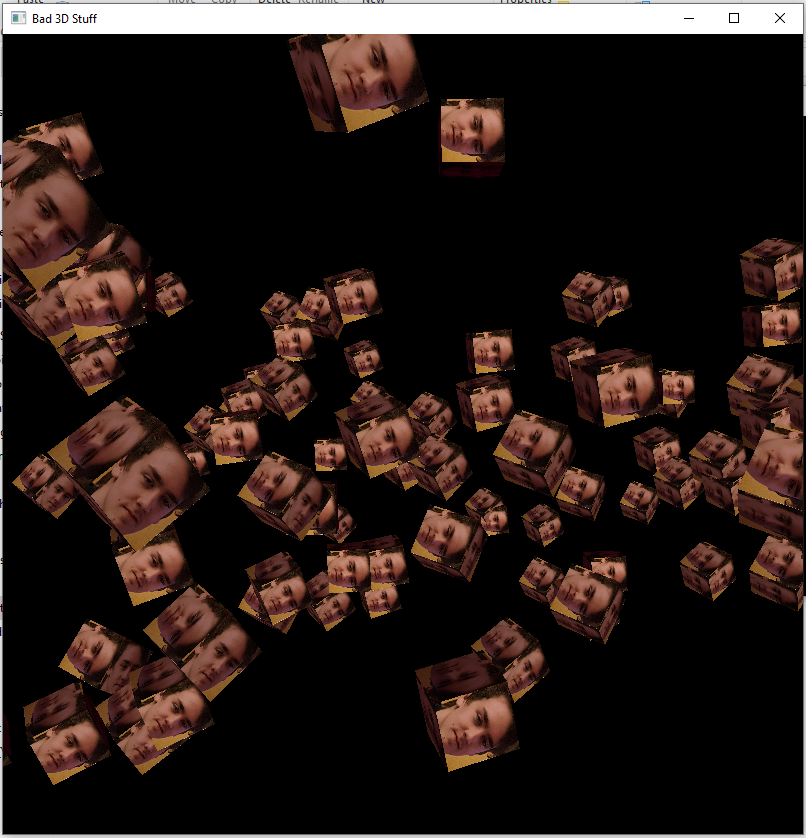
J Key – moves camera to the left

L Key – moves the camera to the right

I Key – zooms the camera in

K Key – Zooms the camera out

There is no current objective in the scene as it is just a 3D scene currently but as stated above it could be turned into the Asteroids game and have similar scoring and player movement with a 3D element. There is a screenshot below.



# UML Tree

# Critical Reflection

In Fundamentals of Games and Graphical System Development we were tasked to create a 3D scene in semester two. This task for me was difficult as I had never coded before this apart from the 2D game created in semester one so 3D would be a new challenge.

I started out by following the tutorials and quickly realised how different the library and creation of 3D objects were compared to the 2D I had previously worked on. Despite this, I stuck to it and tried my hardest to get through the tutorials provided but I ran into a few problems each week which put me behind the class.

Some problems were silly errors such as how the code was laid out. Some were having includes in the wrong classes causing huge errors and breaking the code until I removed the wrong include which was a pain staking process for someone who is new to the coding scene like me. I worked through these errors and finally got to the point where I had completed the tutorials and I felt good.

Although my project is the minimum that was asked for, I now have a greater understanding of the programming language c++ and how versatile it can be in the right hands and I am happy I was able to complete the project to the point it is at even if it is basic.

Some good points to make is that I have learned a lot in the second semester of university and, although it was difficult at times, I have had fun building code from scratch and viewing something that I have created throughout the process. I feel that if I went back now with the knowledge I have I would create an infinitely better 3D scene than what I have. I managed to create the cubes to read from file which I could not do in semester one for my Pacman level so that was an improvement for me.

The algorithms in my code could be better. However, what is implemented works well enough for the program. The structures I feel I have done well as they are clear and labelled through code comments allowing for anyone to be able to look at the code and see what they do. Commenting on code is a skill I tried to pick up on quickly as it is easy to go back to some code and not understand what it does anymore so having those comment are helpful for, not only me, anyone looking at the code I have written. They also do exactly what they are meant to and do not have any extras that should not really be in them.

Overall, this semester has been a difficult but rewarding one. I have felt that I have constantly had work but at the same time I have had fun creating the projects and it is really rewarding to see your code run and work how it was intended. I have loved this university year as a whole and cannot wait to see what other tasks we will be given next year.