Computação Gráfica
(3º ano de LCC)

Trabalho Prático - Parte 3

Relatório de Desenvolvimento

Francisco Oliveira	Luís Costa	Maria Luísa Silva	Tierri Monteiro
(A82066)	(A70070)	(A82124)	(A76359)

4 de Maio de 2020

Conteúdo

Capítulo 1

Introdução

1.1 Contextualização

No âmbito da Unidade Curricular de Computação Gráfica, foi-nos proposto a realização deste trabalho prático. Este trabalho foi dividido em 4 fases, sendo esta a terceira. Por sua vez iremos mostrar as novas alterações feitas nesta parte.

1.2 Resumo

Este relatório inicia-se com uma breve contextualização, seguindo-se a descrição dos problemas propostos e o que deve ser desenvolvido para os solucionar.

De seguida são descritas as alterações efetuadas ás duas aplicações previamente desenvolvidas: o gerador e o motor gráfico. Para o gerador é indicado o método de obtenção de vértices e índices de cada primitiva, e para o motor gráfico são indicadas as alterações realizadas para suportar translações dinâmicas, VBOs e rotações dinâmicas.

Por fim, é apresentado a nova cena do Sistema Solar, terminando este relatório com uma conclusão sobre o trabalho desenvolvido, apontando os seus pontos fortes e fracos, bem como dificuldades sentidas.

Capítulo 2

Gerador

2.1 VOB's

2.1.1 Estrutura

Para esta fase do trabalho foi necessário usar uma estrutura previamente criada para o engine no gerador, chamada vertice, esta constituida por 3 floats. Como se pode verificar no exemplo a seguir:

```
class Vertice {
private:
    float x, y, z;
public:
    Vertice() {
        x = 0;
        y = 0;
        z = 0;
    };
    Vertice(float a, float b, float c) {
        x = a;
        y = b;
        z = c;
    }
    float getX() {
        return x;
    }
    float getY() {
        return y;
    }
    float getZ() {
        return z;
    }

    void setX(float value) {
        x = value;
    }
    void setY(float value) {
        y = value;
    }
    void setZ(float value) {
        z = value;
    }
};
```

2.1.2 Índices

Relativamente aos índices a nossa abordagem foi arranjar uma forma geral de obter todos os indices independete da forma que estivéssemos a querer criar , ou seja :

- Criar os pontos da figura e colocar estes num vetor;
- Criar um vetor igual ao previamente descrito mas sem valores repetidos;
- Encontrar os índices da figura recorrendo aos dois vetores;
- Imprimir os pontos e os respetivos índices;

2.2 Bezier

2.2.1 Leitura

O ficheiro que contém as informações para gerar os vértices do modelo Bezier em questão, está dividido em 4 partes fundamentais:

- Npatches: Número de patches obtido na primeira linha;
- IPatches: Obtido nas Npatches linhas seguintes, cada um com 16 pontos;
- NcontrolP: Número obtido na lina depois de todos os patches;
- ControlP: Obtido nas linhas a seguir ao NcontrolP;

Tendo em conta que a informação é representada desta forma, utilizadomos o getline para obter a primeira linha , guardando esse valor "psize". Foi criado um array com o tamanho 16*Npatches para guardar todos os índices de patches seguintes a serem lidos. De seguida é lido o NcontrolP e por fim guardamos as coordenadas para cada ControlP num vetor, da nossa estrutura Vertice , para posteriormente serem acedidos (através dos índices guardados anteriormente) e calcular os vértices do modelo a representar.

Capítulo 3

Engine

O engine é responsável pelo armazenamento da informação dos modelos a representar, assim como as respetivas transformações geométricas de cada um desses modelos, que no seu conjunto, irão formar o sistema solar que queremos representar.

Uma das mudanças feitas nesta parte do nosso trabalho foi a implementação de VBOs para desenhar os diferentes modelos.

Os VBOs fornecem-nos uma performance bastante melhor, devido ao facto de a renderização ser feita de imediato pois a informação já se encontra na placa gráfica em vez de no sistema, diminuindo assim a carga de trabalho no processador e assim os os pontos em vez de serem desenhados um a um, passando a estar todos guardados num *buffer*.

Capítulo 4

Resultados Obtidos

Nesta parte podemos visualizar o resultado da criação do Sistema Solar.

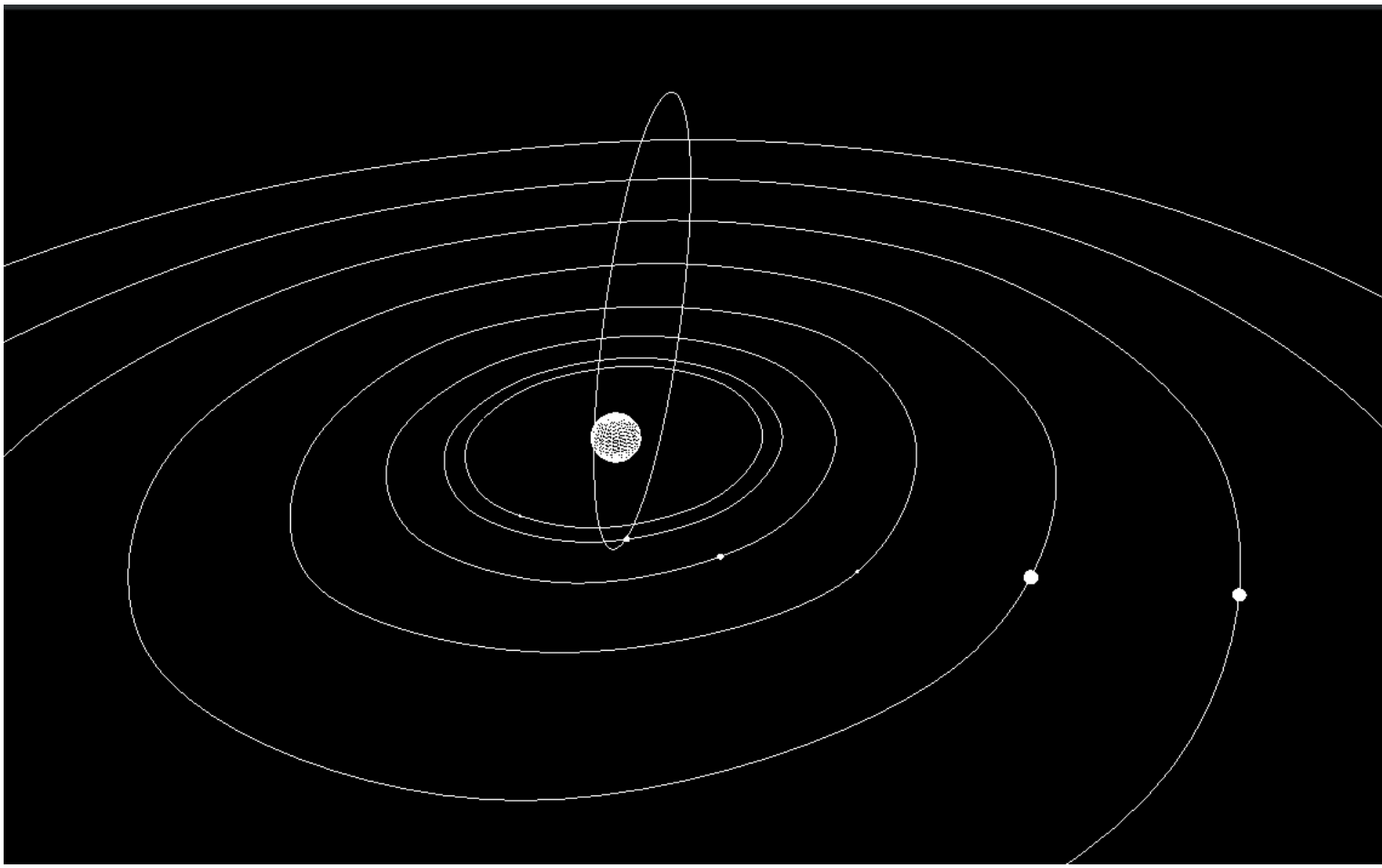

Figura 4.1: Visão Geral do Sistema Solar

Capítulo 5

Conclusão

Com este trabalho foi possível aplicar os conhecimentos teóricos relativamente a curvas de Bezier e CatmullRom.

A terceira fase do projeto foi de complexidade muito superior à segunda, tendo o grupo feito um grande esforçoo conjunto para perceber como aplicar os conhecimentos teóricos relativamente a curvas de Bezier e CatmullRom. Esta serviu de elemento de consolidação de conhecimentos adquiridos nas aulas e também de aprendizagem, tendo o grupo dialogado com outros colegas exteriores ao mesmo, e discutido as melhores soluções.

Em suma, é feita uma apreciação positiva ao trabalho realizado, mas apesar dos nossos esforços não conseguimos implementar a geração da patches de bezier na totalidade sem erros pois depois de várias tentativas nao encontramos o problema com a nossa função na parte de gerar os pontos . O grupo conseguiu melhorar a capacidade de comunicação e entreajuda, para uma tarefa que, tendo sido concluída com sucesso.