

UNIVERSIDADE DO MINHO

DEPARTAMENTO DE INFORMÁTICA

Computação Gráfica - Fase 3
Grupo N°2

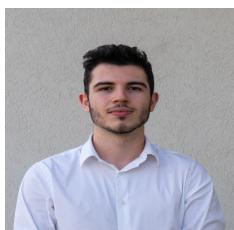
João Correia (A84414)

Marco Pereira (A89556)

Pedro António (A58062)

Rúben Cerqueira (A89593)

28 de junho de 2021



João



Marco



Pedro



Rúben

Capítulo 1

Introdução

O presente relatório visa apresentar a terceira fase do trabalho proposto no âmbito da Unidade Curricular de Computação Gráfica.

Nesta fase foram melhoradas ambas as aplicações. Ao Generator foi adicionada a construção de uma nova primitiva baseada em patches de Bezier; já ao Engine foram adicionadas curvas de Catmull-Rom, de modo a poder animar os vários corpos celestes. Adicionalmente, foram também reorganizados os atributos dentro do XML, por modo a acomodar as novas funcionalidades.

As transformações a aprimorar nesta fase, bem como a informação dada para a sua aplicação, estão seguidamente descritas:

- Translação - recebe o tempo que deve demorar a ocorrer e como filhos recebe os pontos que serão usados de modo a criar a curva de Catmull-Rom a percorrer;
- Rotação - recebe o tempo que uma rotação de 360 graus deve demorar, o lado para o qual deve rodar (clockwise ou counter-clockwise), assim como os eixos a utilizar na rotação);

Relativamente ao desenho das cenas, estas vão ser feitas através de VBOs, tal como já se processava na fase anterior. De modo a melhorar a performance do programa foram removidos alguns smart-pointers desnecessários substituindo-os assim por objetos da classe em questão.

Finalmente, dada a inclusão de tempo, decidimos implementar algumas funcionalidades adicionais, por via a melhorar a experiência do utilizador.

Capítulo 2

Superfícies de Bezier

Um dos objetivos desta fase era a implementação de uma nova primitiva baseada em superfícies de Bezier, com diferentes níveis de tesselação.

O nível de tesselação define os incrementos a ser feitos a \mathbf{u} e \mathbf{v} , sendo o incremento calculado do seguinte modo:

$$u_inc = v_inc = \frac{1}{nível_tesselação}$$

Após este cálculo, temos apenas que ler o ficheiro contendo cada um dos patches de Bezier da figura a criar. Seguidamente são iterados os valores de u e v e calcula-se cada um dos pontos da superfície de Bezier utilizando a fórmula abaixo:

$$p(u,v) = [u^3 \quad u^2 \quad u \quad 1] \begin{bmatrix} -1 & -3 & 3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} P_{00} & P_{10} & P_{20} & P_{30} \\ P_{01} & P_{11} & P_{21} & P_{31} \\ P_{02} & P_{12} & P_{22} & P_{32} \\ P_{03} & P_{13} & P_{23} & P_{33} \end{bmatrix} \begin{bmatrix} -1 & -3 & 3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} v^3 \\ v^2 \\ v \\ 1 \end{bmatrix}$$

Obtém-se assim primitivas como as seguintes:

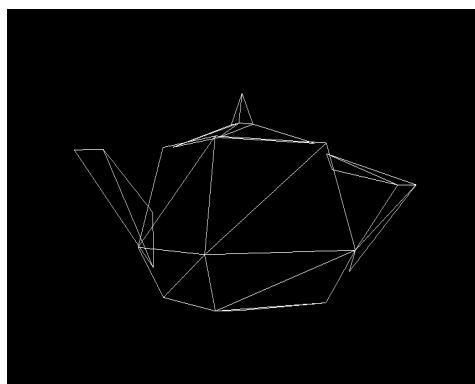


Figura 2.1: Nível de tesselação 1

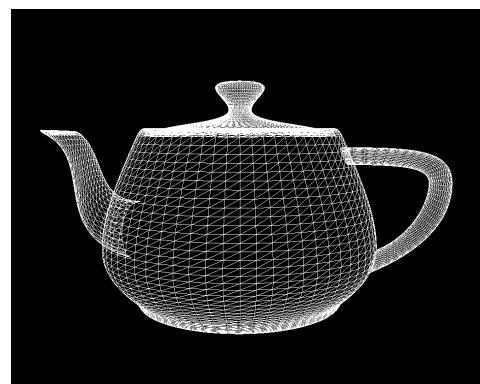


Figura 2.2: Nível de tesselação 16

Capítulo 3

Curvas de Catmull-Rom

Um dos novos requisitos desta fase era a implementação de animações. Desta forma, os corpos celestes presentes no nosso sistema solar podem ser alvo de translações que emulam as suas órbitas em volta do sol, no caso dos planetas, ou volta dos planetas, como é o caso das luas.

Para tal, foi necessária a implementação de uma das temáticas abordadas nas aulas: as curvas de Catmull-Rom. As curvas de Catmull-Rom são formadas através de pelo menos 4 pontos (P_0, P_1, P_2, P_3), os quais manipulamos, em conjunto com uma iteração do valor t , através da equação abaixo apresentada:

$$p(t) = [t^3 \quad t^2 \quad t \quad 1] \begin{bmatrix} -0.5 & 1.5 & -1.5 & 0.5 \\ 1 & -2.5 & 2 & -0.5 \\ -0.5 & 0 & 0.5 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} P_0 \\ P_1 \\ P_2 \\ P_3 \end{bmatrix}$$

Desta forma podemos, através de pontos fornecidos no XML do sistema solar, calcular a órbita de cada corpo celeste. De forma análoga se procede ao cálculo da posição de cada um dos corpos em cada frame, considerando-se o tempo passado em relação ao tempo total que pretendemos para a rotação à volta do Sol/Planeta.

Abaixo temos apresentada uma imagem do Sistema Solar com as várias rotações:

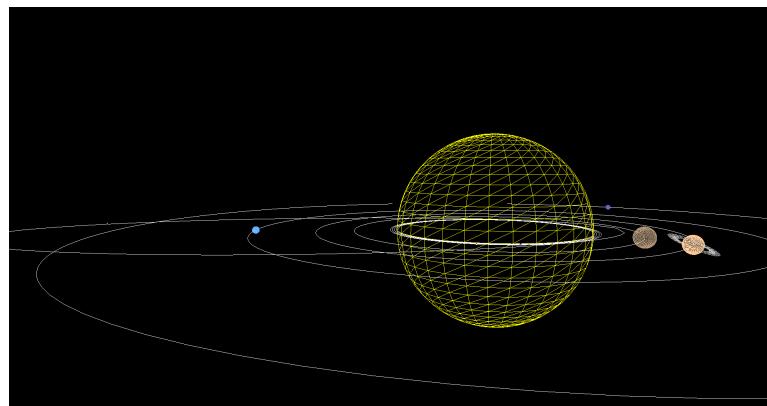


Figura 3.1: Sistema solar

Capítulo 4

Modificações ao ficheiro XML

De forma a poder usufruir das modificações e adições feitas ao funcionamento do Engine, foi modificada a estrutura do ficheiro XML.

Nesta nova fase, o ficheiro XML vai aceitar como translação a informação sobre o tempo a demorar para a mesma e também, aninhados, os pontos a utilizar na criação da curva de Catmull-Rom.

As rotações sofreram também uma modificação, podendo agora receber, além dos eixos onde se pretendem aplicar, o tempo para uma volta de 360 graus, e a informação sobre a direção na qual deve rodar, clockwise ou counter-clockwise.

Além disto, foi adicionado um novo XML com uma versão do sistema solar que contém a cintura de asteróides presente no nosso sistema solar, esta versão sofre de alguns problemas de lentidão em relação à versão sem cintura de asteróides quando tentamos ver as curvas de cada um dos asteróides, cerca de 1000, na próxima versão será feita uma melhoria de performance através de calcular os pontos a desenhar para cada curva apenas uma vez e guardá-los.

O parsing é feito através do uso de tinyxml2, recorrendo ao uso das funções `getFirstChildElement()` e `NextSiblingElement()`. Com estas chamadas conseguimos obter toda a informação necessária do XML.

Capítulo 5

Adições aos controlos de cena

Com a nova adição de animações e com a existência de passagem do tempo, decidimos adicionar e modificar controlos existentes para cada cena.

Assim, foi modificado o seguinte controlo:

- Mudar a apresentação dos eixos xyz - era ' ' passou a ser 'q'

Foram adicionados aos controlos já existentes na fase anterior os seguintes:

- Pausar a simulação - ' '
- Aumentar a velocidade da simulação - '+'
- Diminuir a velocidade da simulação - '-'
- Apresentar as curvas de Catmull-Rom dos corpos celestes - 'c'
- Retroceder/Progredir a simulação (tecla funciona como um switch entre os dois estados) - 'r'

O nosso grupo considera estes controlos suficientes para conseguir ter uma melhor noção de qualquer simulação que venha a ser apresentada no Engine, podendo-se analisar todas as animações com a devida atenção.

Capítulo 6

Conclusão

Ao longo do presente relatório, deu-se a conhecer todas as dificuldades sentidas ao longo da elaboração desta segunda fase do projeto, assim como todas as funcionalidades implementadas.

Apesar do grupo ter cumprido com todas as funcionalidades pedidas e adicionado algumas que não eram requisito, consideramos ainda possível melhorar o projeto com funcionalidades extra.

Algumas das funcionalidades que o grupo pretende implementar futuramente, e que não fazem parte dos requisitos da quarta fase, são: implementar a utilização de frustum culling, assim como adicionar pequenas janelas à simulação para melhorar a percepção das definições da simulação por parte do utilizador.

Consideramos que a entrega desta terceira fase cumpre com todos os requisitos presentes no enunciado, e será seguramente uma base sólida para iniciar o trabalho da quarta e última fase.