



Universidade do Minho

Escola de Engenharia

Licenciatura em Engenharia informática

Mestrado Integrado em Engenharia Informática

Unidade Curricular de Computação Gráfica

Ano Letivo de 2023/2024

Computação Gráfica – Checkpoint 1

João Pedro Mota Baptista a100705

João Pedro da Rocha Rodrigues a100896

João Manuel Machado Lopes a100594

Tiago Nuno Magalhães Teixeira a100665

Computação Gráfica

João Pedro Mota Baptista a100705

João Pedro da Rocha Rodrigues a100896

João Manuel Machado Lopes a100594

Tiago Nuno Magalhães Teixeira a100665

8 março 2024

Resumo

No âmbito da disciplina de Computação Gráfica, fomos desafiados a desenvolver uma mini cena sobre uma engine 3D, bem como provar exemplos de utilização para provar o seu potencial.

Em conformidade com essa solicitação, criamos dois módulos principais, um “Generator”, e um “Engine”.

Finalmente, aplicamos, na prática, todas as conceptualizações e ideias previamente definidas na aplicação final.

Área de Aplicação: Desenvolvimento de Software Gráfico

Palavras-Chave: Computação Gráfica

Índice

ENGINE.....	1
GENERATOR.....	2
TESTES.....	3
CONCLUSÃO	6

Engine

O Engine é responsável por desenhar, utilizando a biblioteca GLUT, os modelos disponibilizados, bem como o ficheiro XML que contém as definições da câmara, e o nome dos modelos a desenhar.

O código do mesmo é inspirado no código das aulas pois, o objetivo é muito similar (com a exceção do parsing do XML e dos ficheiros .3d)

Começa-se por definir algumas variáveis globais que são responsáveis pela rotação da câmara, quando são pressionadas certas teclas.

Logo de seguida, é definida a função `drawModel` que, dando como argumento um ficheiro (.3d), lê o mesmo e, utilizando as funções do GLUT, desenha cada linha (cada uma contendo 3 pontos que formam 1 triângulo).

A função principal (`main`) é responsável pelo parse do ficheiro XML que é dado como argumento do programa. A mesma lê cada linha e, consoante a tag que é apresentada, guarda o valor numa variável. Posteriormente, esses valores são utilizados pela função `renderScene`, com ajuda da `gluLookAt()`, bem como a função `drawModel` mencionada acima.

O restante código é o esqueleto base de um programa desenvolvido em GLUT, igual ao utilizado nas aulas práticas. São definidas funções como `processKeys()`, `processSpecialKeys()`, `renderScene()` (já mencionada), e `changeSize()`.

Generator

A parte de criação dos ficheiros é aplicada através do Generator. Este tem como parte inicial uma função “main” que recebe os argumentos para a construção das figuras. Vai verificar, através dos argumentos fornecidos, qual é a figura a ser desenhada. Após esta verificação, dependendo do resultado da mesma, os argumentos para o desenho vão ser redirecionados para a função particular da figura em questão.

Desta forma, cada figura terá o seu tipo de construção dependendo do número de argumentos e do resultado pretendido.

Em geral, as funções das figuras vão ser bastante semelhantes.

Primeiramente, vão ser calculados os vértices necessários para a criação da figura em questão, sendo estes armazenados.

Posteriormente vão ser calculados os triângulos a ser formados com estes vértices, para poder ser realizado o desenho. Estes triângulos vão ser guardados num ficheiro com o nome “figura.3d”. Vão ser guardados 3 pontos por linha, cada um com as suas coordenadas.

A única exceção a isto vai ser a formação da “box”. O princípio é o mesmo, são calculados os vértices e posteriormente os triângulos. No entanto, cada face desta figura vai ser definida de forma individual, ou seja, cada plano da “box” é feito individualmente, visto que cada um terá de ser tratado de forma a possibilitar o desenho final da figura.

Estes ficheiros “.3d” serão guardados num diretório “Output” para poderem finalmente ser requisitados pelo Engine aquando do desenho gráfico desta figura.

Testes

De forma a comprovar o correto funcionamento dos nossos programas, pusemos à prova os mesmos através da realização dos testes disponibilizados. Apresentamos de seguida os resultados do parsing de cada xml seguido da representação da figura respetiva.

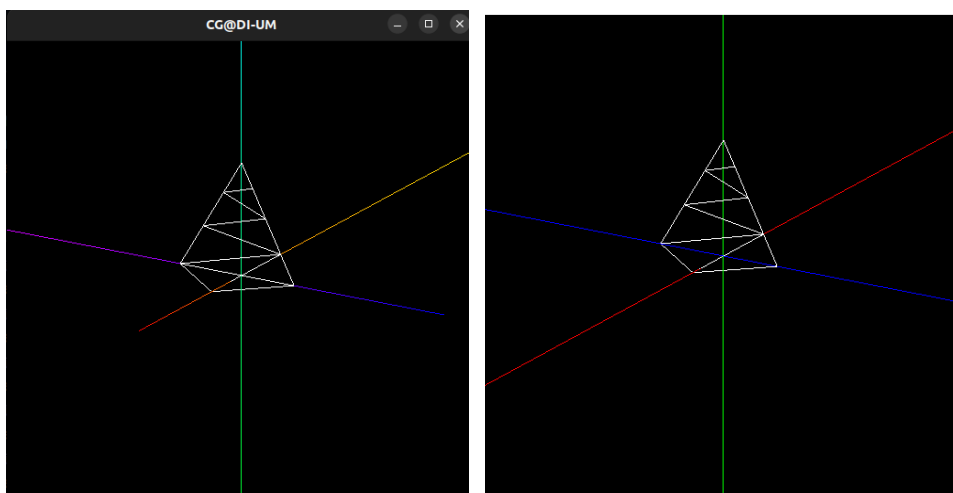


Fig. 1 – Resultado obtido (esquerda) e resultado esperado (direita) do teste 1_1

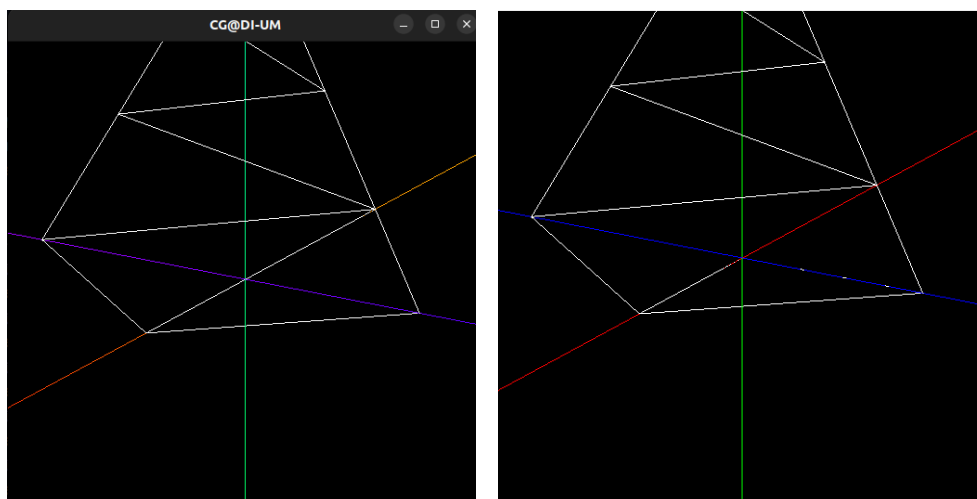


Fig. 2 – Resultado obtido (esquerda) e resultado esperado (direita) do teste 1_2

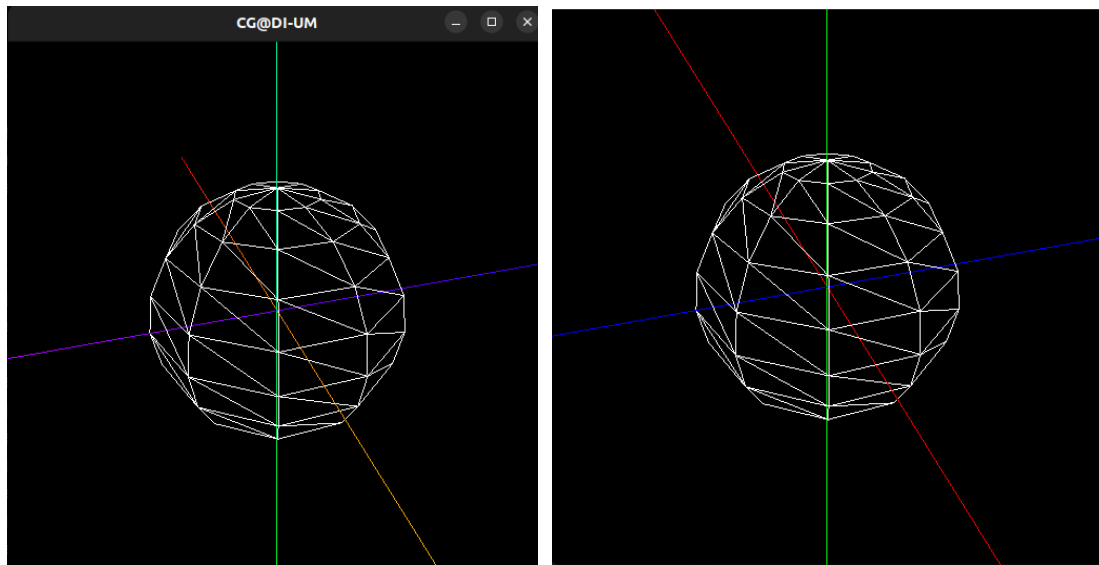


Fig. 3 – Resultado obtido (esquerda) e resultado esperado (direita) do teste 1_3

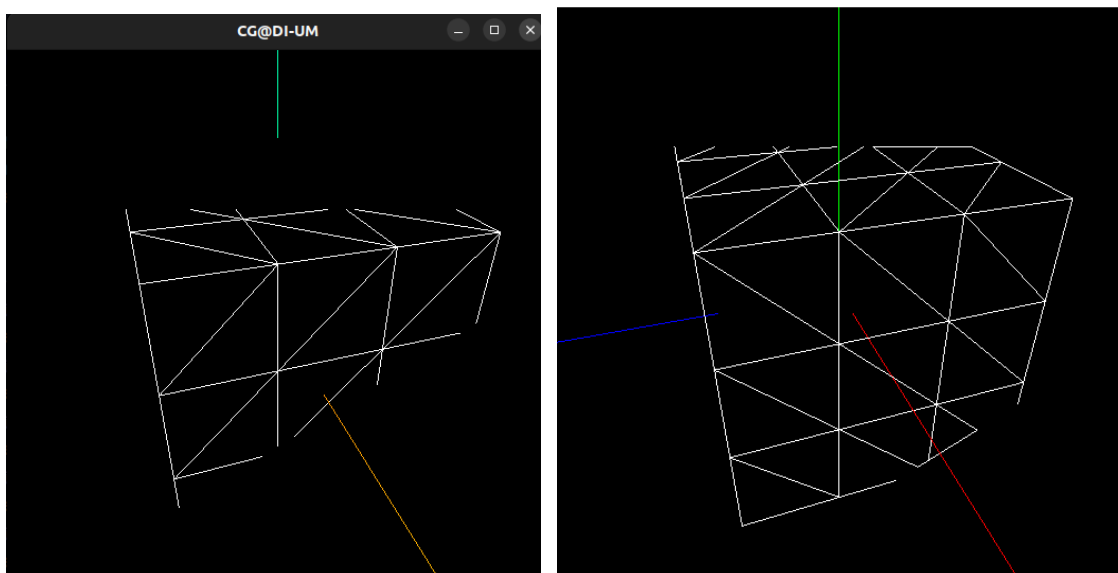


Fig. 4 – Resultado obtido (esquerda) e resultado esperado (direita) do teste 1_4

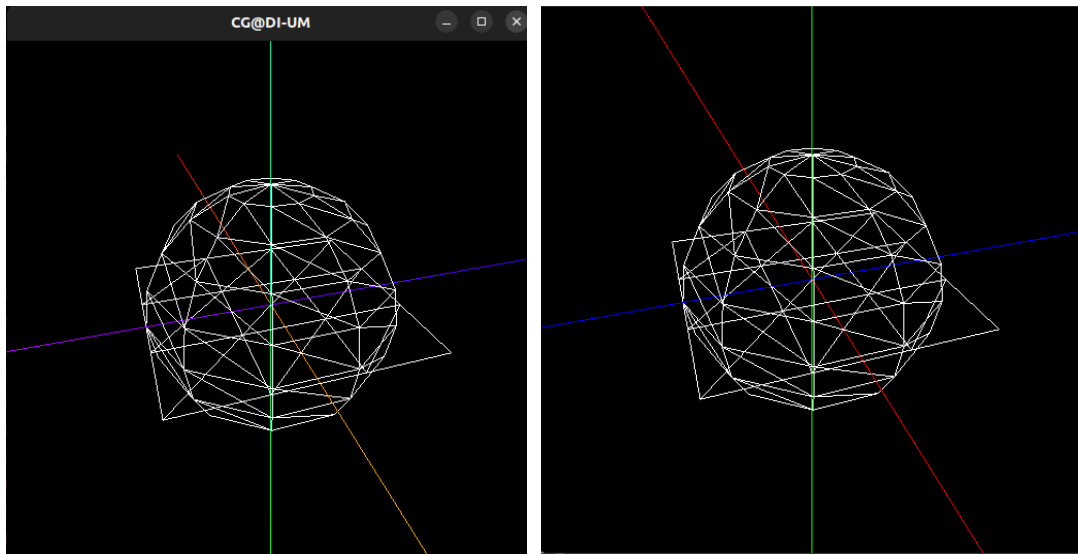


Fig. 5 – Resultado obtido (esquerda) e resultado esperado (direita) do teste 1_5

Assim, como podemos observar pelas figuras e ao comparar os nossos resultados com os esperados, concluímos que os outputs e o funcionamento tanto do generator como da engine estão corretos.

Conclusão

Nesta primeira fase do projeto, aprofundamos o nosso conhecimento sobre os básicos da computação gráfica, nomeadamente sobre criação de modelos 3D de diferentes figuras, com a ajuda biblioteca GLUT.

Acreditamos que atingimos o pretendido, tendo o nosso programa ultrapassado satisfatoriamente todos os testes fornecidos.