

Departamento de Engenharia Informática
2015/2016



UNIVERSIDADE DE COIMBRA

Relatório de Projecto
Computação Gráfica

Poker Room

Fernando Miguel Cardoso Rocha 2012131752
Rui Pedro das Neves Dias 2012168627

- PROJECTO

O projecto de Computação Gráfica do ano corrente tem como tema principal “Desporto”. Assim, decidimos em fazer um modelo de uma sala de jogo de poker.

- OBJECTIVO

Este projecto tem como objectivo principal o controlo de uma câmara/observador sobre o modelo criado, visualizando o modelo tanto como uma câmara “exterior” como do ponto de vista de cada jogador à volta de uma mesa central.

- CONTROLOS

As **setas de cima, baixo, esquerda e direita** são utilizadas para o movimento da câmara/observador quando possível. Para além disso, os botões de 1 a 4 servem para visualizarmos a cena do ponto de vista dos jogadores.

A tecla ‘**L**’ liga a luz pontual situada no topo da cena.

A tecla ‘**F**’ liga os focos de luz em cada canto da sala.

A tecla ‘**P**’ mete o utilizador em modo de visualização 360°.

A tecla ‘**Esc**’ sai do programa.

- CONCEITOS UTILIZADOS

Os conceitos utilizados foram os mesmos lecionados ao longo do semestre.

- **Modelos e transformações**: todos os “objectos” criados na nossa sala/modelo foram feitos à custa de transformações geométricas simples e modelos pertencentes à biblioteca do GLUT, assim como o movimento do observador sobre a cena.
- **Texturas**: as texturas foram utilizadas para a mesa central, chão, tecto, jogadores e parede onde se encontra o espelho. Foram também aplicadas texturas, recorrendo ao *Cull_Face* às cartas de jogo, permitindo assim ter 2 texturas diferentes para o mesmo polígono. Para além das texturas foram também utilizados materiais fornecidos pelos Professores da cadeira.
- **Iluminação**: na iluminação foram implementados 3 tipos. Luz ambiente, uma luz pontual e quatro focos. Sendo que o ponto de iluminação local, como os focos dão para alternar entre ligados e desligados, incidindo sobre os objectos da cena, permitindo ao utilizador uma boa visualização da luminosidade.
- **Reflexões/sombras**: de forma a implementarmos as reflexões, decidimos inserir um plano, com o objectivo de ser um espelho, na sala. Esse espelho reflecte todos os objectos da cena, excepto as cartas, para os jogadores não fazerem batota.

- **Transparências:** os conceitos de transparência foram usados em 2 momentos: Para o desenho da caixa com a chaleira na mesa central e para a implementação da reflexão previamente referida.

- PONTOS FORTES E FRACOS

Um dos pontos fortes do projecto é a sua simplicidade e fácil compreensão. Pois sendo um projecto simples, consegue-se perceber bem as transformações aplicadas, conseguir identificar os pontos luminosos, saber diferenciar texturas de color material, bem como a interação do utilizador com o mesmo se torna mais fácil.

Quanto aos pontos fracos, podemos assumir a não implementação de conceitos mais complexos, não abordados nos conceitos leccionados, tais como motores de particulas, *bump*, *shadow mapping* e *shaders*.

- REFERÊNCIAS

Todo o código utilizado neste projecto foi escrito por nós, salvo algumas excepções em que foram aproveitados excertos dos Trabalhos Práticos realizados e fornecidos pelos Professores ao longo do semestre.

- COMPILAÇÃO E EXECUÇÃO DO PROGRAMA

Para a correcta compilação e execução do programa, este deve ter todos os ficheiros dentro da mesma directoria.

Fazem parte dessa directoria os seguintes ficheiros:

- **main.cpp:** Ficheiro que contém todo o código necessário à implementação do programa.
- **RgbImage.cpp:** Estes ficheiros contém as funções para carregar as imagens necessárias para as texturas.

E as seguintes pastas:

- **cards:** Pasta que contém as texturas das cartas.
- **room:** Pasta que contém as texturas da sala.
- **tuxedo:** Pasta que contém as texturas dos players.

Deve então proceder à compilação do programa, executando na linha de comandos o seguinte comando:

- `g++ main.cpp RgbImage.cpp -IGL -lglut -lGLU -o main` - Para o caso de usar **Linux**

- `g++ RgbImage.cpp main.cpp -framework OpenGL -framework GLUT -o main` - Para o caso de usar **OS X**.

Feito isto, basta executar o ficheiro executável.