

Tópicos Especiais em Computação Gráfica

Prof. Daniel Ribeiro Trindade

Trabalho 1

Objetivos:

- carregamento de malhas de triangulos;
- representação de malhas de triangulos;
- iluminação padrão OpenGL;

Descrição:

O trabalho consiste no carregamento de um arquivo contendo uma malha de triangulos e a renderização dessa malha. A malha deverá ser iluminada usando o esquema básico de iluminação do OpenGL. O projeto exemplo glut do codeblocks contém um exemplo de iluminação básica.

O arquivo de malha é no formato .msh. O formato .msh é um formato simplificado para armazenamento de malha de triângulos. Na primeira linha, aparecem o número de vértices e o número de triângulos (nv nt); em seguida vem os atributos dos vértices: id, coordenadas e normal (id x y z nx ny nz); em seguida a incidência dos triângulos (id v1 v2 v3).

A malha de triângulos deve ser desenhada de duas formas. A primeira usando as funções:

- **glBegin(GL_TRIANGLES)**
- **glEnd()**

A segunda usando vetores de vértices, normais e índices através de funções OpenGL do tipo:

- **glVertexPointer**
- **glNormalPointer**
- **glDrawElements**

Somente uma das formas de renderização descritas acima deve ser usada em determinado momento. O modo de renderização **deve** ser trocado se o usuário apertar a tecla 'r'.

Para verificar o desempenho da aplicação, o aluno **deve** calcular o FPS (Frames por segundo). O FPS representa a quantidade de frames/quadros são desenhados por segundo.

O valor do FPS **deve** ser escrito no interior da janela de desenho e **não** como saída de terminal. Para desenhar texto na janela de desenho, use a função abaixo:

```
// x → posição x dada em pixels na qual o texto será escrito
// y → posição y dada em pixels na qual o texto será escrito
// winwidth → largura da tela em pixels
// winheight → altura da tela em pixels
// str → texto que vai ser desenhado na tela
void printtext(int x, int y, int winwidth, int winheight, char *str) {
    //(x,y) is from the bottom left of the window
    glMatrixMode(GL_PROJECTION);
    glPushMatrix();
    glLoadIdentity();
    glOrtho(0, winwidth, 0, winheight, -1.0f, 1.0f);
    glMatrixMode(GL_MODELVIEW);
    glPushMatrix();
    glLoadIdentity();
    glPushAttrib(GL_DEPTH_TEST);
    glDisable(GL_DEPTH_TEST);
```

```
glRasterPos2i(x,y);
int len = (int) strlen(str);
for (int i=0; i<len; i++)
{
    glutBitmapCharacter(GLUT_
}
glPopAttrib();
glMatrixMode(GL_PROJECTION);
glPopMatrix();
glMatrixMode(GL_MODELVIEW);
glPopMatrix();
}
```

A malha de triângulos **deve** ser posicionada de modo que ela seja vista em sua totalidade.

Os parâmetros de iluminação podem ser estáticos a critério do aluno. O mesmo vale para a cor.

Por último, o aluno **deve** aplicar uma rotação constante ao modelo em torno do eixo **Y**. Isso dará movimento ao objeto.

Prazo de entrega:

O aluno terá duas semanas para fazer o trabalho. Esse prazo começa a contar a partir da data de publicação desse documento.