

Aula Prática Laboratorial n.º 8

Sumário

Tutorial Nate Robins “*transformation*”. Programa C/OpenGL “cubo” baseado no template “*Aula8_template.cpp*”

Tutorial Nate Robins “*transformation*”

Execute o tutorial “*transformation*” de Nate Robins e veja o projecto “*cubo*” a funcionar.

Projecto “cubo”

1. Crie um projecto do tipo consola com base no “*Aula8_template.cpp*” que foi disponibilizado;
2. Crie a função *cubo()*, que cria um cubo de lado 1 com cores diferentes nas várias faces e centrado na origem. Baseie-se na função existente *cubo()* que desenha um polígono de 4 lados;
3. Alterar os valores de *theta[0]*, *theta[1]* e *theta[2]* usados na função *Draw()* na instrução *glRotatef(...)*;
4. Inserir código na função *Timer()* para rodar o cubo em torno dos 3 eixos. Use a função *Mouse()* para controlar o eixo a ser rodado (x,y,z) dependendo da tecla premida. Se for premida a tecla para qual o eixo é o actual, para de rodar;
5. Insira o código para ligar o buffer de profundidade para esconder as faces de trás do cubo:
 - *main()* – acrescentar `GLUT_DEPTH` na instrução `glutInitDisplayMode`
 - *Init()* – descomentar a instrução `glEnable(GL_DEPTH_TEST)`;
 - *Draw()* – acrescentar `GL_DEPTH_BUFFER_BIT` na instrução `glClear`
6. Inserir uma tecla que permita aumentar e diminuir o tamanho do cubo usando a instrução *glScalef(...)*;
7. Use a função *glPolygonMode(...)* para alterar representação da parte de trás dos polígonos para linhas e assim ver se todos as faces estão bem orientadas;
8. Criar uma função *eixos()* para desenhar a parte positiva dos 3 eixos em 3 cores diferentes centrada na origem;
9. Usar a função *eixos()* para desenhar uns eixos pequenos a rodar no canto do ecrã, use a instrução *glTranslatef(...)*;

10. Colocar 3 cubos pequenos em cada um dos eixos do grande;
11. Fazer os 3 cubos pequenos aproximarem-se do cubo grande até o tocarem e depois afastarem-se;
12. Rodar os cubos pequenos em torno do eixo onde estão ao aproximarem-se do grande e evitar que eles rodem quando se afastem;
13. Criar uma função *cubo1()* que crie o cubo usando somente uma face e várias translações e rotações.

Observações

glFrontFace(modos)

Instrução para alterar o lado do polígono que é tratado como frente. *GL_CCW* é a opção por omissão.

modo – *GL_CCW* ou *GL_CW*

glPolygonMode(face, modo)

Instrução para alterar a representação das faces de um polígono

face – *GL_FRONT*, *GL_BACK* ou *GL_FRONT_AND_BACK*)

modo – *GL_FILL*, *GL_LINE* ou *GL_POINT*

glLoadIdentity()

Instrução para carregar a matriz identidade, limpando todas as transformações realizadas

glPushMatrix() *glPopMatrix()*

Instruções para guardar e repor o estado da matriz de transformação usando uma *stack*

glTranslatef(dx, dy, dz)

Instrução para fazer uma translação de *dx* unidades em *x*, *dy* unidades em *y* e *dz* unidades em *z*.

glRotatef(angulo, x, y, z)

Instrução para fazer uma rotação em torno da origem de *angulo* graus em torno do eixo definido pelo vector *x, y, z*.

glScalef(escala_x, escala_y, escala_z)

Instrução para fazer alterar a escala. Os valores *escala_x*, *escala_y*, *escala_z* são um factor de multiplicação.