

Engenharia da Computação
Aluno: André Oliveira de Sousa – 11325684
Professora: Liliane dos Santos Machado

Introdução à Computação Gráfica
Atividade 7

Catavento

Fiz dois cataventos, um usando apenas rotação e utilizando triângulos e o outro usando rotação, translação e utilizando quadrados. Abaixo segue o código em c deles:

1º Catavento

```
#include<stdio.h>
#include<GL/glut.h> // biblioteca para usar o OpenGL
#include<stdlib.h>

void bastao() {
    glLineWidth(5); // Largura da linha
    glColor3f(0, 0, 0); // Cor da linha
    glBegin(GL_LINES); // Função para desenho de linha (bastão)
        glVertex2d(0, 0); // O bastão inicia na origem
        glVertex2d(0,-1); // e se estende na vertical
    glEnd();
}

void helices() {
    glMatrixMode(GL_MODELVIEW); // selecionando a matriz onde vamos aplicar as transformações
    glLoadIdentity();
    float angulo = 30; // ângulo.

    glColor3f(0, 0, 1.5); // Cor das hélices
    for(int i = 0; i < 4; i++) {
```

```

    glRotatef(angulo, 0.0, 0.0, 1.0); // Gira o objeto em torno do eixo Z
    // O giro é em graus, no sentido anti-horário.

    // Desenhando hélices.
    glBegin(GL_TRIANGLES);
        glVertex2d(0.5, 0.5);
        glVertex2f(0.5, 0.0);
        glVertex2f(0.0, 0.0);
    glEnd();

    angulo = 90;

}
glFlush(); // Imprime na tela o que estava armazenado no buffer.

}

void display(){
    glMatrixMode(GL_PROJECTION);
    glLoadIdentity(); // Resetar a posição e orientação do sistema de coordenadas
    glClearColor(1, 1, 1, 1); // Limpa o fundo de janela inserindo a cor especificada
    glOrtho(-3.0, 3.0, -3.0, 3.0, -1.0, 1.0);

    glClear(GL_COLOR_BUFFER_BIT); // Limpa toda a janela para a cor do comando glClearColor.
    bastao();
    helices();
}

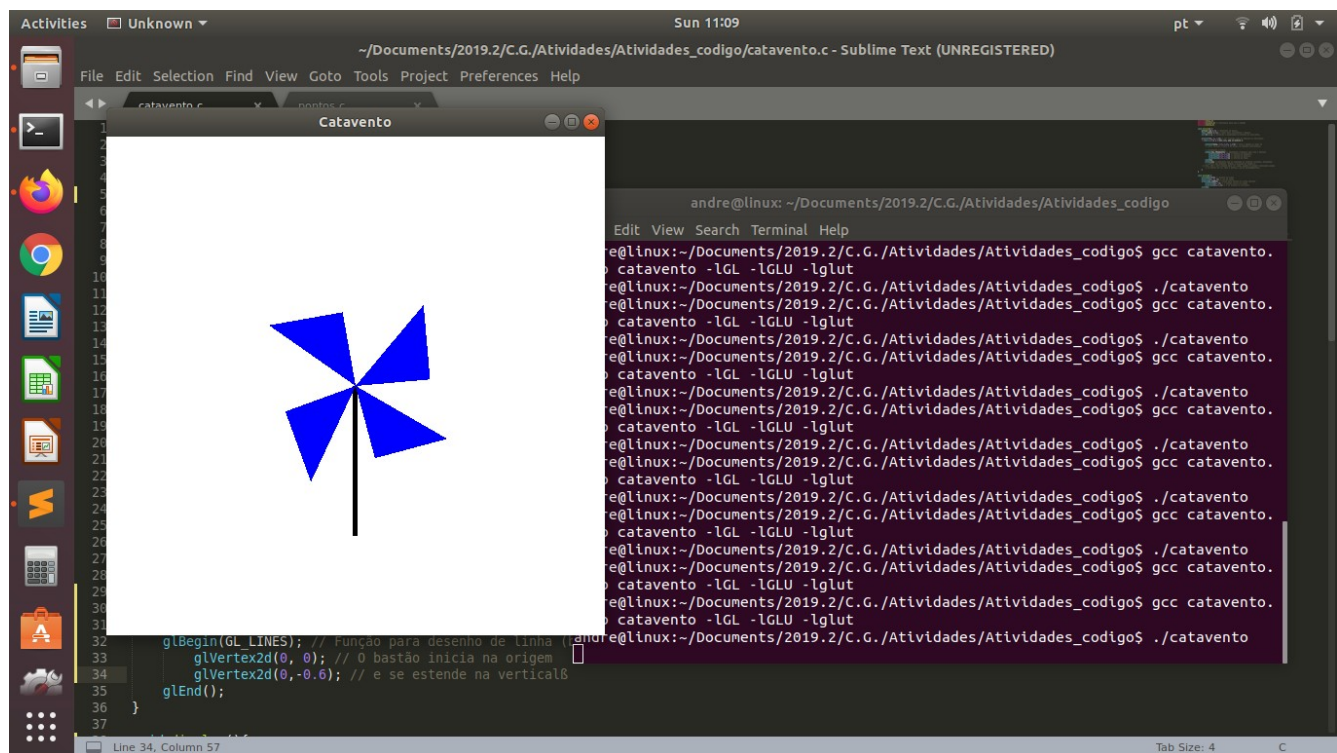
int main(int argc, char** argv){

    glutInit(&argc, argv); // inicia a biblioteca glut
    glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB);
    glutInitWindowSize(512,512); // Tamanho da janela que abrirá
    glutInitWindowPosition(800,100); // Posição em que a janela que abrirá irá aparecer na tela do PC
    glutCreateWindow("Catavento"); // Título da janela
    glutDisplayFunc(display); // chamada da função que realmente desenha. Necessário usar
    glutDisplayFunc
        // para poder usar comandos de redimensionamento de janela, etc.

    glutMainLoop();
    return 0;
}

```

Imagem gerada:



2º Catavento

```
#include<stdio.h>
#include<GL/glut.h> // biblioteca para usar o OpenGL
#include<stdlib.h>

void centro() {

    glColor3f(0, 0, 1.5); // Cor das hélices

    // Desenhando centro.
    glBegin(GL_QUADS);
        glVertex2d(0.25, 0.25);
        glVertex2f(0.25, -0.25);
        glVertex2f(-0.25, -0.25);
        glVertex2f(-0.25, 0.25);
    glEnd();

    glFlush(); // Imprime na tela o que estava armazenado no buffer.
}
```

```

void bastao() {
    glLineWidth(5); // Largura da linha
    glColor3f(0, 0, 0); // Cor da linha
    glBegin(GL_LINES); // Função para desenho de linha (bastão)
        glVertex2d(0, 0); // O bastão inicia na origem
        glVertex2d(0,-1); // e se estende na vertical
    glEnd();
}

void helices() {
    glMatrixMode(GL_MODELVIEW); // selecionando a matriz onde vamos aplicar as transformações
    glLoadIdentity();
    float angulo = 0; // ângulo.

    glColor3f(0, 0, 1.5); // Cor das hélices
    for(int i = 0; i < 4; i++) {

        glRotatef(angulo, 0.0, 0.0, 1.0); // Gira o objeto em torno do eixo Z
        // O giro é em graus, no sentido anti-horário.

        glTranslatef(0.25, 0, 0);

        // Desenhando hélices.
        glBegin(GL_QUADS);
            glVertex2d(0.5, 0.5);
            glVertex2f(0.5, 0.0);
            glVertex2f(0.0, 0.0);
            glVertex2f(0.0, 0.5);
        glEnd();

        //glLoadIdentity();

        angulo = 90;
    }
    glFlush(); // Imprime na tela o que estava armazenado no buffer.

}

void display(){
    glMatrixMode(GL_PROJECTION);
    glLoadIdentity(); // Resetar a posição e orientação do sistema de coordenadas
    glClearColor(1, 1, 1, 1); // Limpa o fundo de janela inserindo a cor especificada
    glOrtho(-3.0, 3.0, -3.0, 3.0, -1.0, 1.0);

    glClear(GL_COLOR_BUFFER_BIT); // Limpa toda a janela para a cor do comando glClearColor.
    centro(); // quadrado central do catavento
}

```

```
glutMainLoop();
return 0;
}
```

[illegible]