

# Relatório de Computação Gráfica

## **Trabalho 3**

Anderson Caio Santos Silva, 7972630

Luana Okino Sawada, 4589724

Robson Marques Pessoa, 8632563

**Enunciado do trabalho:** Confeccionar um ambiente 3D com a OpenGL no qual cada um dos octetos contenha um objeto (por ser um cubo). E o usuário possa controlar, via terceira pessoa neste ambiente, um objeto carregado de algum modelo 3D. A movimentação deve ser nos três eixos (como se o objeto controlado voasse).

**Plataforma utilizada:** O referido trabalho foi confeccionado no sistema operacional Linux. Não houve a utilização de IDE. Apenas foi utilizado um editor de texto e o compilador gcc pelo terminal, o qual faz uso da linguagem C++ e da biblioteca OpenGL.

**Tecnologias externas:** Para a confecção do referido trabalho foram utilizados os materiais de estudos obtidos no site da disciplina e a bibutil, biblioteca de rotinas auxiliares que acompanha o livro “OpenGL – Uma abordagem prática e objetiva”.

**Principais dificuldades:** O passo mais desafiador deste trabalho foi confeccionar corretamente os octetos.

**Divisão de tarefas:** Luana foi responsável por editar o código disponibilizado e adequar ao que precisávamos. Anderson revisou o código, o funcionamento e escreveu o relatório. Robson vai ficar responsável por contribuição majoritária do próximo trabalho.

**Tempo:** Para a realização deste trabalho foi preciso trabalhar por 8 horas.

**Trecho de código que o grupo julga mais importante:** O grupo acredita que o código a seguir foi a peça principal no referido trabalho.

```
void display(void) {
    glClearColor(0.0, 0.0, 0.0, 1.0); //clear the screen to black
    glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);
    //clear the color buffer and the depth buffer
    enable();

    glLoadIdentity();

    glTranslatef(0.0f, 0.0f, -cRadius);
    glRotatef(xrot, 1.0, 0.0, 0.0);
    glColor3f(1.0f, 0.0f, 0.0f);
    //glutSolidCube(2); //Our character to follow

    glPushMatrix();
    //glLoadIdentity();
    DesenhaObjeto(objeto);
    glPopMatrix();

    glRotatef(yrot, 0.0, 1.0, 0.0); //rotate our camera on the y - axis(up and down)
    glTranslated(-xpos, -ypos, -zpos); //translate the screen to the position of our camera

    glColor3f(1.0f, 0.498f, 0.0f);
    cube(); //call the cube drawing function

    glColor3f(1.0f, 1.0f, 0.0f);
    sphere(); //call the sphere drawing function

    glColor3f(0.0f, 1.0f, 0.0f);
    cone(); //call the cone drawing function

    glColor3f(0.0f, 0.0f, 1.0f);
    torus(); //call the torus drawing function

    glColor3f(0.4f, 0.11f, 0.49f);
    dodecahedron(); //call the dodecahedron drawing function

    glColor3f(0.85f, 0.15f, 0.2f);
    octahedron(); //call the octahedron drawing function

    glColor3f(1.0f, 1.0f, 1.0f);
    tetrahedron(); //call the tetrahedron drawing function

    glColor3f(0.0f, 1.0f, 1.0f);
    icosahedron(); //call the icosahedron drawing function

    glutSwapBuffers(); //swap the buffers
    angle++; //increase the angle
}
```

## **Demonstração de funcionalidades:**

O conjunto de teclas WASD é utilizado para controlar a movimentação do objeto (monkey), enquanto que as teclas Q e Z controlam a câmera (que também pode ser controlada pelo mouse). A tecla ESC é pode ser utilizada para encerrar o programa.