CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA DE MINAS GERAIS
CAMPUS VII - UNIDADE TIMÓTEO
Curso: Engenharia de Computação
Disciplina: Computação Gráfica
Professor: Odilon Corrêa

OPENGL – PRÁTICA 02

Objetivo

Aprender como funcionam e como são abordadas em OpenGL algumas das etapas do processo de visualização para geração de uma imagem a partir de um modelo ou conjunto de modelos.

Visualização

Podemos utilizar diferentes sistemas de coordenadas para descrever os objetos modelados em um sistema 2D. Para entender como funciona o processo de visualização bidimensional em OpenGL é importante conhecer o conceito de "universo", que pode ser definido como a região do plano (ou do espaço) utilizada em uma aplicação. Como a descrição geométrica de um modelo normalmente envolve coordenadas geométricas, é preciso adotar um sistema de referência que irá definir uma origem em relação à qual todos os posicionamentos do universo são descritos.

Sistema de Referência do Universo (SRU)

Plano cartesiano com dois eixos (*x* e *y*) perpendiculares entre si e que se cruzam na origem, sendo o eixo *x* na horizontal e orientado para direita, e o eixo *y* vertical e orientado para cima (Figura 1a). Todos os modelos e comandos OpenGL são definidos em relação a este sistema de referência.

Sistema de Referência da Tela ou Dispositivo (SRT ou SRD)

Adotado pelo monitor do computador. Diferencia-se do SRU devido a origem ficar localizada no canto superior esquerdo do monitor (Figura 1b). Portanto, para possibilitar a correta visualização na tela dos modelos definidos no SRU, é necessário fazer uma conversão ou mapeamento, conforme exemplifica a Figura 1.

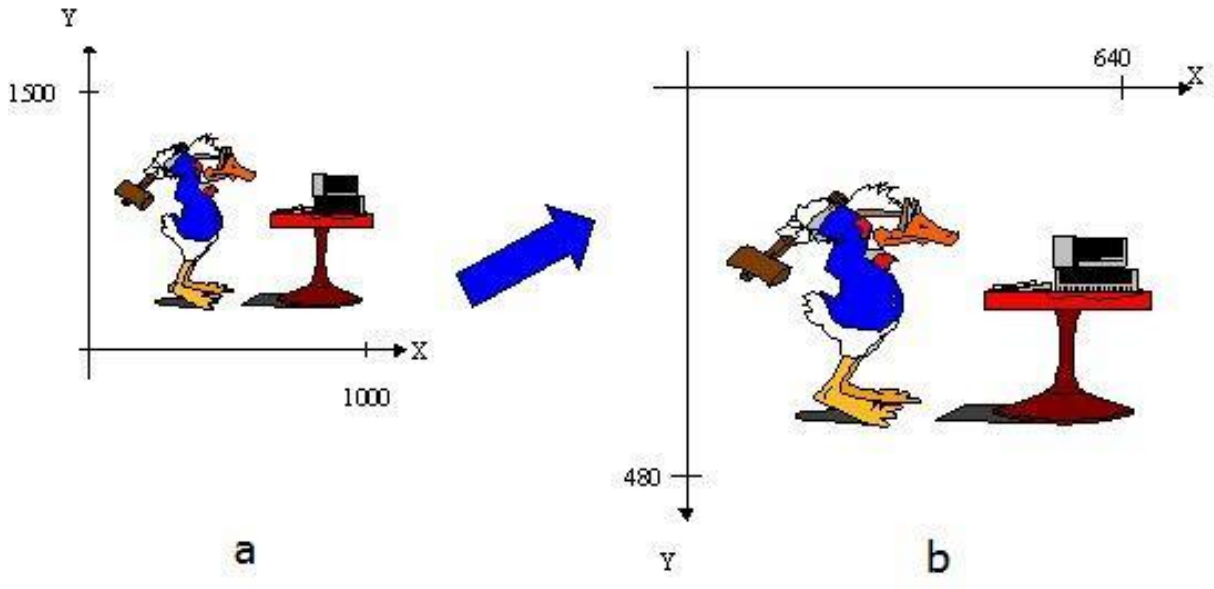

Figura 1 – Mapeamento entre SRU (a) e SRT (b)

Como o universo por definição é infinito, é necessária uma forma de se especificar qual a porção deste que desejamos mapear para a tela. Esta área do universo, que delimita a região de interesse do usuário em um dado instante, chamamos de *window*, janela de seleção ou recorte. Uma *window* é delimitada pelas coordenadas de seus cantos (esquerda, direita, inferior e superior) sempre em relação à SRU.

Recorte (*window*)

A etapa de recorte permite que se defina qual a região do desenho se deseja exibir (Figura 2) e pode ser realizado em OpenGL através da seguinte função:

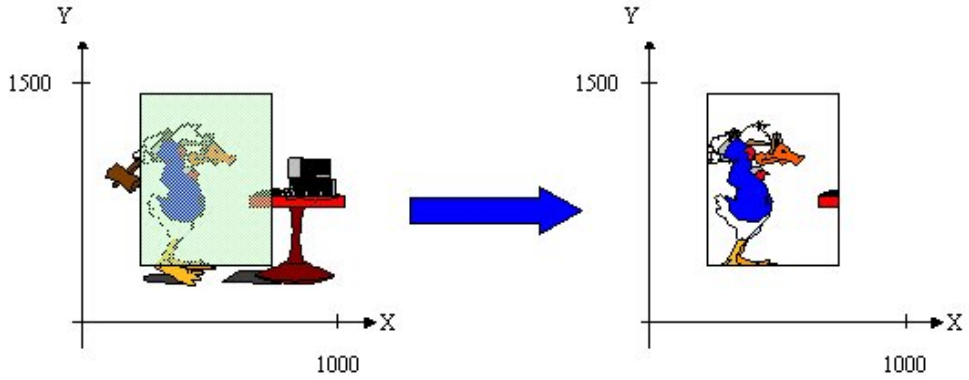

Figura 2 – Exemplo de recorte (*window*)

```
void gluOrtho2D(GLdouble esq, GLdouble dir, GLdouble inf, GLdouble sup)
```

Seus parâmetros correspondem, respectivamente, às coordenadas de cada borda da janela de seleção, isto é, *x* mínimo (borda esquerda), *x* máximo (borda direita), *y* mínimo (borda inferior) e *y* máximo (borda superior).

Mapeamento

A etapa de mapeamento permite que se exiba em uma tela, ou em outro dispositivo, um conjunto de objetos com coordenadas totalmente diferentes daquelas nas quais a tela está definida. Na Figura 1 pode-se observar um exemplo de um desenho criado em coordenadas totalmente distintas da tela sendo mapeado para a mesma.

O mapeamento deve especificar em que parte do monitor deseja-se exibir o conteúdo da *window*. Chamamos esta região de *viewport* ou janela de exibição. Em OpenGL, uma *viewport* é normalmente delimitada pelo tamanho da janela GLUT (ou Freeglut), definida sempre em valores que dizem respeito ao SRD. A seguinte função OpenGL realiza o procedimento de mapeamento:

```
void glViewport (GLint x, GLint y, GLsizei largura, GLsizei altura)
```

Os dois primeiros parâmetros especificam o canto inferior esquerdo da *viewport*, enquanto os dois últimos definem, respectivamente, sua largura e altura.

Exercício

Faça o download do arquivo (OpenGL_Prática02_Projeto.zip) da aula de hoje e utilize o projeto para compreender os conceitos apresentados até o momento.

Figura 3 – Saída do projeto

Faça as modificações necessárias para exibir a saída abaixo:

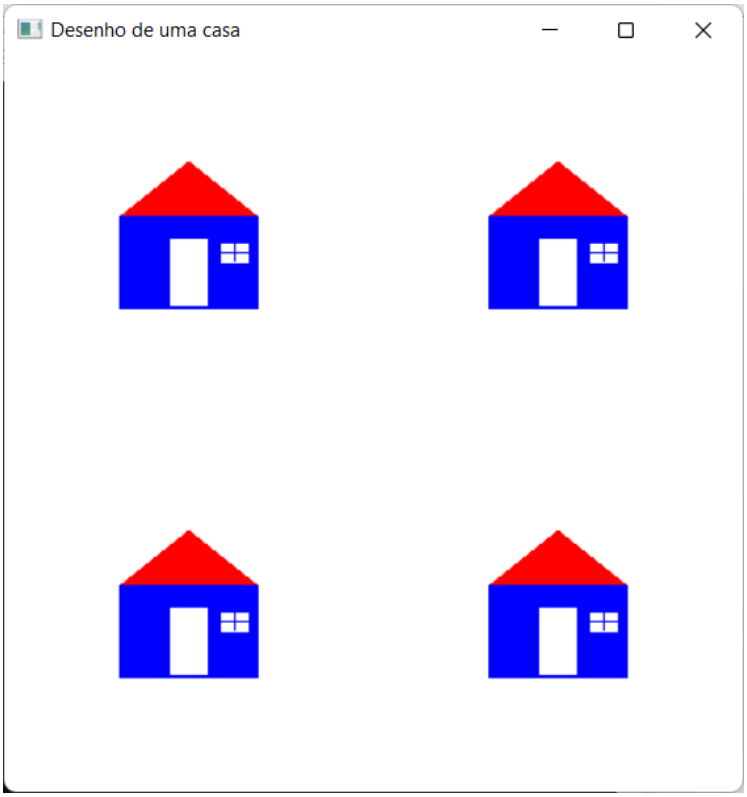

Figura 4 – Saída do exercício

Referências

Todas as informações descritas neste roteiro foram retiradas do livro OpenGL: Uma abordagem prática e objetiva, presente no plano didático da disciplina, e do site da Profa. Isabel Harb Manssour (https://www.inf.pucrs.br/~manssour/OpenGL/Tutorial.html/) e Prof. Márcio Sarroglia Pinho (http://www.inf.pucrs.br/~pinho/CG/)