

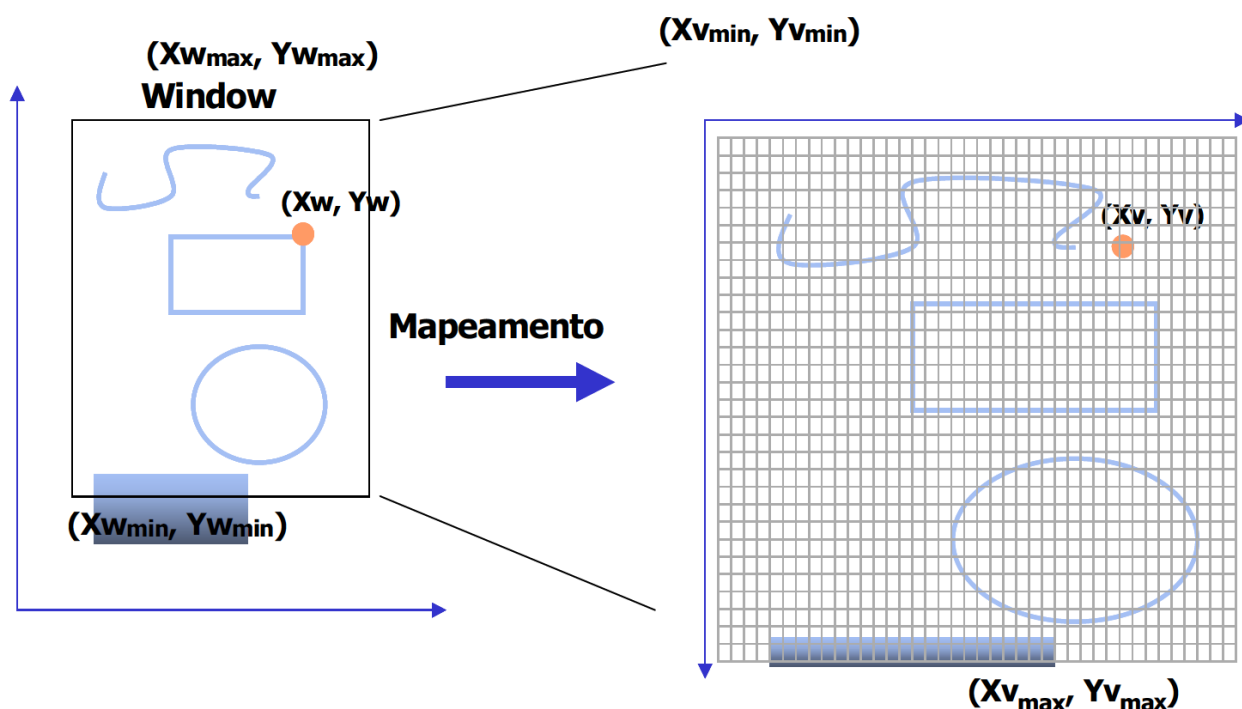
Trabalho Prático 1
Disciplina de Computação Gráfica – 2021
Instituto Federal de Educação, Ciência e Tecnologia do Norte de Minas Gerais – Campus Montes Claros

Tema:

Desenvolva um programa que seja capaz de ler um arquivo contendo informações a respeito dos objetos a serem desenhados, da *window* e da *viewport*, e gerar um arquivo de saída contendo os objetos no sistema de coordenadas de *viewport*.

Descrição:

Para desenvolver este trabalho prático, vocês terão que ler um arquivo texto contendo todas as informações de entrada necessárias (será dado um modelo de arquivo texto em xml para que vocês possam ter ideia do formato básico exigido) e, como saída, deverão gerar um arquivo apenas com os dados dos objetos, porém no sistema de coordenadas da *viewport* (que será obtido pela transformada de *viewport* levando-se em consideração os limites da *window* e as coordenadas de mundo dos objetos).



Nesta primeira versão, vocês não precisarão efetuar o recorte dos objetos (clipping) e não precisarão gerar a interface gráfica do sistema, embora aconselha-se fortemente que isto seja feito (pois será parte integrante da próxima etapa do trabalho). **Para aqueles que implementarem o trabalho com uma interface gráfica (usando QT com o auxílio de canvas) será dado até 5 pontos extras.**

Sobre os requisitos básicos do trabalho:

- O projeto pode ser desenvolvido tanto em Python quanto em C++, porém, será avaliada a qualidade de código e uso das ferramentas de orientação a objetos, (estruturem bem o seu código, criem classes para representação dos diferentes objetos do problema e enviem também instruções de compilação e informações a respeito de bibliotecas extras utilizadas, por exemplo, para ler o arquivo, vocês podem usar um parser para leitura de arquivos XML).
- Criar um conjunto básico de objetos para representar a *window*, o ponto, a reta e o polígono.

- Window: (é como uma janela de recorte do mundo), por enquanto, ela deve armazenar apenas os dados de seus limites (x_{wmin} , y_{wmin} , x_{wmax} , y_{wmax}).
- Ponto: 3 coordenadas reais.
- Reta: 2 Pontos não coincidentes.
- Polígono: lista de, ao menos, 3 pontos onde, o último ponto será ligado ao primeiro. Por enquanto, o polígono poderá ser vazado, ou seja, sem informações do seu interior, também chamado de polígono aramado (ou *wireframe*).

O cálculo da transformada de *Viewport* é definido como uma transformação linear simples entre a *window* e a *viewport*. Para encontrar o valor de x na *viewport*, x_{vp} , realizamos a transformação linear:

$$x_{vp} = \frac{x_w - x_{wmin}}{x_{wmax} - x_{wmin}} \cdot (x_{vpmax} - x_{vpmin})$$

Para encontrar o valor de y na *viewport*, y_{vp} , realizamos uma transformação um pouco diferente, levando em conta a inversão do eixo y :

$$y_{vp} = \left(1 - \frac{y_w - y_{wmin}}{y_{wmax} - y_{wmin}}\right) \cdot (y_{vpmax} - y_{vpmin})$$

Observações finais:

Os trabalhos poderão ser feitos em DUPLAS, porém, códigos iguais entre duplas não serão aceitos, mas vocês poderão debater entre si a respeito da solução do problema e auxílio no uso das tecnologias para desenvolvimento.

É necessário entregar um documento para informar como utilizar seu programa corretamente, isto será utilizado como guia para compilação e teste pelo professor. Caso não seja possível compilar e testar o código o aluno será contactado para esclarecimentos, ou mostrar o funcionamento do programa.