**Relatório Projeto 2 Computação Gráfica e Interfaces**

Neste projeto, decidimos utilizar duas matrizes principais, a mModelView, capaz de guardar os valores necessários para a realização das diferentes transformações, aplicadas aos objetos, com objetivo de mostrar as diferentes projeções existentes. A segunda matriz é a mProjection, também designada como o “volume de visualização”, importante para guardar valores que representem, de uma forma resumida, o quão aproximado o objeto se apresenta do ponto de vista do utilizador. Esta matriz será variada com o uso do “scroll”, permitindo fazer zoom in e zoom out ao objeto pretendido.

1. Para cada uma das projeções, quais os valores que ficam nas matrizes mModel, mView e mProjection?

I – Matriz identidade

|  |  |  |  |
| --- | --- | --- | --- |
| Projeções | mModel | mView | mProjection |
| Alçado Principal | I | [1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1] | [0.6666666666666666,0,0,0,0,0.6666666666666666,0,0,0,0,-0.1,0,0,0,0,1] |
| Planta | I | [1,0,0,0,0,0,-1,0,0,0,-0.9999999999999999,0,0,0,0,1] | [0.6666666666666666,0,0,0,0,0.6666666666666666,0,0,0,0,-0.1,0,0,0,0,1] |
| Lateral Direito | I | [6.123233995736766e-17,0,-1,0,0,1,0,0,1,0,6.123233995736766e-17,0,0,0,0,1] | [0.6666666666666666,0,0,0,0,0.6666666666666666,0,0,0,0,-0.1,0,0,0,0,1] |
| Isometria | I | [0.7071067811865476,0,-0.7071067811865475,0 , -0.40824829046386296, 0.816496580927726,-0.408248290463863,0,0.5773502691896257,0.5773502691896257,0.5773502691896258,0,0,0,0,1] | [0.6666666666666666,0,0,0,0,0.6666666666666666,0,0,0,0,-0.1,0,0,0,0,1] |
| Dimétrica | I | [0.9380821474806573,0,-0.34641288165724754,0,-0.1151820045692663,0.9431035387409434,-0.31191155964105477,0,0.32670321455639784,0.33249919581001963,0.8847085929187115,0,0,0,0,1] | [0.6666666666666666,0,0,0,0,0.6666666666666666,0,0,0,0,-0.1,0,0,0,0,1] |
| Trimetria | I | [0.8739285325641134,0,-0.4860544413646843,0,-0.37565002650222146,0.6345816018773051,-0.6754207933930623,0,0.30844120600078,0.7728558666134543,0.5545789681208176,0,0,0,0,1] | [0.6666666666666666,0,0,0,0,0.6666666666666666,0,0,0,0,-0.1,0,0,0,0,1] |
| Axonométrica Livre | I | [0.8660254037844387,0,0.49999999999999994,0,0.24999999999999994,0.8660254037844387,-0.4330127018922193,0,-0.4330127018922193,0.49999999999999994,0.7500000000000001,0,0,0,0,1] | [0.6666666666666666,0,0,0,0,0.6666666666666666,0,0,0,0,-0.1,0,0,0,0,1] |
| Cavaleira | I | [1,0,-0.7071067811865476,0,0,1,-0.7071067811865475,0,0,0,1,0,0,0,0,1] | [0.6666666666666666,0,0,0,0,0.6666666666666666,0,0,0,0,-0.1,0,0,0,0,1] |
| Gabinete | I | [1,0,-0.3535533905932738,0,0,1,-0.35355339059327373,0,0,0,1,0,0,0,0,1] | [0.6666666666666666,0,0,0,0,0.6666666666666666,0,0,0,0,-0.1,0,0,0,0,1] |
| Oblíqua Livre | I | [1,0,-0.3535533905932738,0,0,1,-0.35355339059327373,0,0,0,1,0,0,0,0,1] | [0.6666666666666666,0,0,0,0,0.6666666666666666,0,0,0,0,-0.1,0,0,0,0,1] |
| Perspetiva | I | [1,0,0,0,0,1,0,0,0,0,0,0,0,0,0.08163265306122448,1] | [0.6666666666666666,0,0,0,0,0.6666666666666666,0,0,0,0,-0.1,0,0,0,0,1] |

1. Onde/Como fazem o ajuste da escala (manipulada com o scroll)?

O scroll, como já foi referido, será aplicado na matriz mProjection, usando a característica deltaY que devolve o deslocamento do “scroll” no sentido positivo ou negativo. Este valor variável será aplicado então a cada momento na mProjection através de um GlobalEventHandler, o “onwheel” aplicado ao canvas. Quando é detetado o “scroll”, o valor deltaY é guardado numa variável zoom, que será aplicada diretamente na matriz mProjection, permitindo variar o volume de visualização.

1. Onde/Como tratam de efetuar o ajuste para que não haja deformação quando se redimensiona a janela?

Para tratar do ajuste e redimensionamento da janela, utilizamos de novo um GlobalEventHandler designado “onresize”, capaz de identificar quando os valores de height e width da janela são alterados. Nesse momento, guardamos os novos valores das dimensões da janela, tal como o mínimo valor entre a dimensão width e height da janela. Depois, guardamos os novos valores das dimensões do canvas a partir das dimensões da janela e aplicamos à função viewport as novas características calculadas: o tamanho do canvas relacionado com o valor mínimo anteriormente calculado para manter sempre o objeto centrado no canvas e o valor mínimo como os dois últimos argumentos do viewport, permitindo que o objeto nunca fique desformatado.

1. Quais os limites que definiram para cada um dos parâmetros de cada projeção?

Para cada um dos parâmetros de projeção, os limites foram os seguintes:

* + Theta: 0º-60º, step: 0.01; Gamma: 0º-60º, step: 0.01;
  + Valor L: 0-1, step:0.01; Alpha: 0º-90º, step: 2
  + Valor D: 0.5-25, step: 0.1