

ATIVIDADE PRÁTICA 2 - IMPLEMENTAÇÃO DO PIPELINE GRÁFICO

GDSCO0051 - Introdução à Computação Gráfica 2021.1

Data de entrega: 11/10/2021, 23h59min.

1 Atividade

Nesta atividade os alunos implementarão a porção geométrica do *pipeline* gráfico, ou seja, os estágios responsáveis por transformarem os vértices do espaço do objeto, tridimensional e com coordenadas reais, até o espaço de tela, bidimensional e com coordenadas inteiras. Uma vez que os vértices estejam no espaço de tela, as arestas entre eles serão rasterizadas utilizando-se o rasterizador de linhas implementado em atividade anterior.

De forma a facilitar o desenvolvimento desta atividade, os alunos utilizarão um *framework* JavaScript fornecido pelo professor.

2 Objetivo

O objetivo deste trabalho é familiarizar os alunos com a estrutura do *pipeline* gráfico através da implementação da sequência de transformações geométricas que o compõem.

3 O Framework

O trabalho deve ser desenvolvido em JavaScript, e o *framework* JavaScript a ser utilizado no desenvolvimento desta atividade se encontra disponível no endereço:

<https://codepen.io/ICG-UFPB/pen/bGwmQpN>

4 Desenvolvimento

Os alunos deverão implementar os estágios geométricos encontrados tipicamente em um *pipeline* gráfico. Estes estágios deverão ser implementados por meio de transformações geométricas descritas na forma de matrizes homogêneas 4×4 .

Abaixo são enumeradas as matrizes de transformação que deverão ser implementadas:

- **Matriz de Modelagem:** Responsável por transformar vértices, originalmente descritos no Espaço do Objeto, para o Espaço do Universo. Esta matriz pode conter um número arbitrário de transformações geométricas (*e.g.* rotações, translações, etc.)
- **Matriz de Visualização:** Responsável por transformar vértices do Espaço do Universo para o Espaço da Câmera. O conteúdo desta matriz deve ser gerado a partir das informações de posição, direção e *up* da câmera.
- **Matriz de Projeção:** Responsável por transformar vértices do Espaço da Câmera para o Espaço de Recorte (ou Projetivo). O único parâmetro a ser utilizado na definição desta matriz será a distância entre o plano de projeção e o centro de projeção da câmera, valor este que, em última análise, determinará a quantidade de distorção perspectiva a ser aplicada à geometria.

- **Homogeneização:** Responsável por transformar pontos do Espaço de Recorte para o Espaço Canônico (ou NDC – *Normalized Device Coordinates*). Esta transformação não envolve o uso de matrizes, e é obtida por meio de um processo chamado de homogeneização, onde todas as coordenadas de um vetor no Espaço de Recorte são divididas por sua coordenada homogênea.
- **Matriz *Viewport*:** Responsável por transformar pontos do Espaço Canônico para o Espaço de Tela.

Uma vez que os vértices estejam no Espaço de Tela, a última etapa deste trabalho consiste em se desenhar a cena final na tela. Para isso, as arestas entre os vértices no Espaço de Tela serão rasterizados utilizando-se o algoritmo de rasterização de linhas implementado em atividade anterior.

Importante 1: Todos os sistemas de coordenadas, com exceção do Espaço de Tela, seguem a convenção *mão direita*.

Importante 2: Não é permitido o uso de nenhuma biblioteca externa. O uso da biblioteca Three.js deve se limitar ao uso das suas funções relacionadas a álgebra linear (representação de matrizes, vetores, e as operações entre estes). Todas as matrizes de transformação exigidas na realização do exercício deverão ser implementadas manualmente, especificando-se o valor de cada elemento das matrizes.

5 Exemplo

O *framework* fornecido pelo professor conta com o modelo de um cubo, com lados de comprimento dois e centralizado na origem. Este modelo é descrito por meio de uma lista de vértices e uma lista de arestas, e foi fornecido *para que os alunos tenham uma referência e possam verificar se os seus pipelines estão implementados de forma correta*. Entretanto, para a entrega do trabalho os alunos deverão incluir, além da image do cubo de referência, imagens de outras geometrias renderizadas com distorção perspectiva.

A figura abaixo ilustra o resultado esperado ao se executar o *pipeline* utilizando-se o modelo do cubo fornecido em conjunto com os seguintes parâmetros:

- Matriz de modelagem igual a identidade.
- Câmera com os seguintes parâmetros: **posição** = (1.3, 1.7, 2.0); **look_at** = (0.0, 0.0, 0.0) e **up** = (0.0, 1.0, 0.0).
- Matriz de projeção com a seguinte composição (como visto em aula):

$$M_{Projecao} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d \\ 0 & 0 & -\frac{1}{d} & 0 \end{bmatrix} \quad (1)$$

, onde $d = 1$.

- Na construção da Matriz *viewport*, considere que a 'tela' (o canvas, neste caso) apresenta resolução de 128×128 pixels.
- Cor utilizada na rasterização das arestas do cubo: R=255, G=0 e B=0.

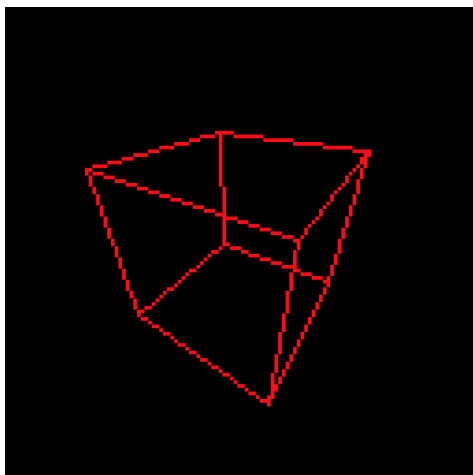


Figure 1: Resultado esperado para o *rendering* do cubo de referência de acordo os parâmetros informados no Capítulo 5.

6 Entrega

Os trabalhos devem ser entregues, via atividade específica do SIGAA, até as **23 horas e 59 minutos** do dia **11/10/2021**. A entrega consistirá em um arquivo compactado (*i.e.* ZIP) contendo um relatório e o código fonte:

1. Relatório no formato PDF, contendo:
 - (a) Nome e número de matrícula do(s) alunos(s).
 - (b) Um parágrafo que descreva a atividade desenvolvida.
 - (c) Breve explicação das estratégias adotadas pelo aluno na resolução da atividade.
 - (d) *Printscreens* e discussão dos resultados gerados, dificuldades e possíveis melhoras.
 - (e) Referências bibliográficas.
 - (f) O aluno pode, se assim desejar, disponibilizar seu código fonte também em um repositório *online* (*e.g.* codepen.io, jsfiddle.net, etc.) e incluir o *link* para este respositório em seu relatório. Entretanto, observa-se que esta disponibilização do código fonte em sites é opcional, não vale nota, e não substitui o envio do código fonte pelo SIGAA.
2. Arquivo contendo o código fonte em JavaScript.

Este trabalho pode ser desenvolvido em duplas.

Importante:

1. Não serão aceitos trabalhos enviados por outro meio que não o SIGAA.
2. Trabalhos entregues até às **23h e 59min** do dia **11/10/2021** serão contabilizados como presença em aula. Trabalhos que **não forem entregues** até às **23h e 59min** do dia **11/10/2021** serão contabilizados como **falta em aula**, e receberão **nota zero**.