

## Aula 3

- Transformações 2D
- Transformações 3D
- Definição de modelos usando faces triangulares
- Remoção de faces ocultas usando “*Back-Face Culling*”

### 1.1 Transformações 2D – Concatenação de sucessivas transformações

Analise o exemplo incompleto **WebGL\_example\_13.html**. Repare no modo como o conteúdo do ficheiro se encontra organizado.

Identifique as principais alterações relativamente aos exemplos anteriores:

- Funções auxiliares que permitem instanciar e multiplicar matrizes de transformação.
- Definição de uma matriz global de transformação – “*Model-View Matrix*” – e aplicação de sucessivas transformações elementares.
- Passagem da matriz global de transformação como argumento ao *vertex-shader*: note como é definida e aplicada a matriz global de transformação a cada vértice.

#### Tarefas:

- Implementar e testar as funcionalidades que permitam aplicar **as transformações elementares em falta**.
- Além da interação usando botões, permitir que o utilizador use o teclado para aplicar as sucessivas transformações.

#### Sugestão:

- Selecione um conjunto adequado de teclas que permita controlar, de modo intuitivo, a aplicação das transformações elementares.

## 1.2 Transformações 2D – Atualização dos parâmetros definindo as três transformações elementares aplicadas

Analise o exemplo incompleto **WebGL\_example\_14.html**. Repare no modo como o conteúdo do ficheiro se encontra organizado.

Identifique a principal alteração relativamente ao exemplo anterior:

- A matriz global de transformação – “*Model-View Matrix*” – resulta da concatenação de **três** transformações elementares: mudança de escala, rotação 2D e translação, por esta ordem.
- Os parâmetros definindo cada uma dessas transformações são diretamente atualizados.

### Tarefas:

- Implementar e testar as funcionalidades que permitam aplicar **as transformações elementares em falta**.
- Além da interação usando botões, permitir que o utilizador use o teclado para aplicar as sucessivas transformações.

## 1.3 Transformações 3D – Rotações em torno dos eixos coordenados

Analise o exemplo incompleto **WebGL\_example\_15.html**. Repare no modo como o conteúdo do ficheiro se encontra organizado.

Identifique a principal alteração relativamente ao exemplo anterior:

- A matriz global de transformação – “*Model-View Matrix*” – resulta da concatenação de **cinco** transformações elementares, para permitir as rotações em torno de cada um dos eixos coordenados.

### Tarefas:

- Implementar e testar as funcionalidades que permitam aplicar **transformações de rotação** em torno dos eixos coordenados.
- Completar, faseadamente, a definição do modelo representando um **cubo centrado na origem**, definindo cada uma das suas faces usando dois triângulos.
- **Atenção** à ordem de definição dos sucessivos vértices de cada triângulo.
- Atribua uma cor diferente a cada um dos triângulos.
- Permitir activar/desactivar a remoção das faces ocultas usando “***Back-Face Culling***”.

**Questões:**

- O que acontece se o modelo intersectar as faces do volume de visualização?
- O que aconteceria se todos os triângulos definindo um modelo tivessem a mesma cor?

**Tarefas:**

- Implementar e testar as funcionalidades que permitam aplicar **as transformações elementares em falta**.
- Definir e visualizar outros **modelos simples**: prisma triangular, paralelepípedo, etc.
- Além da interação usando botões, permitir que o utilizador use o teclado para aplicar as sucessivas transformações.