

## Primeiro Trabalho de Programação:

Usando apenas os conceitos dados nas aulas 6 e 7 ( disponíveis em <http://www2.ic.uff.br/~aconci/CG.html>) desenhe e “anime” a geração de um sólido com os requisitos abaixo descritos.

Esse sólido será mostrado em sala de aula para a professora e aos demais alunos do curso no dia 26/09 ( e ainda prosseguirá sendo desenvolvido nos seus futuros trabalhos de implementação no semestre). Neste dia você deve além de mostrar seu programa explicar como definiu sua estrutura de dados e definiu as diversas matrizes usadas no seu trabalho de implementação, todas elas.

Lembre que não é para usar **ferramentas prontas** e sim você programar as estruturas de dados e operações de multiplicação de matrizes envolvidas. Quanto mais básica sua programação melhor sua nota no curso. **Itens usando ferramentas prontas não serão corrigidos.** Se por acaso usou ferramentas em alguma hora terá que fazer tudo de novo, pois ela não terá algum dos requisitos que pediremos. Assim, também, é mais inteligente você desde o início usar o mínimo de elementos prontos e que seu uso de **sub-rotinas** ou **functions**, da linguagem escolhida, seja o mais básico possível.

Requisitos do programa:

1- Use o objeto que você definiu na **última questão do teste 3**, denominada : “Parte 1 do Primeiro Trab. de implementação” ao desenvolver deste programa. Isto é use a estrutura de dados que você escolheu como sendo o seu sólido e suas coordenadas. Mais especificamente: (A) Use as **coordenadas dos vértices** para desenhá-lo na tela, depois de definir a matriz de projeção no plano xy (ou no plano  $Z=0$ ). Antes disso use a representação do seu objeto em projeções paralelas e em perspectiva na forma que será destinada a você (as mesmas serão usadas em todas as etapas dos nossos futuros trabalhos de implementação); e (B) use a **topologia** para desenhar cada face do seu objeto. Na hora de desenhar **associe a cada face uma cor sólida** (que não seja **branco**, **preto**, **amarelo** e **vermelho**). Chamamos aqui “cor sólida” a que está sem meios tons, ou seja, é constante em toda a face.

2- Estamos com 15 alunos matriculados no curso. Cada aluno antes de desenhar o sólido transformará suas coordenadas de 3D para 2D usando 3 formas dentre as infinitas possibilidades explicadas no pdf da Aula 6: uma em perspectiva com 1 ponto de fuga, outra em projeções isométricas e uma terceira forma de desenho que será em projeção oblíqua. Como abaixo será descrito. Mas a estrutura de dados do objeto sempre permanecerá em 3D, ou melhor em coordenadas homogêneas em 3D. Assim inclua em seu programa uma opção de mostrar na tela as **coordenadas de todos os vértices como uma matriz** ou de indicar a coordenada de um dos vértices que o usuário indique (respondendo, por exemplo, a pergunta que pode aparecer no fim da animação que gera o sólido: **“Você quer ver a coordenada de algum vértice? Diga o numero dele? Se não responder mostrarei todas as coordenadas”**).

3- Se você fez corretamente a primeira questão do Teste 3, então escolheu primeiro um objeto 2D com tantos vértices,  $V$ , quanto for o número de sua ordem na lista de chamada mais 3 e depois fez ele virar algo 3D. Assim cada um dos pontos do seu objeto tinha inicialmente coordenadas  $(x, y, 0)$  e depois  $(x, y, Z_p)$ , onde  $Z_p$  é um plano paralelo ao plano  $Z=0$ . Como nossos elementos vetoriais são retas, ou arestas retas, cada aresta será definida por 2 vértices. Com isso em mente na hora de desenhar o sólido faça o desenho com a **seguinte “animação”**:

a. Se você usou a estrutura de dados **baseada em lados** (edges ou arestas) use esta estrutura de dados para desenhar todas as arestas do seu sólido na cor preta, para desenhar cada aresta use uma opção da sua linguagem de desenhar linha reta de uma cor (que no caso será preta), ainda antes de desenhar cada aresta marque o vértice de início da aresta como um ponto na cor **amarela** e o

vértice final na cor **vermelha**. Desenha este objeto aresta a aresta, como se **fosse uma animação** (use alguma opção de parada da sua linguagem para fazer isso, se preciso). Quando todas as arestas estiverem desenhadas, então desenhe também cada fase, uma de cada cor, também parando depois que cada uma seja desenhada, uma a uma.

b. Se você usou a estrutura de dados **baseada em faces** use esta estrutura de dados para desenhar cada face, uma de cada cor, parando depois que cada uma seja desenhada uma a uma (como uma animação). Faça isso considerando a estrutura de dados das faces baseadas nos vértices e as coordenadas destes que são calculadas como abaixo será esclarecido. Ao desenhar cada vértice use a opção da sua linguagem de por a cor de um ponto, ou pixel na **cor branca**. Mas pinte os vértices face a face na ordem de sua estrutura de dados. De modo que depois de pintar cada face seu programa ficará pintando todos os vértices na cor branca, face por face. Usando a opção do programa para o desenho com alguma interação do usuário, ou pare ele por algum tempo a cada item desenhado, de modo que **os desenhos vão sendo feitos na tela como uma animação**.

4- Para calcular as coordenadas dos pontos originalmente em 3D para dados em 2D antes de serem desenhados, cada aluno usará 3 matrizes diferentes antes de fazer a projeção em  $Z=0$  (veja essa matriz no pdf da aula 6 no slide de título “Projeção paralela ORTOGRAFICA no PLANO  $z=0$ ”). Você fará isso desenhando três vezes seu objeto em três formas diferentes. Uma representará ele em perspectiva (com 1 ponto de fuga) a outra ele estará em uma projeções isométrica e finalmente mostrando ele em uma projeção oblíqua. Mas em cada um dos casos você deve antes desenvolver a matriz que fará isso de acordo com a aula 6 (já mencionada) e com seu número na lista de chamada como definido abaixo.

5- Ainda considere seu objeto transladado na tela por uma transformação de translação que tire ele de sua posição inicial para qualquer lugar do espaço 3D que você quiser de modo que ele apareça melhor no seu desenho. Essa transformação é livre, cada aluno pode fazer experiências livremente, mas não deixe de indicar que matriz em coordenadas homogêneas fará essa transformação.

6- Para definir a matriz de projeção isométrica considere os ângulos mostrados no slide de título “**Projeção paralela isométrica**”. Neste slide, o ângulo teta em torno do eixo x,  $\theta_x$ , deve ser negativo se seu número na lista de chamada for **impar** e o outro ângulo teta com o eixo y,  $\theta_y$ , deve ser positivo. Neste slide, o ângulo teta com o eixo y,  $\theta_y$ , deve ser negativo se seu número na lista de chamada for **par** e o outro ângulo teta com o eixo x,  $\theta_x$ , deve ser positivo. Depois de considerar o sinal destes ângulos adequadamente monte a matriz **Miso** e a use no cálculo de seus pontos em 2D.

7- Para definir a matriz de projeções oblíquas use o slide de título “Projeção paralela oblíqua” suponha projeção **Cabinet**, e considere o ângulo que aparece neste slide como seu número de ordem na chamada multiplicado por 5. Por exemplo se você for o Arthur (número 1 na lista) seu ângulo deve ser 5 graus e se você for o Marcos José (nosso número 15, por enquanto) será 75 graus.

8- Finalmente considere que seu objeto é projetado em perspectiva com um centro de projeção no eixo  $z_{cp}=100+10(\text{seu número de ordem na chamada})$ . Por exemplo, se você for o Arthur (número 1 na lista),  $z_{cp}=100+10=110$ . Se você for o Marcos José (nosso número 15, por enquanto)  $z_{cp}=100+10 \times 15=100+150=250$ .

Qualquer duvida pode entrar em contato primeiro com nosso monitor (sala 518 prédio anexo ao IC) e depois com a professora por e-mail.