

## Parte I - Transformações Geométricas, Câmara, Depth Buffer

1. Considere o seguinte excerto de código :

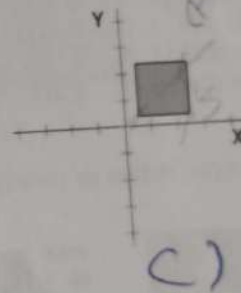
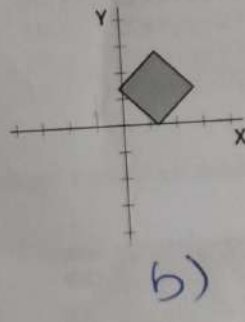
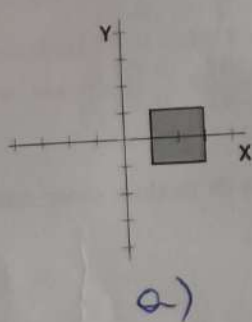
```
translate(0, 0, -3);
drawSphere (); // esfera 1
translate(0, 0, 3);
gluLookAt(3, 0, -3, 3, 0, 0, 0, 1, 0);
translate(0, 0, -3);
drawSphere (); // esfera 2
```

Considerando somente o plano XZ do espaço global, desenhe e identifique a posição das esferas, a posição da câmara e o sistema de eixos da câmara.

2. Considere uma primitiva gráfica para desenhar um cubo com centro na origem e lado com dimensão de 2 unidades, e a seguinte sequência de transformações geométricas a aplicar ao cubo:

```
glRotate(45, 0.0, 0.0, 1.0);
glTranslatef(2.0, 0.0, 0.0);
glRotate(-45, 0.0, 0.0, 1.0);
```

Qual das seguintes opções corresponde ao cubo transformado? Justifique, indicando cada um dos passos intermédios.



3. Considere a matriz A, obtida após uma sequência de transformações geométricas. Indique a sequência incorrecta para gerar a matriz A a partir da matriz identidade, e apresente a matriz resultante da respectiva opção.

$$A = \begin{bmatrix} 4 & 0 & 0 & 4 \\ 0 & 4 & 0 & 4 \\ 0 & 0 & 4 & 4 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

(a) `glTranslatef(1,1,1);`  
`glScalef(4,4,4);`

(b) `glTranslatef(4,4,4);`  
`glScalef(4,4,4);`

(c) `glScalef(4,4,4);`  
`glTranslatef(1,1,1);`

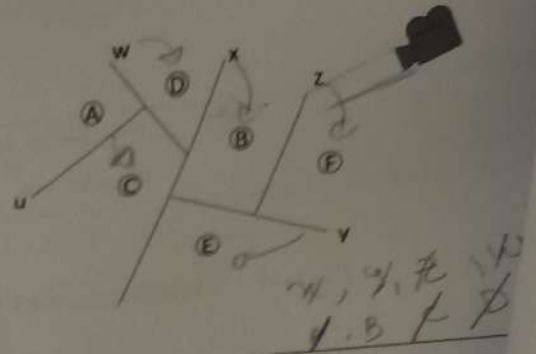
Escola  $\begin{bmatrix} 4 & 4 & 4 \\ 4 & 4 & 4 \\ 4 & 4 & 4 \end{bmatrix}$

T  $\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$

Considere a seguinte divisão do espaço utilizando uma BSP de acordo com a figura.

(a) Construa a árvore definida pela BSB.

(b) Dada a posição da câmara indicada na figura, indique qual a ordem de desenho dos objectos de forma a garantir a ordem de desenho.



## Parte II - Curvas de Bezier; Iluminação; Texturas; View Frustum Culling

Utilizando o método de De Casteljau apresente o diagrama para descobrir graficamente o ponto de uma cúbica de Bézier quando  $t = 0.75$ . Considere os seguintes pontos de controlo (em 2D):

- $P_0 (-1, 0)$
- $P_1 (5, 3)$
- $P_2 (-5, 3)$
- $P_3 (1, 0)$

6. Considere a imagem representativa da mira técnica utilizada pela RTP em 1956 aplicada como uma imagem esquerda, pode ser representado com o seguinte código:

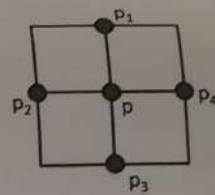
```
glBegin(GL_QUADS);
    glTexCoord2f(0.0, 0.0); glVertex3f(-1.0f, -1.0f, 0.0f);
    glTexCoord2f(1.0, 0.0); glVertex3f( 1.0f, -1.0f, 0.0f);
    glTexCoord2f(1.0, 1.0); glVertex3f( 1.0f,  1.0f, 0.0f);
    glTexCoord2f(0.0, 1.0); glVertex3f(-1.0f,  1.0f, 0.0f);
glEnd();
```

No código que se segue, defina as coordenadas de textura em falta de modo a obter como resultado direita.



```
glBegin(GL_QUADS);
    glTexCoord2f(0.0, 0.0); glVertex3f(-1.0f, -1.0f, 0.0f);
    glTexCoord2f(1.0, 0.0); glVertex3f( 1.0f, -1.0f, 0.0f);
    glTexCoord2f(1.0, 1.0); glVertex3f( 1.0f,  1.0f, 0.0f);
    glTexCoord2f(0.0, 1.0); glVertex3f(-1.0f,  1.0f, 0.0f);
glEnd();
```

7. Considere que se pretende usar uma grelha para representar um terreno. As coordenadas dos pontos da grelha são números inteiros e a dimensão dos lados de cada quadrícula da grelha é uma unidade. Para obter a altura dos pontos da grelha é disponibilizada a função  $h(p_i)$ , sendo  $p_i$  um ponto da grelha. Para se poder calcular a iluminação dos pontos da grelha é necessário calcular a normal em cada ponto. Com base na figura, indique como proceder matematicamente para calcular a normal do ponto  $p$ .

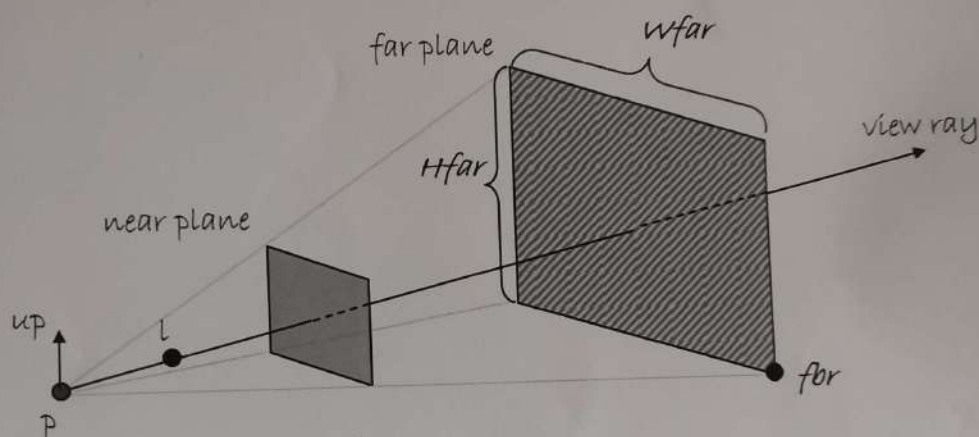


$$h_v = p_3 - p_1$$

$$h_h = p_4 - p_2$$

8. Assuma as seguintes instruções OpenGL:

```
gluPerspective(fov, ratio, nearDist, farDist);
gluLookAt(px, py, pz, lx, ly, lz, upx, upy, upz)
```



Considerando a figura que ilustra um frustum, indique como calcular:

- os valores  $W_{far}$  e  $H_{far}$
- assumindo que os valores da alínea anterior estão calculados, calcule o ponto  $fbr$