INSTITUTO SUPERIOR TÉCNICO

Computação Gráfica

Licenciatura em Engenharia Informática e de Computadores
Alameda / Taguspark

Segundo Teste
18 de Abril de 2012

O teste tem a duração de **1h00**, tolerância incluída. Responda às questões **unicamente** nestas duas primeiras folhas, justificando adequadamente as respostas de desenvolvimento. Só estas duas folhas deverão ser entregues, e como tal, serão as únicas avaliadas. **Identifique-as**! Durante o exame apenas é permitido o uso de caneta e de folhas para rascunho em branco. Não é permitido o uso de calculadoras, telemóveis ou outros dispositivos móveis. Uma resposta errada nas perguntas de escolha múltipla desconta 1/3 da cotação da respectiva questão.

Nota: *cos(10º)=0,985; cos(20º)=0,940; cos(30º) = 0,866; cos(45º) = 0,707; cos(60º) = 0,500; cos(120º)=-0,940*

Respostas: **Versão:** ________

1. a) [1.0v] T_1 = ________________________ T_2 = ________________________

1. b) [1.0v] **1. c) [1.0v]**

2. a) [1.0v] T_1 = ________________________ T_2 = ________________________

2. b) [1.0v] eye_x = ______ eye_y = ______ eye_z = ______

2. c) [0.5v] VPN = ______________________ **2. d) [1.5v]** ________

Identificação do Aluno	
Nome:	Número:

3. a) [1.0v] F = ________ B = ________ RA = ________

3. b) [1.0v] _________

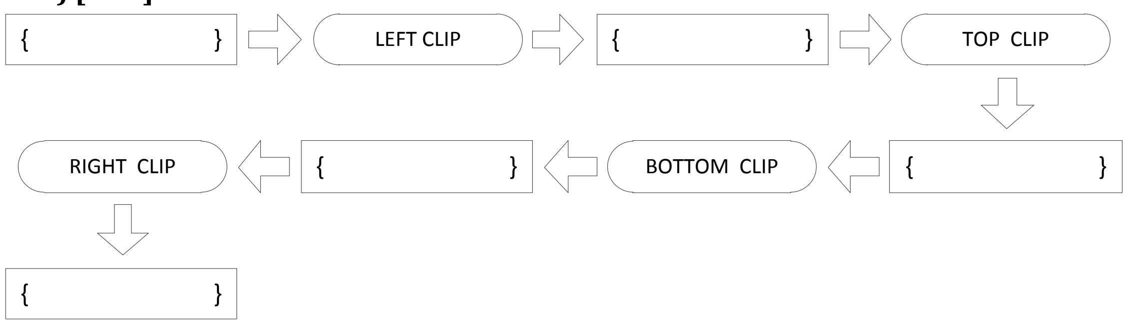
3. c) [1.0v] OC_A: __ __ __ __ __ __ OC_B: __ __ __ __ __ __ OC_C: __ __ __ __ __ __

3. d) [1.0v]

Trivialmente aceite:	**AB BC CA**	(risque o que não interessa)
Trivialmente rejeitada:	**AB BC CA**	(risque o que não interessa)
Subdividida:	**AB BC CA**	(risque o que não interessa)

4. a) [1.0v] OC_A: __ __ __ __ OC_B: __ __ __ __ OC_C: __ __ __ __ OC_D: __ __ __ __

4. b) [2.0v]



5. a) [1.0v] ________ 5. b) [1.0v] θ = ________ 5. c) [2.0v] I =______________________

6. [1.0v] ________

7. [1.0v]

n	*v*	**Face não visível**	**Face visível**
[0,75; 0,5; 0,5]	[0; 0; 1]		
[0,75; 0,5; 0,5]	[-1; -1,5; 2]		

Identificação do Aluno	
Nome:	Número:

1. **[3.0v]** Assuma que a matriz GL_PROJECTION foi inicializada com a matriz identidade. De seguida, introduziu-se o comando:

```
glOrtho(-3.0, 3.0, -2.0, 2.0, 5, 15);
```

Considere que o comando glOrtho() usa um referencial da câmara em que o plano *near* situa-se em z=5 e o plano *far* em z=15. Uma das transformações realizadas internamente pelo OpenGL é a transformação de normalização de modo a gerar um volume de visualização canónico ortogonal que poderá ser $-1\le x,y\le 1$ e $0\le z\le 1$. Sabendo que essa transformação de normalização consiste num produto de duas transformações geométricas, $T_{norm} = T_2 * T_1$

   a) Identifique as duas transformações geométricas.

T_1 – Transformação de Translacção

T_2 - Transformação de escala

   b) Calcule a matriz correspondente à transformação T_1.

$$T_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -5 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

   c) Calcule a matriz correspondente à transformação T_2.

$$T_2 = \begin{bmatrix} 1/3 & 0 & 0 & 0 \\ 0 & 1/2 & 0 & 0 \\ 0 & 0 & 1/(15-5) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

2. **[4.0v]** Assuma que a matriz GL_MODELVIEW foi inicializada com a matriz identidade. De seguida, no processo de estabelecimento da câmara virtual, introduziu-se o comando:

```
gluLookAt(eyex, eyey, eyez, -2.0, 0.0, -2.0, 0.0, 0.0, -1.0);
```

Após a execução deste comando, sabemos que o conteúdo da matriz GL_MODELVIEW resulta do produto de duas transformações geométricas $T_1 * T_2$.

   a) Identifique os dois tipos de transformações geométricas T_1 e T_2.

T_1= Rotação T_2= Translacção

b) Calcule a posição da câmara virtual (os três primeiros argumentos da função `gluLookAt`), sabendo que uma destas transformações é representada pela seguinte matriz:

$$T_? = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

translação de um vector de deslocamento de [-VRPx –VRPy –VRPz]

logo VRP (0 0 -2)

c) Indique a normal ao plano de visualização (view plane normal).

VPN = [-2 0 0] ou [-1 0 0] se normalizado

d) Selecione a matriz que representa a outra transformação geométrica.
(escolha múltipla: indique a opção correcta na página de respostas)

$$A = \begin{bmatrix} 0 & -1 & 0 & 1 \\ 0 & 0 & -1 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & -2 & 0 \\ 1 & 0 & 0 & 2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad C = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$D = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad E = \begin{bmatrix} 1 & -1 & 0 & 0 \\ 0 & 0 & -1 & -1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Para a rotação basta calcular os versores u, v e n.

VPN vale => [-2 0 0] logo n = [-1 0 0]

View-up [0 0 -1] => v'= [0 0 -1] e é ortogonal com n logo v= v'

u = n x v.

u = [0 -1 0]

A matriz de rotação é dada em termos de linhas, respectivamente, pelas componentes dos versores **u,v e -n**.
Assim a resposta correcta é C

3. [4.0v] Considere o seguinte programa OpenGL:

```
void myReshape(GLsizei w, GLsizei h)
{
    glViewport(0, 0, w, h);
    glMatrixMode(GL_PROJECTION);
    glLoadIdentity();
    glOrtho(-2.0f, 2.0f, -2.0f, 2.0f, -2.0f, 2.0f);
}
void myDisplay(void)
```

```
{
    glClearColor(1.0f, 1.0f, 1.0f, 0.0f);
    glClear(GL_COLOR_BUFFER_BIT);
    glColor3f(0.0f, 0.0f, 0.0f);
    glMatrixMode(GL_MODELVIEW);
    glLoadIdentity();
    glTranslate(-0.5f, -0.5f, 0.0f);
    glScale(0.5f, 0.5f, 0.25f);
    glBegin(GL_TRIANGLES); // T
    glVertex3f(2.0f, 2.0f, 4.0f); // A
    glVertex3f(6.0f, 4.0f, 4.0f); // B
    glVertex3f(4.0f, 6.0f, 0.0f); // C
    glEnd();
    glFlush();
}
int main(int argc, char *argv[])
{
    glutInit(&argc, argv);
    glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB);
    glutInitWindowSize (400, 400);
    glutInitWindowPosition (-1, -1);
    glutCreateWindow("Teste");
    glutDisplayFunc(myDisplay);
    glutReshapeFunc(myReshape);
    glutMainLoop();
}
```

a) Indique os valores dos parâmetros F, B e RA da câmara virtual simples definida neste código.

F = -2.0 B = 2.0 RA = 1.0

b) Diga qual das seguintes opções corresponde conteúdo da matriz ModelView imediatamente antes da execução do comando `glBegin()`.
(escolha múltipla: indique a opção correcta na página de respostas)

$$A = \begin{bmatrix} \frac{1}{2} & 0 & 0 & -\frac{1}{2} \\ 0 & \frac{1}{2} & 0 & -\frac{1}{2} \\ 0 & 0 & \frac{1}{4} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad B = \begin{bmatrix} -\frac{1}{2} & 0 & 0 & \frac{1}{2} \\ 0 & \frac{1}{2} & 0 & \frac{1}{2} \\ 0 & 0 & \frac{1}{2} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$C=\begin{bmatrix}\frac{1}{2} & 0 & 0 & 0\\ 0 & \frac{1}{2} & 0 & 0\\ 0 & 0 & -\frac{1}{2} & 0\\ 0 & 0 & 0 & 1\end{bmatrix} \quad D=\begin{bmatrix}\frac{1}{2} & 0 & 0 & \frac{1}{2}\\ 0 & -\frac{1}{2} & 0 & \frac{1}{2}\\ 0 & 0 & \frac{1}{4} & 0\\ 0 & 0 & 0 & 1\end{bmatrix}$$

E: nenhuma das anteriores

$$M_{ModelView}=\begin{bmatrix}1 & 0 & 0 & -1/2\\ 0 & 1 & 0 & -1/2\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}\begin{bmatrix}1/2 & 0 & 0 & 0\\ 0 & 1/2 & 0 & 0\\ 0 & 0 & 1/4 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}=\begin{bmatrix}1/2 & 0 & 0 & -1/2\\ 0 & 1/2 & 0 & -1/2\\ 0 & 0 & 1/4 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}$$

Resposta correcta: A

c) Usando os vértices em coordenadas da câmara e o volume de visualização estabelecido pelo comando `glOrtho`, indique o outcode de cada um dos vértices do triângulo, de acordo com o algoritmo de Cohen-Sutherland (ordem dos bits: z_{min} z_{max} y_{max} y_{min} x_{max} x_{min}).

$$V_{ClippingCoordinates}=M_{Projection}\cdot M_{ModelView}\cdot V_{WorldCoordinates}$$

$$=\begin{bmatrix}\frac{1}{2} & 0 & 0 & 0\\ 0 & \frac{1}{2} & 0 & 0\\ 0 & 0 & -\frac{1}{2} & 0\\ 0 & 0 & 0 & 1\end{bmatrix}\cdot\begin{bmatrix}\frac{1}{2} & 0 & 0 & -\frac{1}{2}\\ 0 & \frac{1}{2} & 0 & -\frac{1}{2}\\ 0 & 0 & \frac{1}{4} & 0\\ 0 & 0 & 0 & 1\end{bmatrix}\cdot V_{WorldCoordinates}$$

$$=\begin{bmatrix}\frac{1}{4} & 0 & 0 & -\frac{1}{4}\\ 0 & \frac{1}{4} & 0 & -\frac{1}{4}\\ 0 & 0 & -\frac{1}{8} & 0\\ 0 & 0 & 0 & 1\end{bmatrix}\cdot V_{WorldCoordinates}$$

$$A_{ClippingCoordinates}=\begin{bmatrix}\frac{1}{4} & 0 & 0 & -\frac{1}{4}\\ 0 & \frac{1}{4} & 0 & -\frac{1}{4}\\ 0 & 0 & -\frac{1}{8} & 0\\ 0 & 0 & 0 & 1\end{bmatrix}\cdot\begin{bmatrix}2\\ 2\\ 4\\ 1\end{bmatrix}=\begin{bmatrix}1/4\\ 1/4\\ -1/2\\ 1\end{bmatrix}$$

$$B_{ClippingCoordinates} = \begin{bmatrix} \frac{1}{4} & 0 & 0 & -\frac{1}{4} \\ 0 & \frac{1}{4} & 0 & -\frac{1}{4} \\ 0 & 0 & -\frac{1}{8} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 6 \\ 4 \\ 4 \\ 1 \end{bmatrix} = \begin{bmatrix} 5/4 \\ 3/4 \\ -1/2 \\ 1 \end{bmatrix}$$

$$C_{ClippingCoordinates} = \begin{bmatrix} \frac{1}{4} & 0 & 0 & -\frac{1}{4} \\ 0 & \frac{1}{4} & 0 & -\frac{1}{4} \\ 0 & 0 & -\frac{1}{8} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 4 \\ 6 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 3/4 \\ 5/4 \\ 0 \\ 1 \end{bmatrix}$$

OC_A: 0 0 0 0 0 0
OC_B: 0 0 0 0 1 0
OC_C: 0 0 1 0 0 0

d) Indique quais as arestas que são trivialmente aceites, rejeitadas ou subdivididas na primeira iteração do algoritmo de Cohen-Sutherland.
(risque o que não interessa)

Todas as arestas são subdivididas.

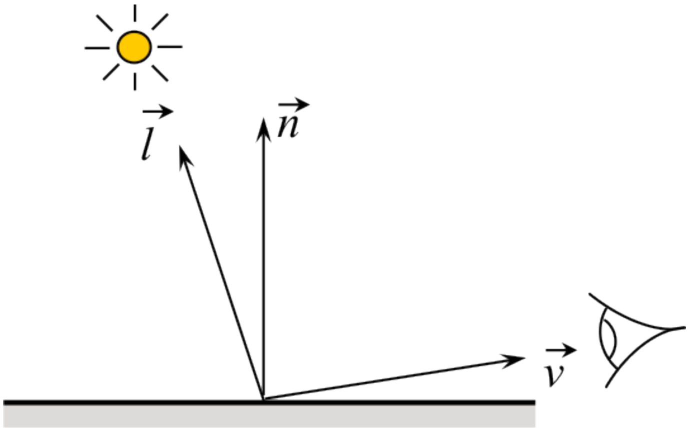
4. **[3.0v]** Considere o polígono *P={A,B,C,D}*, com
$A=[0.1\ 0.6]^T$, $B=[1.2\ \ 0.6]^T$, $C=[1.1\ \ 0.1]^T$ e $D=[0.6\ \ 0.1]^T$,
e o rectângulo de recorte limitado por $x_{min}=y_{min}=0.0$ e $x_{max}=y_{max}=1.0$.

a) De acordo com o algoritmo de Cohen-Sutherland, indique os outcodes associados aos quatro vértices (ordem dos bits: $y_{max}\ y_{min}\ x_{max}\ x_{min}$).
OC_A: 0 0 0 0
OC_B: 0 0 1 0
OC_C: 0 0 1 0
OC_D: 0 0 0 0

b) Indique o conteúdo da lista de vértices à entrada e saída de cada um dos passos do algoritmo de Sutherland-Hodgman aplicado ao polígono P (siga a ordem de recorte fornecida na folha de respostas).

{ A, B, C, D } → LEFT CLIP → { B, C, D, A } → TOP CLIP → { C, D, A, B } → BOTTOM CLIP → { D, A, B, C } → RIGHT CLIP → { A, B', C', D }

5. **[4.0V]** Considere a cena ilustrada na figura abaixo, com uma fonte de luz e uma superfície plana.



Nesta cena, a fonte de luz faz um ângulo com a superfície de 70º e o observador olha para a superfície segundo um ângulo de 10º. As características de iluminação e de reflexão são descritas pelas seguintes funções em OpenGL:

```
GLfloat ambient[] = { 0.1, 0.1, 0.1, 1.0 };
GLfloat diffuse[] = { 1.0, 0.0, 0.0, 1.0 };
GLfloat specular[] = { 0.0, 0.0, 1.0, 1.0 };
GLfloat mat_ambient[] = { 0.5, 0.5, 0.5, 1.0 };
GLfloat mat_diffuse[] = { 1.0, 1.0, 1.0, 1.0 };
GLfloat mat_specular[] = { 1.0, 1.0, 0.0, 1.0 };

glLightfv(GL_LIGHT0, GL_AMBIENT, ambient);
glLightfv(GL_LIGHT0, GL_DIFFUSE, diffuse);
glLightfv(GL_LIGHT0, GL_SPECULAR, specular);
glMaterialfv(GL_FRONT_AND_BACK, GL_AMBIENT, mat_ambient);
glMaterialfv(GL_FRONT_AND_BACK, GL_DIFFUSE, mat_diffuse);
glMaterialfv(GL_FRONT_AND_BACK, GL_SPECULAR, mat_specular);
glMaterialfv(GL_FRONT, GL_SHININESS, 10.0);
```

a) Indique que componentes fazem parte do modelo de reflexão de **Phong**.
(escolha múltipla: indique a opção correcta na página de respostas)

A: Ambiente, Difusa e Lambert
B: Difusa, Especular e Brilho
C: Ambiente, Especular e Difusa
D: Difusa, Global e Brilho
E: Ambiente, Difusa e Reflexiva

Resposta correcta: C

b) Calcule, de acordo com a aproximação de **Blinn**, o valor do ângulo entre o halfway vector e o vector normal à superfície (*n*).

$\Theta = 30°$

c) Calcule a cor do ponto da superfície para onde o observador está a olhar segundo o modelo de reflexão de **Blinn-Phong**.

I = [0,99; 0,05; 0,05]

$I_R = 0{,}1*0{,}5 + 1{,}0*1{,}0*\cos(20^o) + 0{,}0$
$I_G = 0{,}1*0{,}5 + 0{,}0 + 0{,}0$
$I_B = 0{,}1*0{,}5 + 0{,}0 + 0{,}0$

6. **[1.0V]** Na remoção de faces traseiras de um poliedro côncavo por back-face culling…

(escolha múltipla: indique a opção correcta na página de respostas)

A: ... nem todas as faces traseiras são removidas
B: ... exatamente metade das faces frontais ocultas é removida
C: ... todas as faces não visíveis são removidas
D: ... algumas faces não visíveis poderão não ser removidas
E: ... aproximadamente metade das faces traseiras são removidas

Resposta correcta: D

7. **[1.0V]** Em coordenadas do Mundo, dados os vectores das suas normais (*n*) e os correspondentes vectores de visualização (*v*), indique quais das faces são faces visíveis e não visíveis.

(para cada par ***n***,***v*** assinale com uma cruz a opção correcta na página de respostas)

n	***v***	**Face não visível**	**Face visível**
[0,75; 0,5; 0,5]	[0; 0; 1]	X	
[0,75; 0,5; 0,5]	[-1; -1,5; 2]		X