INSTITUTO SUPERIOR TÉCNICO

# Computação Gráfica

Licenciatura em Engenharia Informática e de Computadores
Alameda/Taguspark

**2º Exame**
15 de Julho de 2010

O exame tem a duração de **2h30**, tolerância incluída. Responda às questões **unicamente** nestas três primeiras folhas, justificando adequadamente as respostas de desenvolvimento. Só estas duas folhas deverão ser entregues, e como tal, serão as únicas avaliadas. As restantes folhas podem ser utilizadas como folhas de rascunho. Durante o exame apenas é permitido o uso de caneta. Não é permitido o uso de calculadoras, telemóveis ou folhas em branco para rascunho. Uma resposta errada nas perguntas de escolha múltipla desconta 1/2 da cotação da respectiva questão.

**Respostas:**

**1. [2.0v]** RGB($T_{[6,3]}$) = < ________ , ________ , ________ >

**2. a) [1.0v]** ______________________________________________________________

**2. b) [0.5v]** ____________________

**3. [0.5v]** ____________________

**4. [1.5v]** ____________________

**5. [1.0v]** HSV(P) = < ________ , ________ , ________ >

| Identificação do Aluno | |
|---|---|
| Número: | Nome: |

**6. [1.0v]**

**7. [2.0v]**

| | | |
|---|---|---|
| $R_{amb}$ | $R_{dif}$ | $R_{esp}$ |
| $G_{amb}$ | $G_{dif}$ | $G_{esp}$ |
| $B_{amb}$ | $B_{dif}$ | $B_{esp}$ |

**8. a) [1.0v]**

**8b) [1.0v]**

**9. a) [0.5v]** ___________________

**9. b) [1,5]**

| | Topo | Base | Esquerda | Direita |
|---|---|---|---|---|
| $T_E$ | | | | |
| $T_L$ | | | | |

**9. c) [1.0v]** ______________________________________________________________________________

______________________________________________________________________________________

______________________________________________________________________________________

______________________________________________________________________________________

| **Identificação do Aluno** | |
|---|---|
| Número: | Nome: |

**10. a) [1.0v]** ______________________________________________

**10. b) [1.0v]** ____________

**11. [1.0v]**

**12. a) [0.5v] Vectorial** ☐ **Quadrícula** ☐

**12. b) [1.5v]** **d =** ______________

**x =** ______________

**y =** ______________

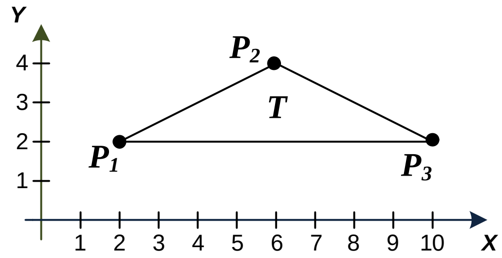
**incrE =** ____________

**incrNE =** ___________

| Identificação do Aluno | |
|---|---|
| Número: | Nome: |

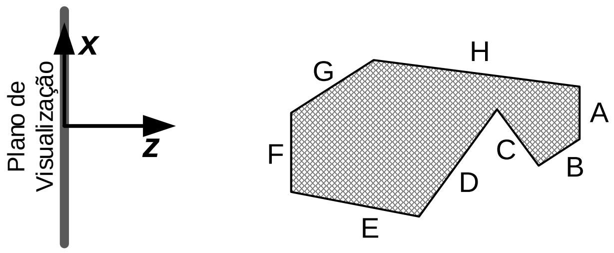
1. **[2.0v]** Está a proceder ao sombreamento do polígono *T*, representado na figura abaixo através da aplicação do método proposto por Gouraud. Neste contexto, calcule a cor da quadrícula com coordenadas [6,3], sabendo que:

   RGB($P_1$)=<1.0,0.0,0.8>, RGB($P_2$)=<0.6,0.4,0.8> e RGB($P_3$)=<0.2,0.8,0.0>




RGB($T_{[6,3]}$) = <0.6, 0.4, 0.6>

2. **[1.5v]** A figura seguinte representa a vista de topo de um objecto já depois de aplicadas as transformações de visualização e projecção.

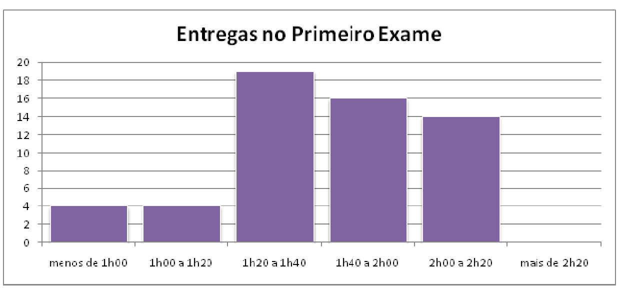



   **a)** Quais as facetas removidas pelo algoritmo de *back-face culling*.
   H, A, B, D

   **b)** Indique quantas vezes foi necessário calcular o produto interno na aplicação do algoritmo de *back-face culling* neste caso em particular.

   Nenhuma, pois o algoritmo de *back face culling* não necessita de calcular o produto interno. Limita-se a considerar o sinal da componente *Z* da normal.

3. **[0.5v]** Imagine que lhe pediram para criar o gráfico de barras ilustrado na figura abaixo e publicá-lo na internet como complemento visual à informação textual existente numa página. Que formato considera mais apropriado para o efeito?




**A:** BMP
**B:** PNG
**C:** JPEG
**D:** XLS
**E:** TIF

B: PNG

4. **[1.5v]** Quantos bytes são necessários para armazenar uma imagem com 400x600 quadrículas, com profundidade de cor 24 bits e canal alfa com 256 níveis, usando a representação directa?

   400*600 = 24000 quadrículas
   24+8 = 32bits = 4 Bytes
   24000*4 = **96000 Bytes**

5. **[1.0v]** Considere a cor do vértice P representada no modelo HSV pelo tuplo HSV(P)=<160, 0.5, 0.7>. Se pretendesse alterar a pureza da cor do vértice P, tornando-a completamente pura, que valores que iria atribuir ao tuplo HSV(P)?

   HSV(P)=<160, 1.0, 1.0>
   **Nota:** HSV(P)=<160, 1.0, 0.7> também é aceite como correcto

6. **[1.5v]** Considere a seguinte transformação descrita no espaço cartesiano:

   x' = x, y' = -y e z' = -z

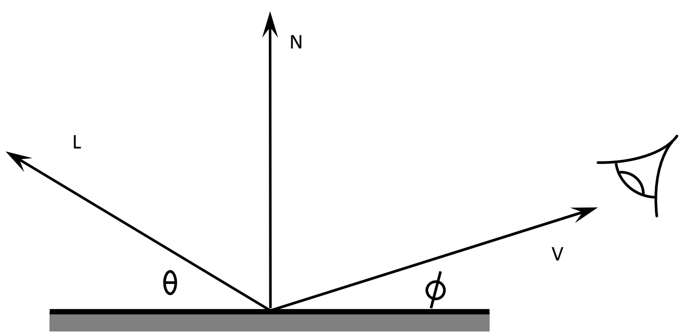
   Determine a matriz correspondente a esta transformação em coordenadas homogéneas.

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

7. **[2.0v]** Considere uma cena muito simples constituída por uma fonte de luz e uma superfície plana. As características de iluminação e de reflexão desta cena são descritas pelas seguintes funções OpenGL:

```
GLfloat ambient[] = { 0.1, 0.1, 0.1, 1.0 };
GLfloat diffuse[] = { 0.0, 1.0, 1.0, 1.0 };
GLfloat specular[] = { 1.0, 1.0, 1.0, 1.0 };
glLightfv(GL_LIGHT0, GL_AMBIENT, ambient);
glLightfv(GL_LIGHT0, GL_DIFFUSE, diffuse);
glLightfv(GL_LIGHT0, GL_SPECULAR, specular);

GLfloat mat_ambient[] = { 0.5, 0.5, 0.5, 1.0 };
GLfloat mat_diffuse[] = { 0.6, 0.6, 0.0, 1.0 };
GLfloat mat_specular[] = { 0.8, 0.8, 0.8, 1.0 };
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);
glMaterialf (GL_FRONT, GL_SHININESS, 100.0);
```



Na figura acima, considere ainda que a fonte de luz está apontada à superfície de modo a realizar um ângulo θ de 30º com o plano; e que o observador olha para a superfície segundo um ângulo ø de 15º também com o plano. Calcule as componentes de cor - ambiente, difusa e especular - no ponto da superfície para onde a câmara aponta segundo o modelo de reflexão de Phong.

**Nota:** cos 15º ≈ 0,967, cos 30º ≈ 0,866, cos 45º ≈ 0,707 e cos 60º = 0,5

*VertexColor = ambient light * ambient material + (max {L . N, 0}) * diffuse light * diffuse material + (max {R . V, 0}) exp shininess * specular light * specular material*

*R:* $0{,}1 \cdot 0{,}5 + 0 \cdot 0{,}6 \cdot \cos 60^o + 1{,}0 \cdot 0{,}8 \cdot (\cos 15^o)^{100}$

*G:* $0{,}1 \cdot 0{,}5 + 1 \cdot 0{,}6 \cdot \cos 60^o + 1{,}0 \cdot 0{,}8 \cdot (\cos 15^o)^{100}$

*B:* $0{,}1 \cdot 0{,}5 + 1 \cdot 0{,}0 \cdot \cos 60^o + 1{,}0 \cdot 0{,}8 \cdot (\cos 15^o)^{100}$

**8. [2.0v]** Suponha que escreveu a seguinte linha de código, no contexto OpenGL:

```
gluLookAt(0.0, -1.0, 3.0, -2.0, -1.0, 3.0, 0.0, 0.0, -1.0);
```

**Nota:** Assinatura da função:

```
void gluLookAt( Gldouble eyex,
                Gldouble eyey,
                Gldouble eyez,
                Gldouble centerx,
                Gldouble centery,
                Gldouble centerz,
                Gldouble upx,
                Gldouble upy,
                Gldouble upz)
```

**a)** Indique a matriz de translação em coordenadas homogéneas referente à transformação de visualização realizada internamente pelo pipeline OpenGL de modo a colocar os objectos da cena no referencial da câmara.

**b)** Calcule a matriz de rotação em coordenadas homogéneas realizada internamente pelo pipeline OpenGL na transformação de visualização.

*A transformação de visualização implica uma mudança de referencial do Mundo para o referencial da Câmara. Assim, em termos de mudança de referencial, esta transformação equivale a fazer coincidir o referencial da* *câmara com o referencial do Mundo. Isso implica efectuar uma translação seguida de uma rotação. A matriz de translação é definida pelo vector de translacção [-VRPx –VRPy –VRPz]. A matriz de rotação é dada em termos de linhas, respectivamente, pelas componentes dos versores u,v e -n.*

*VRP (1º argumento) é o (0 -1 3) e o observador está olhar para (-2 -1 3) (2º argumento) logo VPN [-2 0 0] => n [-1 0 0]*

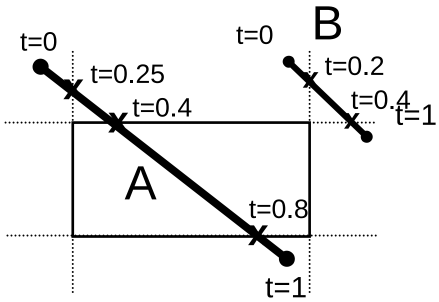
*View-up [0 0 -1] (3º argumento) => v [0 0 -1] e portanto u [0 -1 0] (referencial mão esquerda: u = n x v)*

*Internamente efectuam-se uma translação seguida de uma rotação:*

*RESPOSTA:*

$$\text{Translacção} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & -3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{Rotação} = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ +1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

9. **[3.0v]** Considere o Algoritmo Paramétrico de recorte com a seguinte ordem de recorte ao nível das fronteiras de recorte: topo, base, esquerda e direita.




**a)** O objectivo do Algoritmo Paramétrico é:

**A:** Calcular os OUTCODES dos pontos extremos

**B:** Calcular o menor valor de $t$ para $T_E$ e o maior valor de $t$ para $T_L$

**C:** Calcular as intersecções nas fronteiras de recorte

**D:** Calcular os valores do parâmetro $t$ na janela de recorte

**E:** Calcular o maior valor de $t$ para $T_E$ e o menor valor de t para $T_L$

Resposta E: encontrar o maior Te e o menor Tl (se não ocorrer rejeição trivial)

**b)** Tendo em conta a ordem de recorte acima explicitada, determine o par de valores ($T_E$, $T_L$), no recorte do segmento de recta A face a cada uma das fronteiras de recorte.

**Nota:** relembre-se que os valores de $T_E$ e $T_L$ são inicializados a 0 e 1, respectivamente.

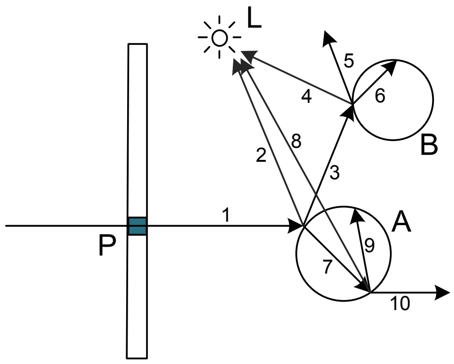
| Te | Tl | |
|---|---|---|
| 0 | 1 | inicial |
| 0.4 | 1 | /* topo: o t de intersecção é maior que o Te logo este assume o novo valor |

0.4 0,8 /* base: o t de intersecção é menor que o Tl logo este assume o novo valor
0.4 0,8 /* esq.: o t de intersecção é menor que o Te logo mantém-se
0.4 0.8 /* direita: t de intersecção é maior que o Tl logo este mantém-se

**c)** Explique como é que o algoritmo rejeita o segmento de recta B

Inicialmente $T_3$=0 e $t_L$=1
Na fronteira de topo fica com $t_E$= 0,4 e $t_L$=1
Quando recortar contra fronteira da direita fica $t_E$=0,4 e $t_L$=0,2.
Assim $T_E$(0,4) > $T_L$ (0,2) logo rejeição

**10. [2.0v]** Considere a seguinte cena, constituída por uma fonte de luz pontual L e duas esferas A e B. Ao recorrer ao algoritmo de *ray-tracing* para a rasterização desta cena definiu-se 2 como número máximo de reflexões.




**a)** Considere que a esfera A é translúcida e reflectora e B apenas reflectora. De todos os raios representados na figura quais são calculados para determinar a cor da quadrícula P?

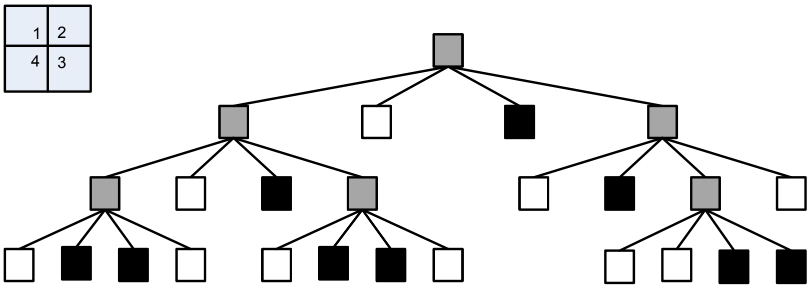
1, 2, 3, 4, 5, 7, 8, 9, 10

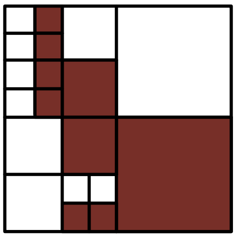
**b)** Indique qual o tipo do raio **10**:

**A:** Reflectido
**B:** Refractado
**C:** *Shadow Feeler*
**D:** Direccionado
**E:** Primário

B: Refractado

**11. [1.0]** Considere a representação por *quadtree* ilustrada na árvore seguinte, onde os quadrantes aparecem da esquerda para a direita. Desenhe na **folha de respostas** a representação no espaço 2D correspondente a esta árvore, assumindo a ordem de identificação de partições indicada.



**12. [2.0v]** Considere que se pretende representar uma linha [AB] num dispositivo gráfico, sendo A=<10,10> e B=<12,11>. O método mais indicado para o fazer neste dispositivo é através da aplicação do algoritmo de Bresenham.

**a)** Na sua opinião este dispositivo gráfico é vectorial ou de quadrícula?

Quadrícula

**b)** Diga quais os valores iniciais para este caso, no algoritmo de Bresenham, das variáveis **d**, **x**, **y**, **incrE** e **incrNE**.

```
d = 2*dy – dx = 2 – 2 = 0
x = 10
y = 10
incrE = 2*dy = 2
incrNE = 2*(dy - dx) =  -2
```