

# - COMPUTAÇÃO GRÁFICA E INTERFACES -

MIEI/FCT/UNL – Ano letivo 2014/2015

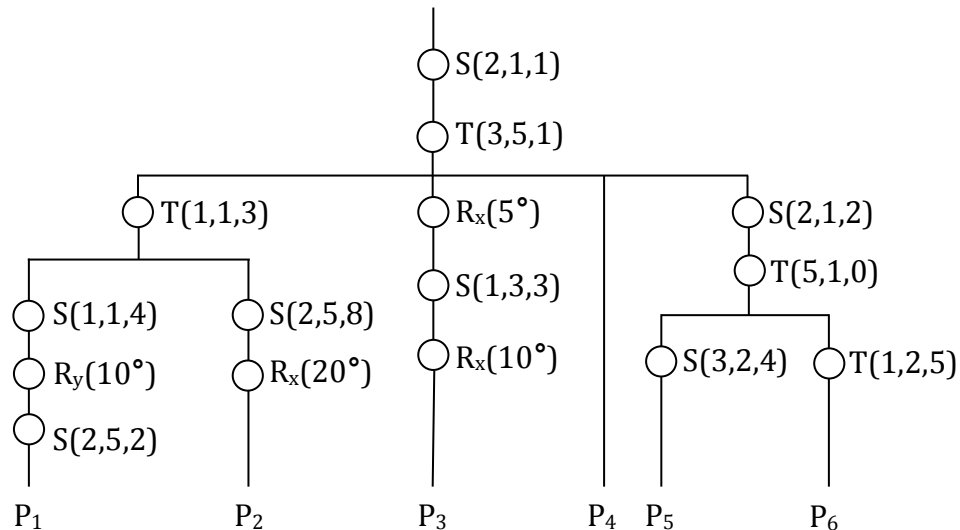
Teste 2 – 2014.12.11

Responda no próprio enunciado, que entregará. Em caso de engano, e se o espaço para as respostas não for suficiente, poderá usar o verso das folhas desde que feitas as devidas referências.

**Não desagrafe** as folhas! A prova, com duração de **1H30**, é **sem consulta**!

## 1. (4 valores)

No grafo de cena abaixo representado, os  $P_i$  representam primitivas geométricas, onde se sabe não ocorrerem quaisquer transformações geométricas.



a) Escreva, em pseudo-código, o programa OpenGL otimizado correspondente, usando o menor número de operações `glPushMatrix()` e `glPopMatrix()`:

S(2,1,1)	S(2,5,8)	P4()
T(3,5,1)	Rx(20°)	S(2,1,2)
glPushMatrix()	P2()	T(5,1,0)
T(1,1,3)	glPopMatrix()	glPushMatrix()
glPushMatrix()	glPushMatrix()	S(3,2,4)
S(1,1,4)	Rx(5°)	P5()
Ry(10°)	S(1,3,3)	glPopMatrix()
S(2,5,2)	Rx(10°)	T(1,2,5)
P1()	P3()	P6()
glPopMatrix()	glPopMatrix()	

b) Simplifique o grafo apresentado por forma a usar o menor número de nós possível e apresente, abaixo, a composição de transformações geométricas e as primitivas às quais se aplicam, apenas

para os ramos onde tenham ocorrido simplificações (por ramo entenda-se o percurso da raiz do grafo até uma dada primitiva):

Ramo 1: S(2,1,1).T(3,5,1).S(2,5,8).Ry(10°).P1

Ramo 2: S(2,1,1).T(3,5,1).Rx(15°).S(1,3,3).P3 (ou S(2,1,1).T(3,5,1).S(1,3,3).Rx(15°).P3)

Ramo 3: O caminho da raiz até P2 também sofre modificações no grafo simplificado, mas as transformações geométricas são exatamente as mesmas e pela mesma ordem

c) Indique o número total de nós do grafo após a simplificação: 12 nós

## 2. (3 valores)

a) Indique qual o nome da componente que faria variar, para uma dada cor de partida nas condições pedidas, definida no modelo **HLS**, de modo a:

- percorrer a escala de cinzentos: Lightness (L)
- percorrer as cores principais do modelo RGB e CMY: Hue (H)
- passar duma cor para o seu equivalente em tons de cinzento: Saturation (S)

b) Se o modelo **HSV** for representado por um cone, como descreveria o lugar geométrico (ponto, linha, triângulo, retângulo, círculo, circunferência, etc.):

- Todas as cores com máxima saturação: superfície cônica (exceptuando a base)
- Todos os tons de cinzento? segmento de reta (ou linha)
- Todas as cores com um mesmo tom? triângulo

## 3. (2,5 valores)

Atendendo ao pipeline gráfico 3D estudado nas aulas, **indique, justificando se:**

- A iluminação poderá ou não ser efetuada após a transformação do volume de visão para o volume canónico: Não pode ser efetuada após a transformação indicada.

A transformação para o volume canónico causa, no caso geral, deformações que alteram os valores dos ângulos, alterando assim os resultados das fórmulas envolvidas nos modelos de iluminação. Também as distâncias são modificadas, o que altera o resultado caso se utilize atenuação das fontes de luz.

- A transformação do volume de visão para o volume canónico, representado por um cubo, é razão suficiente para que haja deformação dos objetos na sua visualização final, num visor: Não é razão suficiente para que haja deformação dos objetos na sua visualização final num visor. Embora haja uma deformação no caso geral (quando o volume de visão não é um cubo), essa deformação poderá ser compensada na transformação para o visor, já perto do final do pipeline, desde que o aspect ratio do visor tenha o mesmo valor que a relação entre a largura e a altura do volume de visão.

#### 4. (3 valores)

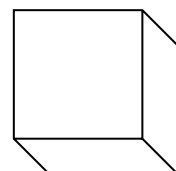
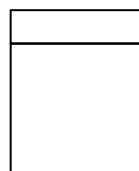
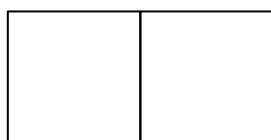
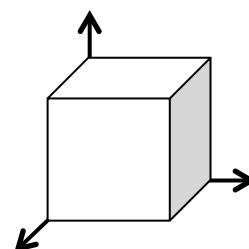
Assinale as afirmações com verdadeiro (V) ou falso (F) na quadrícula da esquerda! **Cada resposta errada descontará 50% da cotação!**

Segundo o modelo de reflexão difusa  $I_{rgb} = I_{p,rgb} K_{d,rgb} \cos(\theta)$ :

F	a cor observada num ponto numa face plana dum objeto só depende da orientação dessa mesma face e da cor do próprio objeto
V	quando o ângulo entre <b>N</b> e <b>L</b> é $90^\circ$ , o objeto não refletirá qualquer luz na direção do observador
F	quando $(\mathbf{N} \cdot \mathbf{L}) = 1$ , assumindo que ambos os vetores estão normalizados, o objeto refletirá a totalidade da luz nele incidente
V	um objeto de cor RGB(1, 0.5, 0.7) poderá ser visto como tendo a cor RGB(0.8, 0, 0.1)
V	para que um objeto cinzento seja visível bastará que uma das componentes da luz não seja nula e a face esteja parcialmente virada para a luz
V	Quando visível, um objeto de cor verde RGB(0,k,0), será sempre visualizado em tons de verde

#### 5. (4.5 valores)

O cubo apresentado ao lado foi projetado recorrendo a diversas projeções planas, apresentando-se a imagem final de cada uma dessas projeções em baixo. Indique, para cada imagem, o tipo de projeção que poderá ter sido utilizado, assim como os nomes e valores dos respetivos parâmetros! Caso exista mais do que uma resposta, bastará indicar apenas uma das hipóteses.



Projeção	oblíqua	ortogonal	oblíqua	oblíqua	axonométrica
Par. 1 (nome, valor)	$l=1$	-	$l=0.2$	$l=0.5$	$\theta=0^\circ$
Par. 2 (nome, valor)	$\alpha=0^\circ$ (ou $180^\circ$ )	-	$\alpha=90^\circ$	$\alpha=-45^\circ$	$\gamma=45^\circ$

a) Para qualquer das imagens acima, qual o número máximo de vezes que um determinado pixel da imagem poderia vir a ser pintado, assumindo que foi usado o algoritmo de z-buffer para efetuar a remoção das superfícies ocultas? 2 Justifique!

Considerando apenas o preenchimento de superfícies, em cada pixel serão projetadas,

no máximo, 2 faces. Ocorrendo em primeiro lugar a pintura da face mais afastada, o número máximo de vezes que um pixel poderá ser pintado seria, assim, 2 vezes.

(Se se considerarem também as linhas, há situações em que poderiam ser pintados 4 vezes)

b) E no caso de se usar previamente o teste de *culling* de faces, como seria a sua resposta à alínea anterior?

No caso de usar o teste de culling de faces, as faces não visíveis seriam previamente descartadas. Assim, considerando apenas o preenchimento das faces, a resposta seria 1 única vez, para as faces visíveis.

(Se se considerar também o desenho das linhas, a resposta dependeria da forma como as mesmas foram desenhadas: 2 para outline de polígonos e 3 para segmentos de reta)

## 6.

a) Complete o seguinte trecho dum programa em OpenGL (JOGL2), por forma a que faça sentido e que o mapeamento da textura seja o mapeamento ortogonal classico, segundo a direção do eixo y:

```
gl.____glBegin____(GL_POLYGON);  
  
for(int i=0; i<nvert; i++) {  
    gl.glTexCoord2f(____x[i]____, ____z[i]____);  
    gl.____glVertex3f____(x[i], y[i], z[i]);  
}  
gl.____glEnd____();
```

b) Que problema poderá surgir quando uma textura, de uma determinada resolução, é mapeada num polígono, cuja área projetada no visor (em pixels) é bastante superior à dimensão (em texels) da textura e de que forma poderá ser atenuado?

Nas condições enunciadas, a textura irá ser ampliada. O procedimento normal, mais simples, consiste em usar um mesmo texel para pintar vários pixels. O efeito que daqui resulta é geralmente denominado por efeito mosaico. Em alternativa a textura poderá ser filtrada, usando não apenas a cor de um texel, mas a de vários, para determinar a cor de cada pixel. O processo poderá ser efetuado por interpolação, suavizando assim as transições dum texel para um outro seu vizinho.

*Boa sorte!*