

N.º _____ Nome _____

Duração da prova: 75 minutos

Perguntas de escolha múltipla: cada resposta incorrecta desconta 1/3 do valor da pergunta

Parte Teórica

30% – 8 valores mín.

- a. Da comparação de uma interface WIMP (*Windows, Icons, Menus and Pointers*) com uma de linha de comandos resulta normalmente que
- i. A segunda está mais vocacionada do que a primeira para ser usada por utilizadores experientes
 - ii. O esforço de memorização e a carga cognitiva impostos aos utilizadores são maiores na segunda do que na primeira
 - iii. A probabilidade de ocorrência de erros de interação é maior na segunda do que na primeira
 - iv. Todas as anteriores
- b. Um sistema gráfico dotado de um *frame buffer* RGBA de 1024 x 1024 x 32 bits
- i. Permite a reprodução de imagens compostas por mais do que 1 milhão de píxeis
 - ii. Permite a reprodução de imagens com 256 níveis de vermelho, 256 níveis de verde e 256 níveis de azul
 - iii. Permite a reprodução de imagens com mais de 16 milhões de cores
 - iv. Todas as anteriores
- c. Em que condições é que um vector não é alterado por uma transformação linear afim?
- i. Quando o vector for nulo
 - ii. Quando a transformação for uma translação
 - iii. Quando a transformação for representada pela matriz identidade
 - iv. Todas as anteriores
- d. No modelo da câmara virtual adoptado pelo OpenGL, as transformações de modelação servem para
- i. Posicionar os objectos na cena e, se for caso disso, deformá-los
 - ii. Posicionar e orientar a câmara
 - iii. Ampliar e reduzir a fotografia tirada pela câmara
 - iv. Nenhuma das anteriores

- e. O sistema de equações $x = \cos(u)$, $y = \sin(u)$, $z = v$, em que $0 \leq u < 2\pi$ e $-0.5 \leq v \leq 0.5$, constitui uma parametrização válida da superfície de
- Um cubo unitário centrado na origem
 - Um cilindro de raio e altura unitários centrado na origem
 - Uma esfera unitária centrada na origem
 - Nenhuma das anteriores
- f. O conhecimento do vector normal é necessário ao cálculo
- Das componentes ambiente e difusa de iluminação
 - Das componentes ambiente e especular de iluminação
 - Das componentes difusa e especular de iluminação
 - Das componentes ambiente, difusa e especular de iluminação
- g. No modelo de iluminação do OpenGL é possível definir um factor de atenuação da intensidade luminosa
- Constante, isto é, que não depende da distância da fonte de luz ao objecto iluminado
 - Linear, isto é, proporcional à distância da fonte de luz ao objecto iluminado
 - Quadrático, isto é, proporcional ao quadrado da distância da fonte de luz ao objecto iluminado
 - Todas as anteriores
- h. A correcção perspectiva permite
- Corrigir o efeito de discretização (*aliasing*) que decorre da utilização de *frame buffers* de baixa resolução
 - Corrigir o efeito de diminuição da dimensão aparente de um objecto quando a distância do mesmo à câmara aumenta
 - Corrigir o efeito de deformação que decorre da utilização de técnicas simples de interpolação linear no mapeamento de texturas em polígonos
 - Nenhuma das anteriores

N.º _____ Nome _____

Parte Teórico-Prática

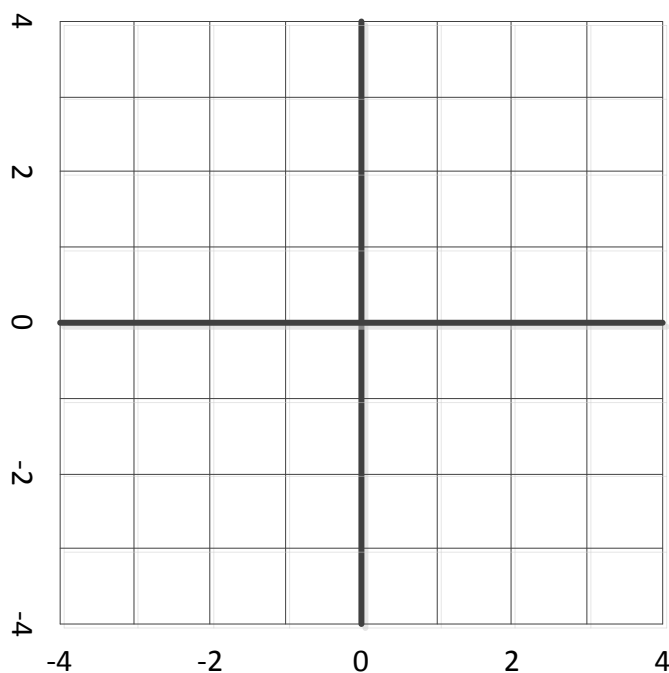
40% – 8 valores mín.

Nota:

Em todas as perguntas, a menos que algo seja dito em contrário, assuma a posição da câmara por omissão e o semieixo positivo dos ZZ como representando a cota positiva.

- a. Esboce o desenho que resulta do seguinte extracto de código.

```
glBegin(GL_LINE_LOOP);  
    glVertex2f(1.0, 1.0);  
    glVertex2f(2.0, 1.0);  
    glVertex2f(2.0, 2.0);  
    glVertex2f(3.0, 2.0);  
glEnd();
```



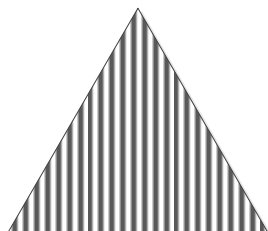
- b. Considere uma cena constituída por um labirinto e um personagem. Admita que a posição do centro do personagem é representada pelas variáveis $\{obj.x, obj.y, obj.z\}$, a orientação do personagem por $obj.dir$ e a altura por $CHARACTER_HEIGHT$. Sabendo que a distância da câmara ao personagem é representada por $CAMERA_DISTANCE$, configure a câmara de modo a obter uma vista de frente do personagem semelhante à ilustrada na figura.



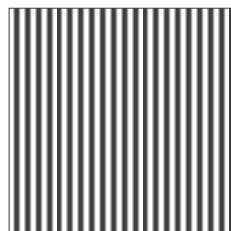
```
gluLookAt ( _____, _____, _____,
            _____, _____, _____,
            _____, _____, _____) ;
```

N.º _____ Nome _____

- c. Considere uma cena composta por um cubo de material $\{0.0, 0.25, 0.33\}$ e um cone de material $\{1.0, 0.0, 0.5\}$. Sabendo que existe apenas uma fonte de luz, como deveria configurar a sua componente difusa de modo a que apenas o cubo seja visível?
- $\{1.0, 1.0, 1.0\}$
 - $\{1.0, 0.0, 0.0\}$
 - $\{0.0, 1.0, 0.0\}$
 - $\{0.0, 0.0, 1.0\}$
- d. Suponha que pretende aplicar uma textura ao polígono ilustrado em a) com a imagem de textura representada em b). Complete o seguinte extracto de código.



a) polígono texturizado

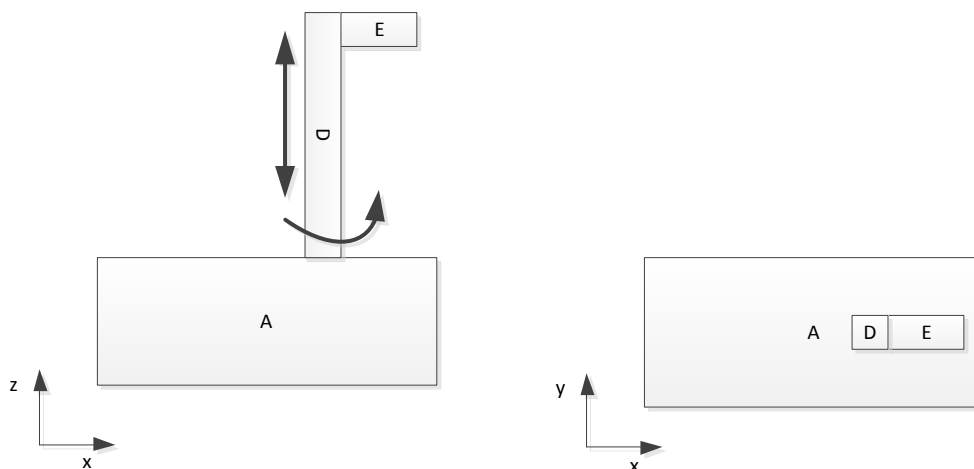


b) textura

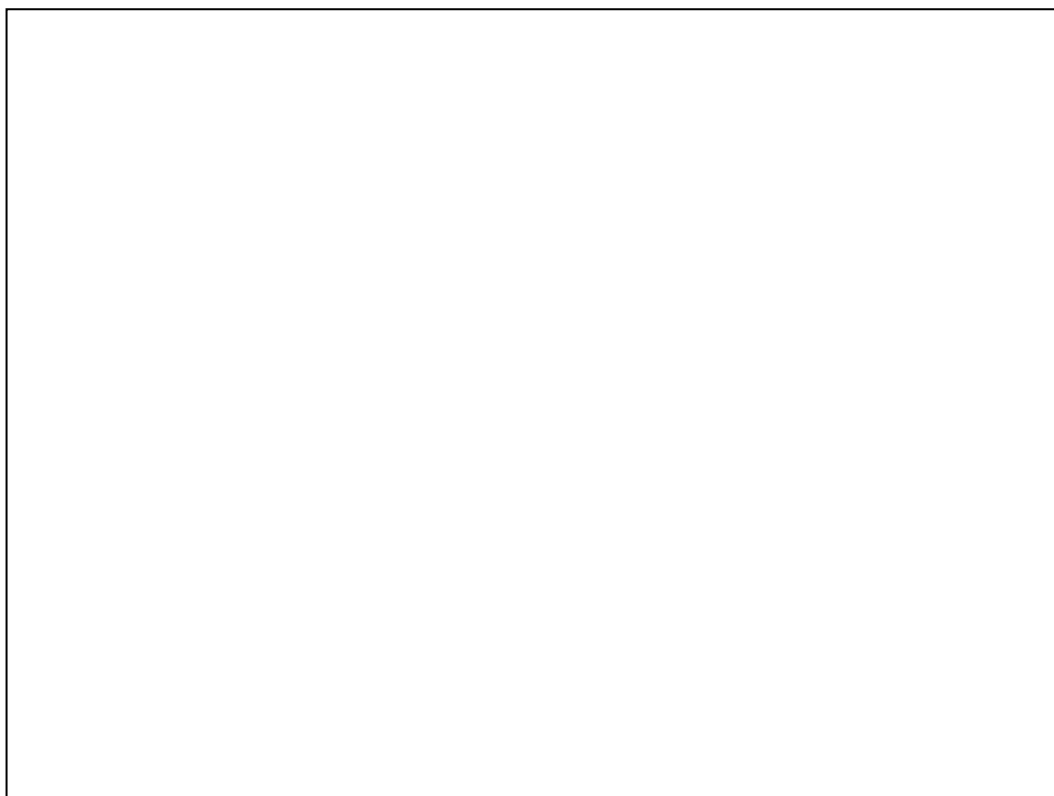
```
glBegin(GL_TRIANGLES);
    glTexCoord2f(_____, _____);
    glVertex3f(0.0, 0.0, 0.0);
    glTexCoord2f(_____, _____);
    glVertex3f(10.0, 0.0, 0.0);
    glTexCoord2f(_____, _____);
    glVertex3f(5.0, 10.0, 0.0);
glEnd();
```


N.º _____ Nome _____

- e. Pretende-se construir o objecto da figura seguinte (um modelo rudimentar de um submarino e respectivo periscópio) recorrendo à função `glutSolidCube()`. Os três elementos possuem dimensões distintas definidas pelas constantes L_n , A_n e C_n , em que n designa o elemento (A, D ou E). Pretende-se ainda que o periscópio (elementos D e E) seja articulado, podendo subir e descer, bem como rodar, como ilustrado nos alçados lateral e de topo da figura.



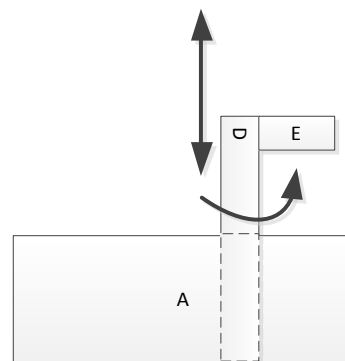
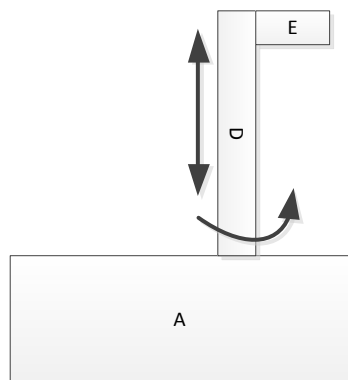
- i. Desenhe a árvore de cena correspondente.



-
- This image shows a blank sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.

N.º _____ Nome _____

- iii. Escreva o código que deve colocar no *timer* para animar o objecto anterior de modo a que o periscópio comece na posição inferior, vá subindo e, uma vez atingida a posição mais elevada, efectue uma rotação completa, após o que volta a descer até à posição original. As posições máxima e mínima do periscópio podem ser deduzidas da figura seguinte. Assuma a existência de uma variável inteira *modelo.periscopio.estado*, a qual deverá assumir os valores 1, 0 ou -1 consoante o periscópio se encontre a subir, a rodar ou a descer, respectivamente.



```
void Init()
{
    // 1: a subir
    // 0: a rodar
    // -1: a descer

    modelo.periscopio.estado = 1;
}

void Timer(int value)
{
    glutTimerFunc(estado.delay, Timer, 0);
```

}