

Parte Teórica e Teorico-Prática

60% – 45 minutos

Para cada uma das questões que se seguem assinale a resposta que entende ser a mais correcta.

1. O design de uma interface do tipo WIMP
 - a. não deve contemplar o uso de teclas aceleradoras, pois o mesmo implica um esforço de memorização considerável por parte do utilizador
 - b. deve ser centrado no sistema e não no utilizador, pois o comportamento deste último é imprevisível
 - c. não deve basear-se em analogias com o mundo real, pois pode suscitar confusão na mente do utilizador
 - ☒ d. nenhuma das anteriores
2. A utilização de códigos de cores em interfaces com o utilizador
 - a. deve ser evitada a todo o custo, por causa dos utilizadores daltónicos
 - ☒ b. deve ser redundante, de modo a ter em conta as limitações dos utilizadores daltónicos
 - c. pode ser usada livremente, desde que no *splash screen* da aplicação apareça um aviso a desaconselhar o seu uso por utilizadores daltónicos
 - d. nenhuma das anteriores
3. As mensagens de erro
 - ☒ a. devem ser fraseadas de forma positiva e construtiva
 - b. devem atribuir a culpa do erro ao utilizador
 - c. devem conter antropomorfismos
 - d. todas as anteriores
4. As representações gráficas matriciais
 - a. permitem efectuar operações como a rotação e a escala sem perda de precisão
 - b. são representações obsoletas, tendo sido substituídas pelas suas congéneres vectoriais
 - ☒ c. caracterizam-se por uma complexidade de processamento $O(n.^{\circ} \text{ de píxeis})$
 - d. todas as anteriores
5. A representação de vectores em coordenadas homogéneas
 - a. obriga a que o seu comprimento seja sempre unitário
 - b. distingue-se da dos pontos pelo facto de w ser diferente de zero
 - c. não é de todo possível, pois as coordenadas homogéneas só se aplicam aos pontos
 - ☒ d. nenhuma das anteriores
6. Na composição de transformações lineares
 - ☒ a. a ordem das transformações é importante
 - b. as dimensões e ângulos do objecto transformado são sempre preservadas
 - c. as translações precedem sempre as rotações
 - d. nenhuma das anteriores

7. Na codificação explícita de sólidos
- a. há uma lista de vértices e as faces referenciam os seus vértices através de ponteiros
 - b. há uma lista de arestas e as faces referenciam as suas arestas através de ponteiros
 - ☒ c. há muita redundância de informação
 - d. as adjacências são determinadas em tempo constante
8. Uma forma de determinar o vector normal a um triângulo genérico consiste em
- a. calcular o produto escalar dos vectores definidos pelos vértices do triângulo e multiplicar o valor assim obtido pelas coordenadas do ponto médio do triângulo
 - ☒ b. calcular o produto vectorial dos vectores definidos pelos vértices do triângulo e dividir o vector resultante pelo seu comprimento
 - c. o vector normal a um triângulo tem sempre as componentes (0, 0, 1)
 - d. não é possível, pois para calcular o vector normal são necessários pelo menos 4 vértices
9. O modelo de iluminação do OpenGL
- a. permite especificar fontes de luz do tipo projector
 - b. suporta efeitos atmosféricos como o nevoeiro e a atenuação
 - c. é calculado apenas nos vértices das superfícies
 - ☒ d. todas as anteriores
10. A componente especular do modelo de iluminação de Phong
- a. não depende da posição do observador
 - ☒ b. é característica dos materiais altamente polidos
 - c. não pode ser usada em fontes de luz omnidireccionais
 - d. todas as anteriores

Pergunta de desenvolvimento

Considere o modelo de uma figura humana (chamemos-lhe Homer). Suponha que a mesma se encontra na origem, orientada para o semi-eixo positivo dos xx . Considere também que a direcção “para cima” é a correspondente ao semi-eixo positivo dos zz . Sabendo que dispõe das transformações de translação (representada por $T(tx, ty, tz)$), de rotação (representada por $R(\text{ângulo}, \text{eixo})$) e de escala (representada por $S(sx, sy, sz)$), indique como procederia para:

- a. reduzir em 20% o perímetro abdominal do Homer, sem alterar a sua altura

$S(0.8, 0.8, 1)$

- b. orientar o Homer para o semi-eixo positivo dos yy

$R(90^\circ, zz)$

- c. deslocar o Homer de 3 unidades ao longo do eixo dos yy

$T(0, 3, 0)$

11. Para que serve a instrução `glFrustum`?

- ☒ a. para definir uma projecção em perspectiva
- b. faz um efeito de saturação das cores
- c. para definir uma projecção ortográfica
- d. faz um efeito de alta definição de cores

12. Em OpenGL as transformações possíveis de aplicar a um objecto são

- a. translações
- b. rotações
- c. escalas
- ☒ d. todas as anteriores

13. A instrução `glBindTexture` permite

- ☒ a. activar uma textura
- b. atribuir uma textura a um objecto
- c. carregar uma textura para memória
- d. definir propriedades da textura

14. O vector normal de uma superfície ou vértice

- a. é utilizado para cálculos de iluminação
- b. define a perpendicular à superfície ou vértice
- c. deve ser unitário
- ☒ d. todas as anteriores

15. Para obter um efeito de repetição de uma textura (3 vezes na horizontal e 1 vez na vertical) no seguinte objecto, quais as coordenadas de texturas que deveriam ser usadas?

```
glBegin(GL_QUAD);  
    glTexCoord(...); glVertex2f(1, 1);  
    glTexCoord(...); glVertex2f(3, 1);  
    glTexCoord(...); glVertex2f(3, 3);  
    glTexCoord(...); glVertex2f(1, 3);  
glEnd();
```

- a. (0, 0) (0, 1) (1, 1) (1, 0)
- b. (0, 0) (1, 0) (1, 1) (0, 1)
- ☒ c. (0, 0) (3, 0) (3, 1) (0, 1)
- d. (0, 0) (0, 3) (1, 3) (1, 0)

16. O OpenGL é

- a. dependente do sistema operativo
- b. dependente do hardware gráfico utilizado
- c. dependente do sistema gestor de janelas utilizado
- ☒ d. nenhuma das anteriores

17. A diferença entre a instrução `glVertex3f` e `glVertex3fv` é

- a. a primeira define um vértice e a segunda um vector de vértices
- ☒ b. a primeira define um vértice usando três argumentos e a segunda usando um vector de três posições
- c. a primeira define três vértices e a segunda define três vértices em formato vector
- d. a primeira define um vértice em coordenadas 3D e a segunda em coordenadas homogéneas

18. A instrução `glutPostRedisplay` deve ser usada para

- a. indicar ao OpenGL que terminámos de desenhar a cena
- ☒ b. indicar ao GLUT que pretendemos forçar o redesenho da cena
- c. indicar ao GLUT que vamos fazer alterações aos objectos após o desenho da cena
- d. exportar a cena para formato PostScript

19. O OpenGL suporta nativamente os seguintes tipos de objectos

- ☒ a. pontos, linhas, polígonos
- b. cilindros, esferas, cones, *torus* e chaleiras
- c. cubos
- d. todos os anteriores

20. A instrução `glLoadIdentity` aplicada à matriz de model/view coloca a câmara

- a. no infinito a “olhar” para a origem
- ☒ b. na origem a “olhar” para o semi-eixo negativo dos zz
- c. no infinito a “olhar” para o centro da janela
- d. na origem a “olhar” para o semi-eixo positivo dos zz