

## Sistemas Gráficos e Interacção

Época Normal

2023-02-01

N.º \_\_\_\_\_ Nome \_\_\_\_\_

**Duração da prova:** 45 minutos

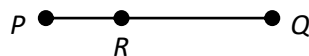
**Cotação de cada pergunta:** assinalada com parêntesis rectos

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

### Parte Teórica

10%

- a. **[3.3]** A visualização, no ecrã de um vulgar computador, de um gráfico descrito no formato SVG (*Scalable Vector Graphics*)
- i. Só é possível em sistemas cuja arquitectura contempla um processador gráfico (GPU)
  - ii. Requer técnicas sofisticadas de reconhecimento de padrões
  - iii. ☒ Requer a realização prévia de uma operação de rasterização
  - iv. Nenhuma das anteriores
- b. **[3.3]** Dados dois pontos distintos  $P$  e  $Q$  e a combinação linear afim  $R = \alpha P + (1 - \alpha)Q$ , qual o valor de  $\alpha$  para o qual o ponto  $R$  fica duas vezes mais próximo de  $P$  do que de  $Q$ ?



- i.  $\alpha = -0.33$
  - ii.  $\alpha = 0.33$
  - iii. ☒  $\alpha = 1 - 0.33$
  - iv. Nenhuma das anteriores
- c. **[3.3]** Numa projecção em perspectiva
- i. O volume de visualização tem a forma de um paralelepípedo
  - ii. ☒ O volume de visualização tem a forma de um tronco de pirâmide
  - iii. A dimensão aparente dos objectos visualizados diminui com a diminuição da distância à câmara
  - iv. Nenhuma das anteriores

- d. **[3.3]** Qual das seguintes técnicas de codificação de malhas poligonais permite desenhar eficientemente a malha sem que cada aresta seja desenhada duas vezes?
- i. Explícita
  - ii. Apontadores para uma lista de vértices
  - ☒ iii. Apontadores para uma lista de arestas
  - iv. Nenhuma das anteriores
- e. **[3.3]** O conhecimento do vector normal não é necessário ao cálculo
- ☒ i. Da componente ambiente de iluminação
  - ii. Da componente difusa de iluminação
  - iii. Da componente especular de iluminação
  - iv. Nenhuma das anteriores
- f. **[3.3]** Uma função de mapeamento de textura
- i. Devolve, para cada ponto do espaço de textura, o ponto correspondente da superfície do objecto
  - ii. Corresponde à forma com que a textura é usada para “embrulhar” (*wrap*) o objecto
  - iii. Pode basear-se na descrição paramétrica da superfície do objecto ao qual a textura está a ser aplicada
  - ☒ iv. Todas as anteriores

## Sistemas Gráficos e Interação

Época Normal

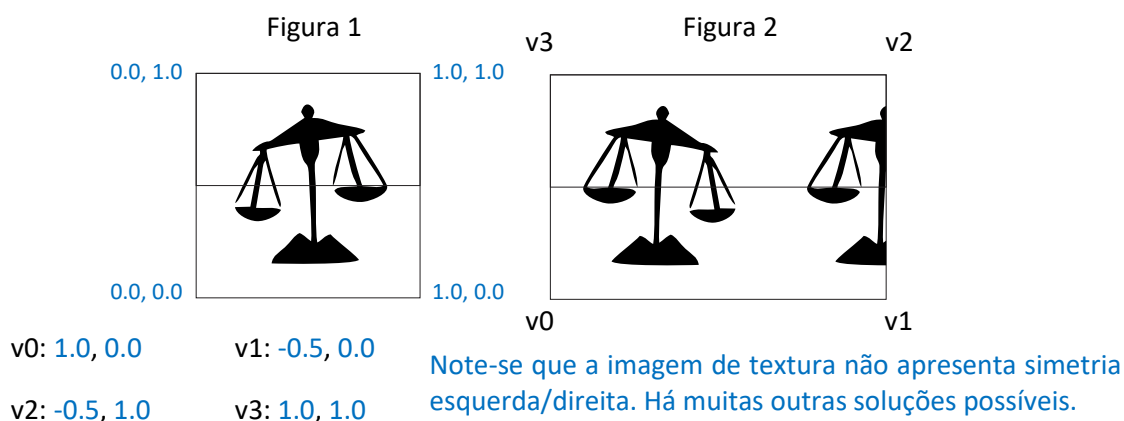
2023-02-01

N.º \_\_\_\_\_ Nome \_\_\_\_\_

### Parte Teórico-Prática

20%

- a. **[4.0]** Pretende-se mapear a textura representada na Figura 1 num rectângulo, de modo que este fique com o aspecto ilustrado na Figura 2. Indique as coordenadas de textura correspondentes a cada um dos vértices do polígono.



- b. **[2.0]** Considere um cubo constituído por um material cor-de-rosa (1.0, 0.5, 1.0), iluminado por uma única fonte de luz de cor azul-celeste (0.0, 0.5, 1.0). Quais as componentes primárias (R, G, B) da cor resultante? Indique os cálculos realizados.

$$R = 1.0 * 0.0 = 0.0 \quad G = 0.5 * 0.5 = 0.25 \quad B = 1.0 * 1.0 = 1.0$$

- c. **[1.4]** Um *Object3D* e um *Group* distinguem-se principalmente por o *Group*

- i. Permitir ter sub-objectos
- ii. Permitir incorporar *meshes*
- iii. Facilitar a leitura do código
- iv. Permitir integrar *materials*

- d. **[1.4]** Numa *OrthographicCamera*, a opção *far* é útil para

- i. Definir o campo de visão (*fov*)
- ii. Evitar desenhar objectos muito distantes da câmara
- iii. Controlar a resolução da cena
- iv. Activar sombras

- e. **[1.4]** A biblioteca *lil-gui* permite facilmente

- i. Alterar características de objectos
- ii. Informar o utilizador de valores alterados de características de objectos
- iii. Agrupar características de objectos em *folders*
- iv. Todas as anteriores

- f. **[1.4]** Especificar os vértices das faces de sólidos 3D sempre com a mesma sequência (horária ou anti-horária quando vista do exterior) permite
- i. Aumentar a rapidez da visualização
  - ii. Agrupar as faces mais facilmente
  - iii. Facilitar a integração de texturas
  - iv. Usar índices na identificação dos vértices
- g. **[1.4]** Para se obter sombras numa cena 3D tem de se activar esta funcionalidade
- i. Nas luzes
  - ii. Nos materiais e *renderer*
  - iii. Nas luzes, *meshes* e *renderer*
  - iv. Nas luzes, materiais e *renderer*
- h. **[1.4]** Um *Environment Map* é particularmente útil para simular
- i. Sombras
  - ii. Iluminação
  - iii. Reflexões
  - iv. Física
- i. **[1.4]** Para implementar rapidamente um sistema de controlo da vista do utilizador pode-se usar
- i. *OrbitControls*
  - ii. *CameraControls*
  - iii. *ViewControls*
  - iv. *GUIControls*
- j. **[1.4]** Um *RayCaster* é muito útil no three.js para permitir ao utilizador
- i. Obter visualizações mais realistas
  - ii. Mudar a iluminação na cena
  - iii. Mostrar informação de objectos 3D
  - iv. Interagir com objectos 3D na cena
- k. **[1.4]** O *RoughnessMap* permite
- i. Ajustar a altura dos vértices de uma *mesh*
  - ii. Ajustar a posição (x, y) dos vértices de uma *mesh*
  - iii. Definir áreas mais ou menos brilhantes/baças
  - iv. Definir áreas mais ou menos transparentes/opacas
- l. **[1.4]** A propriedade *fog* de um material é útil para o seguinte cenário
- i. Simular nevoeiro no fundo de uma floresta
  - ii. Quando se pretende definir densidades variáveis de nevoeiro
  - iii. Ajustar a influência da iluminação no nevoeiro
  - iv. Representar uma sala sem nevoeiro com vista para o exterior com nevoeiro