

## Sistemas Gráficos e Interação

Época Normal

2022-02-15

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] As representações gráficas matriciais

- i. São representações obsoletas, tendo sido substituídas pelas suas congéneres vectoriais
- ii. Caracterizam-se por uma complexidade de processamento  $O(n^2)$  de vértices / vectores
- iii. Permitem efectuar operações como a rotação e a escala sem perda de precisão
- iv. Nenhuma das anteriores

b. [3.3] Considere dois pontos genéricos P e Q (não coincidentes) e o ponto R resultante da combinação linear afim  $R = (1 - \alpha) * P + \alpha * Q$

- i. Se  $\alpha = 0$ , então o ponto R coincide com o ponto Q
- ii. Se  $\alpha = 1$ , então o ponto R coincide com o ponto P
- iii. Se  $\alpha = \frac{1}{2}$ , então o ponto R coincide com o ponto médio do segmento PQ
- iv. Se  $0 < \alpha < 1$ , então o ponto R não pertence ao segmento PQ

c. [3.3] Numa projecção ortográfica

- i. O volume de visualização tem a forma de um tronco de pirâmide
- ii. O volume de visualização tem a forma de um tronco de cone
- iii. A dimensão aparente dos objectos visualizados diminui com o aumento da distância à câmara
- iv. Nenhuma das anteriores

- d. **[3.3]** Na codificação de sólidos com base em ponteiros para uma lista de vértices
- i. Cada face do sólido armazena explicitamente a lista ordenada das coordenadas dos seus vértices
  - ii. Há uma lista de vértices e as faces referenciam os seus vértices através de apontadores para essa lista
  - iii. A redundância de informação é maior do que na codificação explícita
  - iv. Nenhuma das anteriores
- e. **[3.3]** Os holofotes (*spotlights*) constituem um exemplo de uma fonte de luz
- i. Posicional
  - ii. Direccional
  - iii. Omnidireccional
  - iv. Nenhuma das anteriores
- f. **[3.3]** No mapeamento de texturas em three.js, o processo de filtragem designado por trilinear
- i. Escolhe o texel que mais se aproxima do centro do píxel no mipmap que melhor se adequa ao contexto de minificação existente
  - ii. Calcula uma média pesada da matriz de 2 x 2 texels que mais se aproxima do centro do píxel no mipmap que melhor se adequa ao contexto de minificação existente
  - iii. Escolhe o texel que mais se aproxima do centro do píxel em cada um dos dois mipmaps que melhor se adequam ao contexto de minificação existente; em seguida, efectua uma interpolação linear destes dois valores
  - iv. Calcula uma média pesada da matriz de 2 x 2 texels que mais se aproxima do centro do píxel em cada um dos dois mipmaps que melhor se adequam ao contexto de minificação existente; em seguida efectua uma interpolação linear destes dois valores

## Sistemas Gráficos e Interação

Época Normal

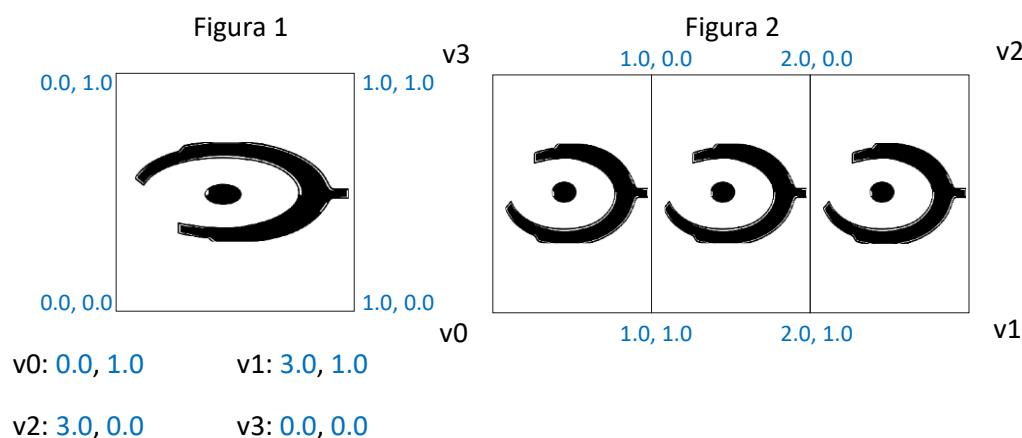
2022-02-15

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 uma esfera constituída por um material cor-de-laranja (1.0, 0.5, 0.0), iluminada por uma única fonte de luz de cor verde-azulada (0.0, 0.8, 1.0). Quais as componentes primárias (R, G, B) da cor resultante? Indique os cálculos realizados.

$$R = 1.0 * 0.0 = 0.0$$

$$G = 0.5 * 0.8 = 0.4$$

$$B = 0.0 * 1.0 = 0.0$$

- c. **[1.4]** Um objecto *Mesh* combina

- i. *Geometry e Material*
- ii. *Geometry e Camera*
- iii. *Material e Scene*
- iv. *Light e Scene*

- d. **[1.4]** Numa *OrthographicCamera*, o factor de ampliação (*zoom*) pode ser ajustado com

- i. *Campo de visão (fov)*
- ii. *Largura e Altura*
- iii. *Aspect ratio*
- iv. *Far*

- e. **[1.4]** Se se usar índices para definir uma geometria, diminui o número de

- i. *Vértices*
- ii. *Arestas*
- iii. *Materiais*
- iv. *Faces*

- f. [1.4] Ao usar uma *DirectionalLight* na posição (0.0, 0.0, 0.0), orientada no sentido positivo do eixo Z, vão ser iluminados
- i. Apenas os objectos acima ( $z > 0.0$ ) do plano OXY
  - ii. Apenas os objectos abaixo ( $z < 0.0$ ) do plano OXY
  - iii. Apenas os objectos que incluam a origem (0.0, 0.0, 0.0)
  - ☒ iv. Todos os objectos
- g. [1.4] Quando se usa uma *SpotLight*, obtém-se um *overhead* computacional de  $n$  renderizações para calcular as sombras, sendo  $n = ?$
- ☒ i. 1
  - ii. 2
  - iii. 4
  - iv. 6
- h. [1.4] Numa *AmbientLight* define-se, normalmente
- i. Cor e Alcance
  - ☒ ii. Cor e Intensidade
  - iii. Cor e Posição
  - iv. Posição e Direcção
- i. [1.4] Para se obter uma renderização mais rápida de texturas deve-se usar
- i. *LinearFilter*
  - ☒ ii. *NearestFilter*
  - iii. *ExtralelinearFilter*
  - iv. *CubicFilter*
- j. [1.4] Para implementar *picking* no three.js, a opção mais usada consiste em usar
- i. *ClickEngine*
  - ii. *PickingMode*
  - iii. *MouseXtra*
  - ☒ iv. *RayCaster*
- k. [1.4] O *FogExp2* tem uma resposta, quando comparada com a do *Fog*
- ☒ i. Mais realista do ponto de vista da física
  - ii. Computacionalmente mais complexa
  - iii. Mais simples de ajustar
  - iv. Equivalente em termos de resultado final
- l. [1.4] A técnica de pós-processamento permite implementar facilmente
- ☒ i. O aspecto de um filme a preto e branco
  - ii. Superfícies com maior detalhe
  - iii. Uma renderização mais eficiente
  - iv. A optimização de utilização de memória RAM