

COMPUTAÇÃO GRÁFICA 7937-30_43701_R_E1_20241

CONTEÚDO

Revisar envio do teste: QUESTIONÁRIO UNIDADE I

Usuário

Curso

COMPUTAÇÃO GRÁFICA

Teste

QUESTIONÁRIO UNIDADE I

Iniciado

Enviado

Status

Completada

Resultado da
tentativa

Tempo decorrido

Resultados exibidos Todas as respostas, Respostas enviadas, Respostas corretas, Comentários, Perguntas respondidas incorretamente

Pergunta 1

0,5 em 0,5 pontos



Dois termos muito empregados em computação gráfica e processamento de imagens são rasterização (*rastering*) e renderização (*rendering*). O termo rasterização pode ser entendido como:

Resposta
Selecionada:

✓ a.

O processo de conversão da representação vetorial para a matricial (*pixels*).

Respostas:

✓ a.

O processo de conversão da representação vetorial para a matricial (*pixels*).

b.

Uma transformação matemática para converter modelos matemáticos em imagens.

c. O processo de sensibilizar a superfície de um dispositivo *raster*.d. Uma transformação ponto-a-ponto de translação de *pixels*.

e.

Um outro nome para o algoritmo de Bresenham para retas e circunferências.

Comentário da
resposta:

Resposta: A

Comentário: rasterização é a tarefa de tomar uma imagem descrita vetorialmente, no espaço vetorial R^2 ou R^3 , e convertê-la em uma imagem *raster* (matriz de *pixel*) para a saída em vídeo ou impressora. Renderização é o processo pelo qual se pode obter o produto de um processamento digital qualquer.

Pergunta 2

0,5 em 0,5 pontos



Modelo que se define como um conjunto de células localizadas em coordenadas contínuas, implementadas numa matriz 2D. Cada célula, também chamada elemento da imagem, elemento da matriz ou *pixel*, é referida por indexes de linha e coluna. Como vantagens desse modelo, pode-se dizer que os dados possuem uma estrutura simples e operações de superposição que são implementadas de maneira fácil e eficaz. Indique qual das opções essa definição se refere.

Resposta Seleccionada: ☒ c. Modelo *raster*.

- Respostas:
- a. Modelo vetorial.
 - b. Modelo de câmera.
 - ☒ c. Modelo *raster*.
 - d. Modelo geométrico.
 - e. Modelo gaussiano.

Comentário da resposta: Resposta: C
Comentário: células localizadas em coordenadas contínuas se referem ao espaço vetorial R^2 ou o Sistema de Referência do Universo (SRU). Se essas células serão implementadas em uma matriz de *pixel* ou no Sistema de Referência do Dispositivo (SRD), estamos nos referindo à rasterização de imagens. Logo, a definição se refere a um modelo *raster*.

Pergunta 3

0,5 em 0,5 pontos



Na representação de um objeto é necessário utilizar sistemas de referência. Qual o sistema de referência que trabalha com valores entre 0 e 1 e torna a geração de imagens independente do dispositivo?

Resposta Seleccionada: ☒ d. Sistema de Referência Normalizado (SRN).

- Respostas:
- a. Sistema de Referência do Universo (SRU).
 - b. Sistema de Referência do Objeto (SRO).
 - c. Sistema de Referência do Mundo (SRM).
 - ☒ d. Sistema de Referência Normalizado (SRN).
 - e. Sistema de Referência Axial (SRA).

Comentário da resposta: Resposta: D
Comentário: SRN: é um sistema que trabalha com as coordenadas normalizadas, isto é, com valores entre 0 e 1, em que $0 \leq x \leq 1$ e $0 \leq y \leq 1$, sendo x e y as

coordenadas horizontais e verticais. O SRN serve como um sistema de referência intermediário.

SRO: é um sistema de referência local utilizado para definir a posição e a orientação de um objeto em relação ao Sistema de Referência Universo. Cada objeto pode ter seu próprio sistema de referência objeto, que é usado para realizar transformações geométricas em relação a outros objetos na cena.

SRM: é o mesmo que o SRU, portanto, é um sistema de referência global que é utilizado para definir a posição dos objetos em relação a um ponto fixo no espaço, geralmente, o ponto de origem do sistema de coordenadas cartesianas. Esse sistema de referência é importante para definir a posição inicial dos objetos em uma cena.

SRA: não existe esse sistema em computação gráfica.

Pergunta 4

0,5 em 0,5 pontos



(Adaptado – POSCOMP 2002) Considere uma cena representada no sistema de referência do universo (SRU), uma *window* definida pelo par de coordenadas (0,0)-(100,100) e uma *viewport* definida pelo par de coordenadas (20,30)-(300,100). Considere ainda que as coordenadas que definem *window* e *viewport* correspondem, respectivamente, aos limites inferior esquerdo e superior direito de ambas. Analise as afirmativas levando em consideração os conceitos clássicos de *window* e *viewport* e assinale a alternativa correta.

I. *Window* e *viewport* estão definidas no SRU.

II. No processo de mapeamento dessa *window* para essa *viewport* não haverá modificação na relação de aspecto.

III. O mapeamento da *window* redefinida pelo par de coordenadas (0,0) – (50,50) para a mesma *viewport* (20,30)-(300,100) corresponde a uma operação de *zoom in* sobre o mesmo universo (nota: *zoom in* = aproximar / *zoom out* = afastar).

Resposta Seleccionada: ☒ c. III é verdadeira.

Respostas:

a. I e II são verdadeiras.

b. I e III são falsas.

☒ c. III é verdadeira.

d. II e III são verdadeiras.

e. Todas são falsas.

Comentário da resposta:

Resposta: C

Comentário:

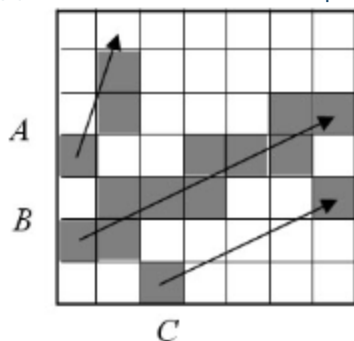
Afirmativa I – falsa, pois a *viewport* é definida no SRD.

Afirmativa II – falsa, todo processo de rasterização implica em modificação no aspecto da imagem, pois a *window* é representada por coordenadas contínuas e a *viewport* por coordenadas discretas em uma matriz de *pixel*.

Afirmativa III – verdadeira, a *viewport* é maior que a *window*. Logo, a imagem rasterizada se apresentará aumentada, correspondendo à operação de *zoom in*.



A figura representa uma *viewport* de 7 x 7 *pixels*, em que os *pixels* são representados por quadrados. Nela foram riscados três segmentos de reta, na forma de vetores, rotulados como A, B e C. Para cada um, foram acesos *pixels* que representariam a rasterização desses segmentos. Contudo, de acordo com os critérios utilizados para construir o algoritmo de Bresenham para o traçado de linhas, todas estão incorretamente representadas. A seguir estão listados 4 erros mais comuns e 5 afirmativas sobre em quais segmentos eles estão presentes. Julgue as afirmações como verdadeiras (V) ou falsas (F) e assinale a alternativa que contém a correspondência correta.



Erro 1: A linha não começa ou termina nos pontos extremos do segmento.

Erro 2: Há *pixels* acesos que não são os mais próximos do segmento.

Erro 3: Há dois *pixels* acesos indevidamente em uma mesma linha/coluna.

Erro 4: Há *pixels* que deveriam ser acesos e não foram, deixando falhas na representação do segmento.

- () O segmento A apresenta os erros 1 e 2.
- () O segmento B apresenta o erro 3.
- () O segmento C apresenta o erro 4.
- () O segmento A é incremental em y e não pode ter dois *pixels* na mesma linha.
- () Os segmentos B e C são incrementais em x e não podem ter dois *pixels* na mesma coluna.

Resposta Seleccionada: ☒ a. V, V, V, V, V.

Respostas: ☒ a. V, V, V, V, V.

b. F, F, F, F, F.

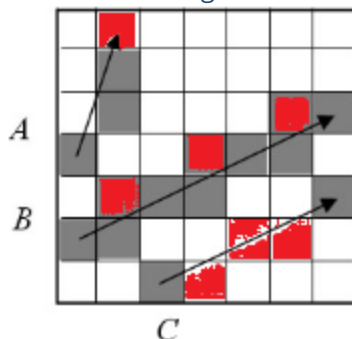
c. V, F, V, F, V.

d. F, V, F, V, F.

e. V, V, F, F, F.

Comentário da resposta: Resposta: A

Comentário: na figura estão destacados, em vermelho, os erros citados.



Pergunta 6

0,5 em 0,5 pontos



O primeiro passo do algoritmo de Bresenham para traçado de circunferência envolve plotar o primeiro ponto (0, r) no início de cada um dos quatro quadrantes. Considerando que (xc, yc) são as coordenadas do centro no SRD, assinale a alternativa que executa essa tarefa:

Resposta Seleccionada: ☒ d. PLOTA(xc, yc+r); PLOTA(xc-r, yc); PLOTA(xc, yc-r); PLOTA(xc+r, yc).

Respostas:

a. PLOTA(0, r); PLOTA(-r, 0); PLOTA(0, -r); PLOTA(r, 0).

b. PLOTA(xc, yc+r); PLOTA(xc-r, yc); PLOTA(xc-r, yc-r); PLOTA(xc+r, yc+r).

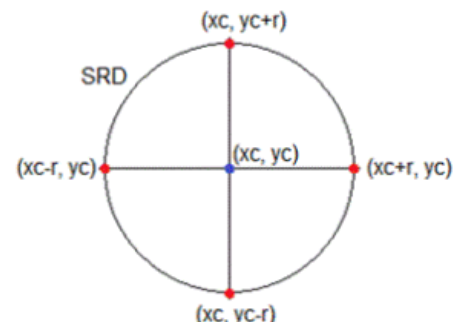
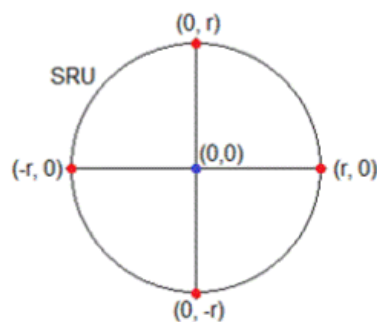
c. PLOTA(xc, yc+r); PLOTA(yc-r, xc); PLOTA(xc, yc-r); PLOTA(yc+r, xc).

☒ d. PLOTA(xc, yc+r); PLOTA(xc-r, yc); PLOTA(xc, yc-r); PLOTA(xc+r, yc).

e. PLOTA(0, yc+r); PLOTA(xc-r, 0); PLOTA(0, yc-r); PLOTA(xc+r, 0).

Comentário da resposta:

Resposta: D
Comentário:



Observando a figura e fazendo a correspondência das coordenadas do SRU para o SRD, obtemos a sequência de pontos descritas na alternativa D.

Pergunta 7

0,5 em 0,5 pontos



(Enade) Avalie as afirmativas a seguir e assinale a alternativa correta.

I. Um terminal *raster* apresentará o efeito "pisca-pisca" quando a cena for muito complexa.

II. No mapeamento *window-to-viewport*, mantendo-se a *viewport* fixa e aumentando-se o tamanho da *window*, provoca-se o efeito de *zoom-in*.

III. No algoritmo do ponto médio para traçado de círculos, se $f(x_M, y_M) = r^2 - x^2 - y^2 < 0$, o ponto (x_M, y_M) é interior à circunferência.

Resposta Seleccionada: ☒ e. III é correta.

Respostas:

a. Todas as afirmativas são corretas.

b. Todas as afirmativas são incorretas.

c. II é correta.

d. III é incorreta.

☒ e. III é correta.

Comentário Resposta: E

da resposta: Comentário: essa é uma questão típica de Enade, em que se examinam as competências sobre mais de um tema. A afirmativa I está associada tanto ao número de primitivas em uma cena, o que aumenta sua complexidade, quanto à frequência de resposta do *hardware* a um trabalho específico. Nesse caso, devemos lembrar que a formação de uma imagem na tela é realizada pela varredura de feixes de elétrons na matriz retangular de *pixels*. Geralmente, os monitores de vídeo utilizam o modo não entrelaçado, por meio do qual é feita a varredura de todas as linhas do dispositivo de uma só vez, o que requer altas frequências de paginação. Para evitá-las, recorre-se a uma técnica designada por entrelaçamento. No monitor em modo entrelaçado, a varredura é feita em duas fases, em que primeiro são percorridas as linhas ímpares e depois as pares. No entanto, essa técnica implica em um efeito de cintilação, "pisca-pisca" ou *flicker*. Vemos logo que tal efeito decorre da velocidade do *hardware* e sua resolução, nada tendo a ver com a complexidade da cena ou o número de primitivas a serem rasterizadas. Assim, a afirmativa I é falsa. A afirmativa II também é falsa, pois, aumentando-se a *window*, a figura representada em seu interior fica proporcionalmente menor. Ao mapearmos o conteúdo da *window* para a *viewport*, esse parecerá menor, o que corresponde a um efeito de *zoom-out* ou afastamento. Apenas a afirmativa III está correta. A inequação que aparece nessa afirmativa deriva da equação cartesiana da circunferência centrada no ponto (0,0), ou seja: $x^2 + y^2 = r^2 \Rightarrow r^2 - x^2$

- $y^2 = 0$ corresponde aos valores de x e y sobre a circunferência. Quando $r^2 - x^2 - y^2 > 0$, os pontos x e y estão fora e quando $r^2 - x^2 - y^2 < 0$, os pontos estão dentro da circunferência.

Pergunta 8

0,5 em 0,5 pontos



Presente em diversos segmentos da atividade humana, como nas artes, medicina, arquitetura, segurança pública, propaganda, processamento de dados, lazer, educação, entre outras, a Computação Gráfica (CG) é a área da Ciência da Computação que mobiliza os recursos matemáticos para gerar, manipular e interpretar imagens com a ajuda do computador. Praticamente tudo que poderíamos realizar usando imagens por meios tradicionais, pode ser tratado mais rápida e eficientemente com a CG. Uma vez entendida a real natureza dessa verdadeira caixa de ferramentas tecnológica é **incorreto** afirmar que:

Resposta ☒ e.

Selecionada: Computação Gráfica é uma disciplina do curso de Ciência da Computação ou Sistemas de Informação em que se aprende a desenhar e manipular imagens com *softwares* apropriados.

Respostas: a.

Computação Gráfica é a área da Ciência da Computação que trata da geração, manipulação e interpretação da informação, de maneira visual, utilizando o computador.

b.

Computação Gráfica é a área da Ciência da Computação que estuda representações visuais a partir de especificações geométricas de seus componentes.

c.

Computação Gráfica é a área da Ciência da Computação que trata da imagem em sua forma digital, suas transformações e melhoramentos.

d.

Computação Gráfica é a área da Ciência da Computação que cuida da especificação dos componentes da imagem a partir de sua representação visual.

✓ e.

Computação Gráfica é uma disciplina do curso de Ciência da Computação ou Sistemas de Informação em que se aprende a desenhar e manipular imagens com *softwares* apropriados.

Comentário Resposta: E

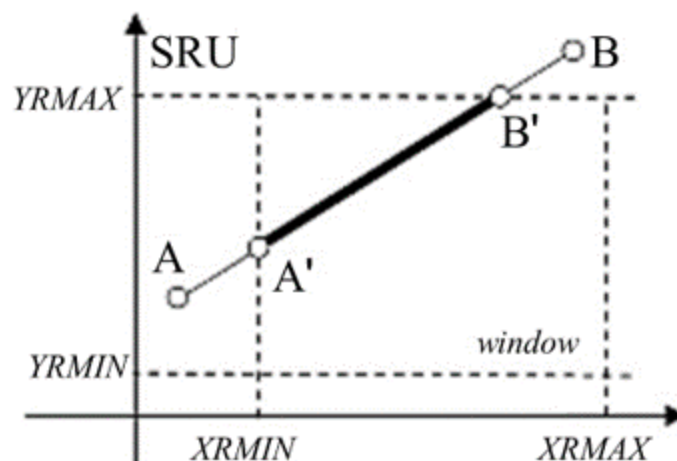
da resposta: Comentário: embora possa haver cursos em que seja fundamental aprender a usar programas de manipulação gráfica, como cursos de nível técnico, Comunicação Visual e *Design* Gráfico, em que a CG não é caracterizada como um fim em si; nos cursos de Ciência da Computação, Sistemas de Informação e Engenharia da Computação, em que o objetivo é desenvolver os aplicativos que irão servir para manipulação gráfica ou desenvolver pesquisa científica e tecnológica para seu avanço como área da Ciência da Computação, a disciplina de CG deve enfatizar o uso da lógica de programação e algoritmos mobilizando todos os recursos matemáticos necessários para sua realização.

Pergunta 9

0,5 em 0,5 pontos



Quando traçamos a aresta (segmento de reta) entre dois vértices (pontos) A e B, apenas o segmento A'B' que se encontra no interior da *window* deve ser mapeado para a *viewport* (o recorte, ou *clipping*, na figura a seguir). Pense no caso mais simples em que só o vértice A encontra-se no exterior da *window* (ou seja, o segmento de reta representável é o A'B). Assinale a alternativa que contém as coordenadas corretas do ponto A' no SRU para o caso em que $A = (0.5, 1.5)$, $B = (2.5, 2.5)$ e uma *window* limitada por $(1.0, 1.0) - (3.0, 3.0)$.



Resposta Selecionada: ☒ a. $A' = (1,0; 1,625)$

Respostas: ☒ a. $A' = (1,0; 1,625)$

b. $A' = (1,625; 1,0)$

c. $A' = (1,236; 1,625)$

d. $A' = (1,0; -1,625)$

e. $A' = (1,625; 1,236)$

Comentário da Resposta: A

resposta: Comentário: para calcularmos o valor de A' , primeiro devemos calcular a equação da reta AB. Assim:

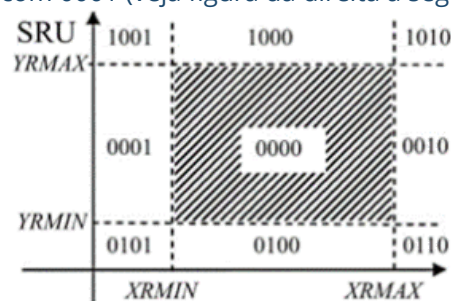
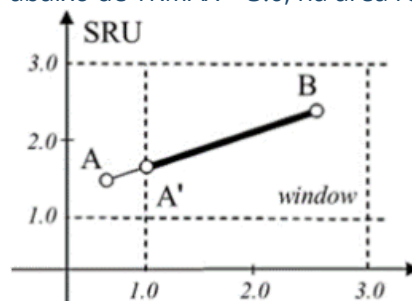
Fazendo $X_{Ri} = 0,5$, $Y_{Ri} = 1,5$, $X_{Rf} = 2,5$ e $Y_{Rf} = 2,5$, temos:

$$m = \frac{Y_{Rf} - Y_{Ri}}{X_{Rf} - X_{Ri}} = \frac{2,5 - 1,5}{2,5 - 0,5} = 0,25$$

$$b = Y_{Ri} - m \cdot X_{Ri} = 1,5 - 0,25 \cdot 0,5 = 1,375$$

$$y = 0.25x + 1.375$$

Depois precisamos identificar em que lugar o ponto A' aparece nas bordas da *window*. Essa é a parte difícil do algoritmo, mas vamos fazê-lo visualmente. Note que o ponto $A = (0.5, 1.5)$ está à esquerda e acima de $(X_{RMIN}, Y_{RMIN}) = (1.0, 1.0)$ e abaixo de $Y_{RMAX} = 3.0$, na área rotulada com 0001 (veja figura da direita a seguir).



Ou seja, A' está sobre a reta vertical $x = X_{RMIN} = 1.0$

Sabendo isso, podemos calcular o valor da coordenada y do ponto A' pela equação da reta fazendo $x = 1.0$. Assim:

$$y = 0.25 \cdot 1.0 + 1.375 = 1.625$$

Logo: $A' = (1.0, 1.625)$

Nota: o que acabamos de fazer é parte do que chamamos de *clipping*. Para desenharmos o segmento de reta $A'B$ na *viewport*, devemos calcular ainda os valores das coordenadas dos *pixels*

correspondente aos pontos A' e B na *viewport* e uni-los usando o algoritmo DDA ou Bresenham.



Considere uma função $PLOTA(X,Y)$ que acende um *pixel* na posição (X,Y) da tela. Considere, ainda, o *pixel* de coordenadas $(0, 0)$ situado no canto inferior esquerdo da tela (coordenadas em um sistema lógico). Com o algoritmo do ponto médio (Bresenham), obtivemos, para o segundo octante, o ponto $(300, 400)$. Como a circunferência que queremos rasterizar está centrada no *pixel* $(700,100)$, a chamada para a função que irá acender, por simetria, o *pixel* no quarto octante é:

Resposta Selecionada: ☒ d. $PLOTA(300,400)$

Respostas:

- ☐ a. $PLOTA(400,300)$
- ☐ b. $PLOTA(700,100)$
- ☐ c. $PLOTA(400,100)$
- ☒ d. $PLOTA(300,400)$
- ☐ e. $PLOTA(700,300)$

Comentário Resposta: D

da resposta: Comentário: trata-se de um problema no domínio da *viewport*, não sendo necessária qualquer conversão entre o SRU e o SRD. Como vimos, o algoritmo de Bresenham para circunferências calcula os valores de *pixel* apenas para o segundo octante e considerando a circunferência centrada na origem $(0,0)$. Uma vez calculado o ponto, ao plotarmos o resultado na *viewport*, devemos deslocá-lo relativamente ao centro da circunferência estabelecido pelo usuário. Isso se faz somando-se aos valores de (x, y) os valores de (x_c, y_c) do centro da circunferência.

← OK