

UNIVERSIDADE FEDERAL DE SÃO CARLOS – UFSCar  
Centro de Ciências Exatas e Tecnologia – CCET  
Departamento de Computação – DC

Computação Gráfica e Multimídia (2015)

Prof. Dr. Murillo Rodrigo Petrucelli Homem

Com a finalidade de padronizar a construção de aplicativos que utilizam recursos gráficos, surge a ideia do desenvolvimento dos chamados *Sistemas Gráficos* (SG). Dentre as principais características desejáveis para um SG podemos mencionar:

- *Minimalidade*: um SG deve implementar uma API (*Application Programmer Interface*) mínima, mas funcional;
- *Portabilidade*: um SG deve ser independente do dispositivo;
- *Separabilidade*: a arquitetura do SG deve isolar os detalhes de implementação da API gráfica;
- *Disponibilidade*: deve suportar a maioria das plataformas existentes.

Considerando as características elencadas acima, passaremos a implementar um conjunto de funções (em linguagem C) para construir um pacote gráfico simplificado, mas funcional, capaz de gerar (sintetizar) objetos bidimensionais (2D) e tridimensionais (3D) que serão visualizados em um dispositivo gráfico de saída matricial (por exemplo, um monitor de vídeo).

Algumas questões teóricas...

- 1) O que se entende por dispositivos gráficos?
- 2) Discuta as principais diferenças entre dispositivos gráficos vetoriais e dispositivos gráficos matriciais; considere os casos de dispositivos de entrada e de saída.
- 3) Qual é considerada a melhor combinação de dispositivos gráficos vetoriais e matriciais levando em conta as tecnologias atuais?
- 4) O que é um Sistema de Referência (SR)? Qual sua função? Discuta os três tipos de SR estudados e discutidos em sala de aula.
- 5) Quais são as expressões matemáticas para conversão entre os SR mencionados na questão anterior?
- 6) O que se entende por janela de visualização (ou retângulo de visualização)? O que se entende por “view-port”? Discuta os conceitos considerando os Sistema de Referência.
- 7) Faça uma breve pesquisa e discuta as diferenças entre os conceitos de “vector graphics” e “raster graphics”.
- 8) Discuta o conceito de rasterização. Faça um paralelo com as questões de números 3 e 7. Não esqueça de considerar também os SR.

- 9) Por que os algoritmos de conversão de primitivas gráficas vetoriais para dispositivos matriciais são importantes na área de Computação Gráfica?
- 10) Quais as características desejáveis em um bom algoritmo para o traçado de linhas (segmentos de retas) em Computação Gráfica?
- 11) Discuta a abordagem do algoritmo de *Bresenham*, para traçado de linhas retas.
- 12) O algoritmo de *Bresenham* é capaz de traçar linhas verticais? Em caso negativo, que modificações você poderia incorporar no algoritmo para que este possa desenhar segmentos de retas na vertical?
- 13) Discuta o funcionamento do algoritmo de *Bresenham* analisando o valor da variável que armazena o erro.
- 14) Quais são as duas etapas envolvidas no processo de preenchimento de polígonos de formas arbitrárias?

### Implementações...

Nos exercícios que pedem a visualização de algum objeto, utilize a função (ou uma modificação dela), baseada no *X-Window*, disponibilizada no *Moodle*. Ainda, se desejarem, podem utilizar as estruturas de dados simplificadas também disponibilizadas.

- 15) Implemente uma função para converter dados entre um SRU (ou do mundo) e um SRN (chamada de *transformação de visualização*). Que parâmetros devem ser passados para a função? O que deve retornar? Aproveite para refletir sobre o que chamamos em aula de “tamanho do mundo”!
- 16) Implemente uma função para converter dados entre um SRN e um SRD. Que parâmetros devem ser passados para a função? O que deve retornar?
- 17) Considere um SRU para o plano definido pelo usuário e um SRD descrito por uma grade retangular finita de tamanho fornecido pelo usuário e implemente uma função para desenhar um segmento de reta, em um dispositivo gráfico de saída, dados os pontos extremos  $p_1 = (x_1, y_1)$  e  $p_2 = (x_2, y_2)$  do segmento, sendo  $x_1, y_1, x_2$  e  $y_2$  variáveis no SRU. Nesse exercício, não se espera que seja implementada a abordagem do algoritmo de *Bresenham*. Por exemplo, considere a equação afim ou do primeiro grau para sua implementação, isto é,  $f(x) = \mathbf{a}x + \mathbf{b}$ , onde  $\mathbf{a}$  e  $\mathbf{b}$  são os coeficientes angular e linear, respectivamente,  $x$  um número real e o par ordenado  $(x, y = f(x))$  as coordenadas de um ponto no plano.
- 18) No exercício acima, discuta os casos particulares quando temos segmentos de reta horizontais ou verticais.
- 19) Uma vez implementada a função proposta no exercício anterior, considere algumas abordagens para determinar as coordenadas inteiras dos pontos que pertencem ao segmento de reta no dispositivo gráfico considerado: i) adote o critério de selecionar *as entradas* imediatamente acima e abaixo do ponto de intersecção do segmento com cada vertical na grade de pontos; ii) selecione *as entradas* cujas coordenadas são obtidas arredondando-se os valores das coordenadas de algum ponto do segmento; iii) selecione em cada vertical *a entrada* mais próxima do ponto de intersecção ao do segmento com a reta vertical; iv) selecione em cada vertical *a entrada* mais próxima do ponto de intersecção ao do segmento com a reta horizontal.

- 20) Justifique a “aparência” dos “segmentos de reta” obtidos por cada uma das abordagens no exercício anterior.
- 21) Implemente o algoritmo do ponto médio (*Bresenham*) para traçado de segmentos de retas. Considere os casos particulares, como, por exemplo, linhas verticais.
- 22) Implemente um algoritmo que, dados uma sequência de pontos no plano, ligue esses pontos (no SR adequado!), na ordem em que são apresentados, para formar um polígono regular. Que parâmetros devem ser passados para a função? O que esta deve retornar? Reflita com cuidado sobre essas questões.
- 23) Dados dois pontos arbitrários no SRU, visualize no monitor o “segmento de reta” que une esses pontos (reflita sobre a janela de visualização adequada e sobre o número de entradas do monitor virtual para esse exercício).
- 24) Dados dois pontos arbitrários no SRU, considere uma matriz de 640x480 entradas que deverá ser mapeada no dispositivo gráfico de saída (tela do monitor virtual). Divida essa matriz em quatro regiões distintas de mesmo tamanho (320x240 entradas). Cada uma dessas regiões será uma *viewport* no dispositivo de saída. Faça a exibição simultânea (visualize) do traçado do segmento de reta entre os pontos dados nas quatro *viewports*.

OBSERVAÇÃO: Do exercício 25 ao 32, considere uma janela de visualização que permita incorporar todos os objetos.

- 25) Considere um retângulo definido por *quatro coordenadas inteiras* em um espaço de coordenadas contínuas bidimensional. Implemente uma função para preencher esse retângulo com uma cor pré-definida pelo usuário, considerando uma grade retangular onde cada entrada desta grade corresponda a um pixel no espaço discreto de visualização.
- 26) Considere um retângulo definido por *quatro coordenadas reais* (incluindo coordenadas negativas). Implemente uma função para preencher este retângulo considerando uma grade retangular onde cada entrada desta grade corresponda a um pixel no espaço discreto de visualização.
- 27) No exercício anterior, qual foi o critério que você adotou para preencher os *pixels* da grade discreta nos quais existe a intersecção com as arestas do retângulo?
- 28) Implemente uma função para preencher um retângulo considerando apenas os *pixels* da grade discreta que não interceptam as arestas do retângulo, ou seja, considere apenas os pontos estritamente interiores ao polígono.
- 29) Diferentemente do exercício anterior, considere o preenchimento considerando todos os *pixels* que o polígono intercepta.
- 30) Conforme discutido em sala de aula, implemente um algoritmo para preencher um retângulo, definido por coordenadas reais, preenchendo os *pixels* da grade discreta, que tenham intersecção com as arestas esquerda e inferior do retângulo e não preenchendo os *pixels* da grade que interceptam o retângulo nas arestas superior e direita. Qual a vantagem desta abordagem?
- 31) Refaça todos os exercícios anteriores, mas agora considere polígonos arbitrários, incluindo aqueles onde existam intersecção de arestas.

- 32) Implemente um programa onde o usuário entre com as coordenadas reais de quatro polígonos de formas arbitrárias. O número de pontos de cada polígono será dado pelo usuário. Definidos os polígonos no espaço bidimensional, preencha cada um deles com uma cor diferente seguindo a abordagem do exercício 30. Considere também o caso onde não exista separação entre dois polígonos, ou seja, as coordenadas de algumas arestas são as mesmas para duas figuras.
- 33) Implemente uma função que, considerando uma janela de visualização que contemple parcialmente um objeto no SRU, seja capaz de preenchê-lo adequadamente no SRD. Reflita sobre o que é necessário considerar para a execução correta desse algoritmo.
- 34) Considere um sistema de *coordenadas cartesiano bidimensional* e 5 (cinco) pontos que definem um objeto no plano, dados por  $P_1=(3,3)$ ,  $P_2=(6,3)$ ,  $P_3=(6,6)$ ,  $P_4=(4.5,8)$  e  $P_5=(3,6)$ . Implemente 6 funções para realizar a translação, escalonamento, rotação e espelhamentos em relação ao eixo x, em relação ao eixo y e em relação aos dois eixos, simultaneamente. Você não deve passar as matrizes de transformação para os parâmetros formais das funções, apenas as coordenadas do ponto e os parâmetros necessários para as operações que serão realizadas dentro de cada rotina. As funções devem retornar os pontos modificados. Uma vez implementadas as operações, aplique cada uma delas sobre o objeto definido pelos pontos acima. (Não é para trabalhar com o conceito de Operadores Lineares ainda neste exercício)
- 35) Considere o objeto definido pelos pontos do exercício anterior. Aplique sobre ele uma operação de rotação de  $70^\circ$  (ou algum outro ângulo qualquer). Juntamente com a operação de rotação, verifique que uma outra operação aconteceu implicitamente sobre os pontos que definem o objeto. Que operação foi essa? Por que isso aconteceu? Você saberia como resolver este problema?
- 36) Implemente uma função que, dado um objeto posicionado em algum lugar arbitrário do plano, centralize o mesmo na origem do sistema de coordenadas. Para simplificar, considere o objeto definido pelos pontos dados no exercício 34.
- 37) Considere a função implementada no exercício anterior e antes de fazer uma rotação de  $70^\circ$ , centralize o objeto no centro do sistema de coordenadas. Que conclusões você pode retirar deste exercício?
- 38) A transformação de cisalhamento (*shearing* ou *skew*) no plano é uma operação que distorce o formato de um objeto. Neste tipo de operação, aplica-se um deslocamento aos valores das coordenadas do objeto proporcional ao valor das outras coordenadas de cada ponto transformado. Por exemplo, uma distorção na direção x, proporcional a coordenada y, pode ser obtida considerando as seguintes novas coordenadas:  $x' = x + S \cdot y$  e  $y' = y$ . Note a presença da coordenada y na composição da coordenada x', onde S é um valor constante. Implemente uma função para realizar o cisalhamento sobre um objeto dado. Descreva a matriz que define a transformação de cisalhamento neste caso particular.
- 39) Suponha um sistema de coordenadas homogêneas para o SRU bidimensional. Como ficam as matrizes de transformação para cada uma das operações dos exercícios acima (a partir do 34)?
- 40) Implemente uma função que multiplique uma matriz por um vetor e, considerando o objeto definido no exercício 34, realize alguns experimentos para translação, escalonamento, rotação e espelhamentos em relação ao eixo x, em relação ao eixo y e em relação aos dois eixos, simultaneamente, deste objeto, considerando o conceito de coordenadas homogêneas. (A ideia deste exercício é incorporar o conceito de Operadores Lineares)