

Trabalho I

A Entrada

A entrada do algoritmo será basicamente um conjunto de formas geométricas básicas (retângulos, círculos, etc) dispostos numa região do plano cartesiano .

Considere a Ilustração 1. Cada forma geométrica é definida por uma coordenada âncora (marcada, na figura, por um pequeno ponto vermelho) e por suas dimensões. A coordenada âncora do círculo é o seu centro e sua dimensão é definida por seu raio (r , na figura). A coordenada âncora do retângulo é seu canto inferior esquerdo¹ e suas dimensões são sua largura (w) e sua altura (h). A coordenada âncora de um texto, normalmente, é o início do texto, porém, pode ser definida como o meio ou o fim do texto. Por fim, uma linha é determinada por duas âncoras em suas extremidades. As coordenadas que posicionam as formas geométricas são valores reais.

Cada forma geométrica é identificada por um número inteiro.

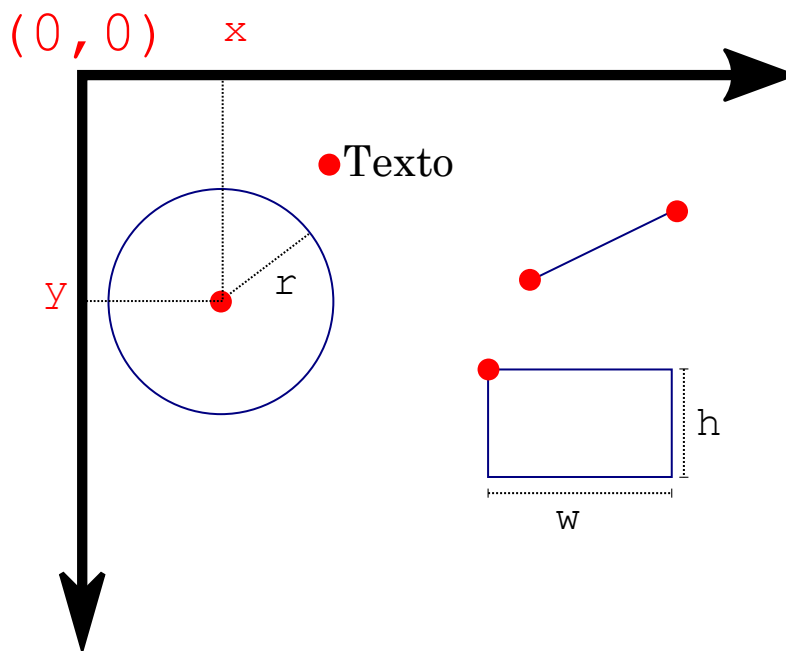


Ilustração 1: Formas no plano

As tabelas abaixo mostram os formatos dos arquivos de entrada (.geo e .qry). Cada comando tem um certo número de parâmetros. Os parâmetros mais comuns são:

- i, j, k : número inteiro, maior ou igual a 1. Identificador de uma forma geométrica.
- r : número real. Raio do círculo.
- x, y : números reais. Coordenada (x,y) .
- cor : string. Cor válida dentro do padrão SVG.²

Alguns comandos utilizam memória auxiliar para armazenar identificadores

¹ Note que o plano cartesiano está desenhado "de ponta-cabeça" em relação à representação usual.

² <http://www.december.com/html/spec/colorsvg.html>.

<https://www.w3.org/Graphics/SVG/IG/resources/svgprimer.html>

comando	parâmetros	descrição
c	i x y r corb corp	<i>desenhar círculo. corb é a cor da borda e corp é a cor do preenchimento</i>
r	i x y w h corb corp	<i>desenhar retângulo: w é a largura do retângulo e h, a altura. corb é a cor da borda e corp é a cor do preenchimento</i>
l	i x1 y1 x2 y2 cor	<i>Desenhar linha com extremidades nos pontos (x1,y1) e (x2,y2), com a cor especificada.</i>
t	i x y corb corp a txto	<i>desenha o texto txto nas coordenadas (x,y) e com a cores indicadas. corb é a cor da borda e corp é a cor do preenchimento. O parâmetro a determina a posição da âncora do texto: i, no início; m, no meio, f, no fim. O texto txto é o último parâmetro do comando. Pode incluir espaços em branco e se estende até o final da linha.</i>
comandos .geo		

Algumas consultas fazem referência a filas e listas. O programa deve manter, pelo menos 1 fila para o polígono corrente e uma lista para as figuras selecionadas.

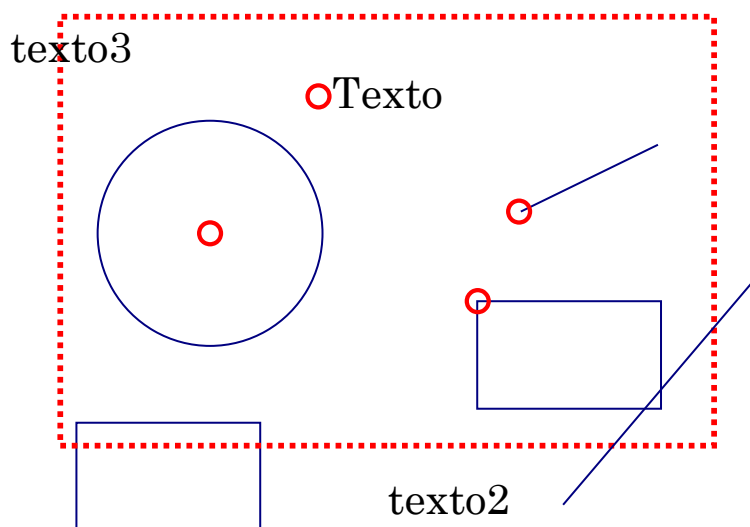
comando	parâmetros	descrição
inp	i	<i>Inserir no polígono corrente as coordenadas da âncora da figura identificada por i. No caso de linhas, inserir a extremidade de menor x (se empate, de menor y) TXT. Reportar a coordenada inserida e os dados da respectiva figura</i>
rmp		<i>Remove a coordenada mais antiga inserida no polígono corrente. TXT. Reportar a coordenada removida</i>
pol	i d e corb corp	<i>Produz um conjunto de linhas (e os insere no "banco de dados"). As linhas produzidas correspondem à borda do polígono e as linhas de seu preenchimento. Os identificadores das linhas produzidas são sequenciais, a partir de i. O parâmetro d é a distância entre as linhas do preenchimento; e é a espessura das linhas; corb a cor das linhas do polígono; corp, a cor das linhas do preenchimento.</i>
clp		<i>Remove todas as coordenadas do polígono corrente</i>
sel	x y w h	<i>Seleciona as figuras inteiramente dentro da região especificada pelos parâmetros. (Desconsidera seleções anteriores) SVG: desenhar região. Desenhar um anel vermelho em volta da âncora das figuras selecionadas. TXT: reportar identificador e tipo das figuras selecionadas</i>
sel+	x y w h	<i>Semelhante à operação sel, porém, acrescenta (união) as figuras selecionadas às aquelas anteriormente selecionadas. ATENÇÃO: as figuras selecionadas por este comando devem suceder (na lista) às aquelas selecionadas por comandos anteriores. SVG: igual a sel. TXT: igual a sel. Também informar o total de figuras anteriormente selecionadas e o número total das figuras selecionadas por este comando</i>

comando	parâmetros	descrição
dels		Remove do “banco de dados” todas as figuras selecionadas. SVG: A figuras removidas naturalmente não aparecerão no SVG (pois, não existem mais) TXT: Reportar o id e os dados das figuras removidas
dps	i dx dy corb corp	Cria novas formas, semelhantes às selecionadas, porém, transladadas de dx e dy, cujos identificadores começam em i e são incrementados sequencialmente .
ups	corb corp dx dy n	Altera as cores e a posição de n figuras antes (se $n < 0$) ou depois (se $n > 0$) da última figura selecionada pelo último sel ou sel+. Atribui corb à borda das figuras e corp ao preenchimento. Translada a figura em dx unidades na horizontal e incrementalmente dy na vertical. Por exemplo, sejam 3 figuras com âncoras $(x1,y1)$, $(x2,y2)$, $(x3,y3)$. Elas serão transladada para $(x1+dx,y1+dy)$, $(x2+dx,y2+2*dy)$ e $(x3+dx,y3+3*dy)$. Note que dx e dy podem ser valores negativos.
Comandos .qry		

A Saída

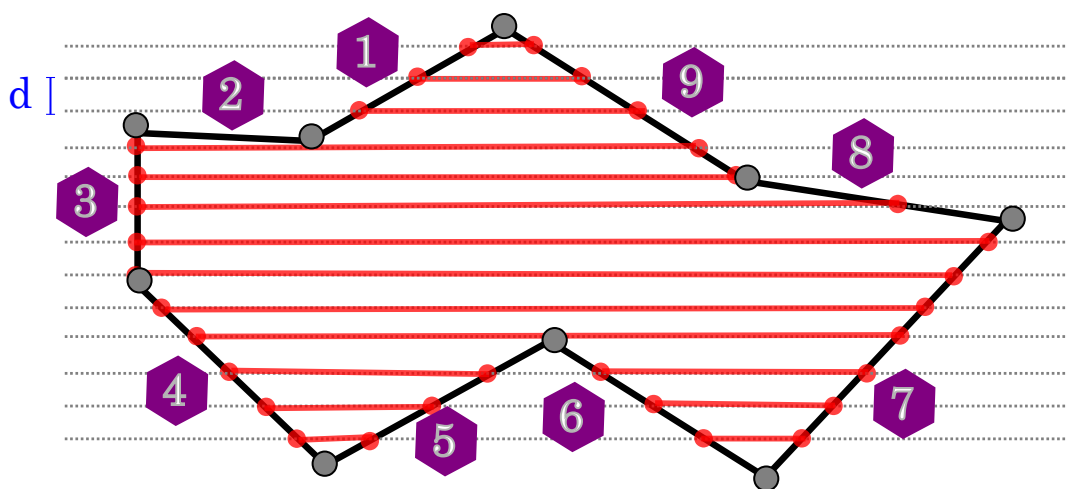
A saída deve estar de acordo com a descrição dos comandos. Vale a pena ilustrar alguns destes comandos.

Os comandos **sel** e **sel+** devem mostrar a região de seleção e a âncora das figuras selecionadas.



A figura abaixo ilustra o efeito do comando **pol**. Os pontos cinzas correspondem às coordenadas produzidas pelo comando **inp**. Os segmentos pretos são as bordas do polígono determinado pela ordem de execução de comandos **inp**. Estes segmentos devem ser acrescentados ao “banco de dados”. Os segmentos vermelhos constituem o preenchimento do polígono. Também devem ser inseridos no “banco de dados”. Porém, os pontos vermelhos e as linhas tracejadas cinzas são apenas explicativas e não devem ser inseridas no “banco de dados”.

O algoritmo para o preenchimento do polígono faz uma linha horizontal (cinza tracejada na figura) “andar” sobre o polígono em saltos de distância **d**. As intersecções desta linha com segmentos das bordas do polígono (ponto vermelho) determinam dos extremos das linhas de preenchimento.



9 lados + 16 linhas preenchimento = 25 linhas inseridas

IMPLEMENTAÇÃO

As estruturas de dados lista e fila devem ser implementadas conforme o TAD mostrado em aula.

É **terminantemente proibido** declarar structs nos arquivos de cabeçalho (.h).

O programa deve estar bem modularizado (arquivos .h e .c). Cada estrutura de dados deve estar em um módulo separado. O arquivo .h deve estar muito bem documentado (lembre-se que é um “contrato”).

Obs.: usar implementação dinâmica e encadeamento duplo para lista; implementação estática (circular) para as filas.

AValiação

Espera-se uma atitude pró-ativa para a aquisição dos conhecimentos (i.e., estudo) para resolver o problema proposto.

A avaliação consistirá da execução dos testes e da inspeção de código. Além disso, deverá ser produzido um vídeo de, aproximadamente, 5 minutos no qual apresenta o programa. O aluno deve ter por objetivo convencer o avaliador que: (a) o programa funciona; (b) o programa foi bem implementado. O vídeo deve ser colocado no Youtube e o seu link deve estar anotado no arquivo leia-me.

O Que Entregar

Submeter no Classroom o arquivo .zip com os fontes , conforme descrito anteriormente.

RESUMO DOS PARÂMETROS DO PROGRAMA TED

Parâmetro / argumento	Opcional	Descrição
-e <i>path</i>	S	Diretório-base de entrada (BED)
-f <i>arq.geo</i>	N	Arquivo com a descrição da cidade. Este arquivo deve estar sob o diretório BED .
-o <i>path</i>	N	Diretório-base de saída (BSD)
-q <i>arqcons.qry</i>	S	Arquivo com consultas. Este arquivo deve estar sob o diretório BED .

RESUMO DOS ARQUIVOS PRODUZIDOS

-f	-q	comando com sufixo	arquivos
<i>arq.geo</i>			arq.svg
<i>arq.geo</i>	<i>arqcons.qry</i>		arq.svg arq-arqcons.svg arq-arqcons.txt
<i>arq.geo</i>	<i>arqcons.qry</i>	<i>sufx</i>	arq.svg arq-arqcons.svg arq-arqcons.txt arq-arqcons-sufx.[svg txt] ³

ATENÇÃO:

- * os fontes devem ser compilados com a opção `-fstack-protector-all`.
- * adotamos o padrão C99. Usar a opção `-std=c99`.

³ Podem ser produzidos os respectivos arquivos .svg e/ou .txt, dependendo da especificação do comando.