MAC 5747 – Geometria Computacional
Prof. José Coelho de Pina
Monitor: Fábio Henrique Viduani Martinez

LISTA DE EXERCÍCIOS 4 – RESPOSTAS

1. (Exercício 35.2-2 de [1]) Dados dois segmentos de reta a e b que não se intersectam e são comparáveis na abscissa x, mostre como usando produto interno (i. e. as primitivas `Left` ou `Lefton`, ou ..., ou o que você precisar) podemos decidir se $a >_x b$ em tempo $O(1)$.

   [Note que no algoritmo `Intersecção-de-Segmentos` as operações `Insert, Delete, Above` e `Below` usam esta comparação.]

   Sejam $a = a_-a_+$ e $b = b_-b_+$ dois segmentos de reta que não se interceptam e que são comparáveis na abscissa x. Suponha, sem perda de generalidade, que $a_-[X] < a_+[X]$ e $b_-[X] < b_+[X]$. Como a e b são comparáveis em x, temos dois casos:

   Caso 1. $b_-[X]$ está no intervalo $[a_-[X], a_+[X]]$. Ou seja, $a_-[X] \leq b_-[X] \leq a_+[X]$.
   Temos neste caso que $a >_x b$ se

   $$\texttt{Left}(b_-, b_+, a_-) \vee \texttt{Left}(b_-, b_+, a_+).$$

   Caso 2. $a_-[X]$ está no intervalo $[b_-[X], b_+[X]]$. Ou seja, $b_-[X] \leq a_-[X] \leq b_+[X]$.
   Temos então que $a >_x b$ se

   $$\texttt{Left}(a_-, a_+, b_-) \vee \texttt{Left}(a_-, a_+, b_+).$$

6. (Exercício 35.32-2 de [1]) Um *disco* (ou círculo) é a região do plano limitada por uma cricunferência. Um disco é dado através das coordenadas do seu centro e do valor do seu raio. Dois discos intersectam se eles têm algum ponto em comum. Descreva um algoritmo de complexidade de tempo $O(n \log n)$ que dados n discos decide se qualquer dois deles intersectam.

   Dados n discos, queremos saber, em tempo $O(n \log n)$, se dois deles se intersectam. A idéia para construção do algoritmo que soluciona este problema é a mesma idéia utilizada no problema em que queremos saber se dois segmentos se intersectam: fazer uma linha de varredura varrer os discos da esquerda para direita.

   As linhas verticais de varredura devem passar pelos extremos esquerdo e direito de cada disco. Desse modo, para cada disco (c_i, r_i), onde $c_i = (x_i, y_i)$ é o centro do i-ésimo disco e r_i é seu raio, computamos os pontos $e_i = (c_i - r_i, y_i)$ e $d_i = (c_i + r_i, y_i)$, para todo $i = 1, \ldots, n$. Esta lista de $2n$ pontos é inicialmente ordenada pela coordenada x.

   A linha de varredura determina a ordem em que os discos aparecem na coordenada y. Dois discos que se intersectam são vizinhos nesta ordenação quando encontramos o extremo esquerdo do segundo ou quando encontramos o extremo direito do primeiro.

   O algoritmo `Intersecção-de-Discos` é descrito a seguir e, como mencionado, é muito semelhante ao algoritmo para determinar se dois segmentos se intersectam.

Intersecção-de-Discos

Entrada: n discos $\langle (c_1, r_1), (c_2, r_2), \ldots, (c_n, r_n) \rangle$.

Saída: dois discos se intersectam/não se intersectam (`sim/não`).

01. $T \leftarrow \emptyset$
02. **para** cada centro $c_i = (x_i, y_i)$ do disco $d_i = (c_i, r_i)$, crie dois pontos $(x_i - r_i, y_i)$ e $(x_i + r_i, y_i)$
03. ordene a lista de $2n$ pontos pela coordenada x, com desempate ordenando pela coordenada y
04. **para** cada ponto p da lista ordenada de pontos
05.   **se** p é extremo esquerdo de um disco d
06.   Insere(T, d)
07.   **se** (Acima(T, d) existe e intersecta d) **ou** (Abaixo(T, d) existe e intersecta d)
08.   **devolva** "intersectam"
09.   **se** p é extremo direito de um disco d
10.   **se** Acima(T, d) **e** Abaixo(T, d) existem e se intersectam
11.   **devolva** "intersectam"
12.   Remove(T, d)
13. **devolva** "não se intersectam"

Note que a verificação da intersecção de dois discos, realizada nas linhas 07 e 10, pode ser feita em tempo constante, bastando computar a distância entre os pontos dos discos correspondentes e a soma de seus raios, comparando estes valores.

Portanto, como no algoritmo Intersecção-de-Segmentos, o tempo de execução do algoritmo Intersecção-de-Discos é $O(n \log n)$.