Geometria Computacional

Cristina G. Fernandes

Departamento de Ciência da Computação do IME-USP

http://www.ime.usp.br/~cris/

segundo semestre de 2018

Dentro ou fora?

Candidata a diagonal $P[i]P[j]$ está no interior do polígono?

Está no cone das arestas vizinhas do polígono?

```
NoCone(n, P, i, j)
    u ← i − 1 (mod n)
    w ← i + 1 (mod n)
    se Esquerda(P[u], P[i], P[w])                    ▷ P[i] é convexo
        então devolva Esquerda⁺(P[i], P[j], P[u]) e
                      Esquerda⁺(P[j], P[i], P[w])
        senão devolva não (Esquerda(P[i], P[j], P[w]) e
                           Esquerda(P[j], P[i], P[u]))
```

Teste de diagonal

Quase uma diagonal...

QuaseDiagonal(n, P, i, j)

para $k \leftarrow 0$ **até** $n-1$
    $\ell \leftarrow k + 1 \pmod{n}$
    se $k \neq i$ **e** $k \neq j$ **e** $\ell \neq i$ **e** $\ell \neq j$
        então se Intersecta$(P[i], P[j], P[k], P[\ell])$
            então devolva FALSO
devolva VERDADE

Diagonal de fato...

Diagonal(n, P, i, j)

devolva NoCone(n, P, i, j) **e** QuaseDiagonal(n, P, i, j)

Tempo de execução: $\Theta(n)$

Interseção de segmentos

Ponto c está no segmento ab

Entre(a, b, c)
se não Colinear(a, b, c)
    então devolva FALSO
se $a[X] \neq b[X]$ ▷ ab não é vertical
    então devolva $a[X] \leq c[X] \leq b[X]$ **ou** $b[X] \leq c[X] \leq a[X]$
    senão devolva $a[Y] \leq c[Y] \leq b[Y]$ **ou** $b[Y] \leq c[Y] \leq a[Y]$

Interseção entre ab e cd

Intersecta(a, b, c, d)
se IntersectaProp(a, b, c, d)
    então devolva VERDADE
devolva Entre(a, b, c) **ou** Entre(a, b, d)
        ou Entre(c, d, a) **ou** Entre(c, d, b)

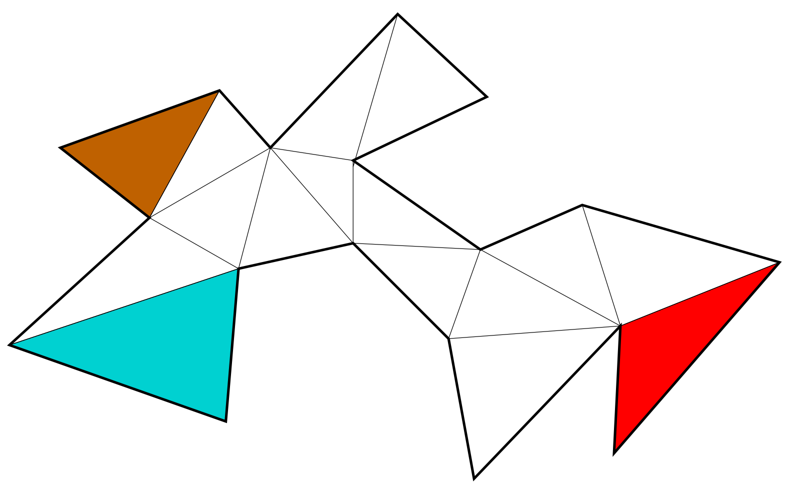
Orelhas de polígonos

Teorema (Meister's Two Ears Theorem): Todo polígono com pelo menos 4 vértices possui pelo menos duas orelhas.

Segue do teorema abaixo.

Teorema 3: Seja P um polígono com pelo menos 4 vértices e T uma triangulação de P. Então pelo menos dois triangulos de T formam orelhas de P.

Prova: (Feita na aula.)



Triangulação em $\mathrm{O}(n^3)$: use orelhas!

PontaDeOrelha(n, P, i)

$j \leftarrow (i-1) \bmod n$
$k \leftarrow (i+1) \bmod n$
devolva Diagonal(n, P, j, k)

Triangulação em $\mathrm{O}(n^3)$: use orelhas!

PontaDeOrelha(n, P, i)
  $j \leftarrow (i-1) \bmod n$
  $k \leftarrow (i+1) \bmod n$
  devolva Diagonal(n, P, j, k)

Triang-n3(n, P)
  se $n > 3$
    então $i \leftarrow 0$
      enquanto não PontaDeOrelha(n, P, i) **faça**
        $i \leftarrow i + 1$
      imprima $\{(i-1) \bmod n, (i+1) \bmod n\}$
      $m, P' \leftarrow$ Remove(n, P, i)
      Triang-n3(m, P')

Triangulação em $\mathrm{O}(n^3)$: use orelhas!

PontaDeOrelha(n, P, i)
$j \leftarrow (i-1) \bmod n$
$k \leftarrow (i+1) \bmod n$
devolva Diagonal(n, P, j, k)

Triang-n3(n, P) $\triangleright$ sem recursão de cauda
enquanto $n > 3$
$i \leftarrow 0$
enquanto não PontaDeOrelha(n, P, i) **faça**
$i \leftarrow i + 1$
imprima $\{(i-1) \bmod n, (i+1) \bmod n\}$
$P[i \,..\, n-2] \leftarrow P[i+1 \,..\, n-1]$ $\triangleright$ remove o i
$n \leftarrow n - 1$

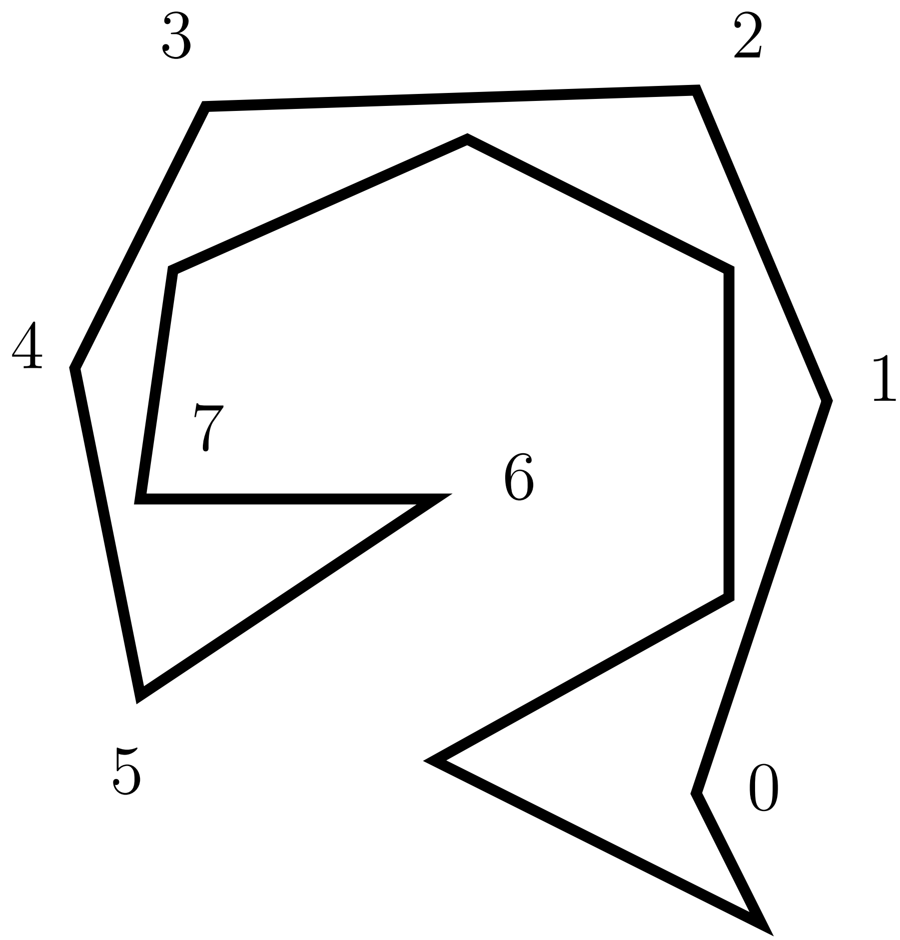
Triangulação em $\mathrm{O}(n^3)$: use orelhas!

```
PontaDeOrelha(n, P, i)
    j ← (i − 1) mod n
    k ← (i + 1) mod n
    devolva Diagonal(n, P, j, k)
```

```
Triang-n3(n, P)          ▷ sem recursão de cauda
    enquanto n > 3
        i ← 0
        enquanto não PontaDeOrelha(n, P, i) faça
            i ← i + 1
        imprima {(i − 1) mod n, (i + 1) mod n}
        P[i . . n − 2] ← P[i + 1 . . n − 1]     ▷ remove o i
        n ← n − 1
```

Pior caso: $\Theta(n^2)$ chamadas a Diagonal

Exemplo ruim



Triangulação em $\mathrm{O}(n^2)$

Triang-n2(n, P)

MarcaOrelha(P)
enquanto $n > 3$ **faça**
    $v_2 \leftarrow P$
    enquanto não $orelha[v_2]$ **faça**
        $v_2 \leftarrow prox[v_2]$
    $v_1 \leftarrow prev[v_2]$
    $v_3 \leftarrow prox[v_2]$
    imprima $\{vert[v_1], vert[v_3]\}$
    $prox[v_1] \leftarrow v_3$
    $prev[v_3] \leftarrow v_1$
    $P \leftarrow v_3$
    $n \leftarrow n - 1$
    $orelha[v_1] \leftarrow$ PontaDeOrelha(P, v_1)
    $orelha[v_3] \leftarrow$ PontaDeOrelha(P, v_3)

Triangulação em $\mathrm{O}(n^2)$

Triang-n2(n, P)

MarcaOrelha(P)
enquanto $n > 3$ **faça**
  $v_2 \leftarrow P$
  enquanto não $orelha[v_2]$ **faça**
    $v_2 \leftarrow prox[v_2]$
  $v_1 \leftarrow prev[v_2]$
  $v_3 \leftarrow prox[v_2]$
  imprima $\{vert[v_1], vert[v_3]\}$
  $prox[v_1] \leftarrow v_3$
  $prev[v_3] \leftarrow v_1$
  $P \leftarrow v_3$
  $n \leftarrow n - 1$
  $orelha[v_1] \leftarrow$ PontaDeOrelha(P, v_1)
  $orelha[v_3] \leftarrow$ PontaDeOrelha(P, v_3)

Pior caso: $\Theta(n)$ chamadas a Diagonal.

Orelhas com listas ligadas

PontaDeOrelha(P, v)
1 $u \leftarrow prev[v]$
2 $w \leftarrow prox[v]$
3 **devolva** Diagonal(P, u, w)

Orelhas com listas ligadas

PontaDeOrelha(P, v)

$u \leftarrow prev[v]$
$w \leftarrow prox[v]$
devolva Diagonal(P, u, w)

MarcaOrelha(P)

$v \leftarrow P$
repita
    $u \leftarrow prev[v]$
    $w \leftarrow prox[v]$
    $orelha[v] \leftarrow$ Diagonal(P, u, w)
    $v \leftarrow w$
até que $v = P$

Orelhas com listas ligadas

PontaDeOrelha(P, v)

$u \leftarrow prev[v]$
$w \leftarrow prox[v]$
devolva Diagonal(P, u, w)

MarcaOrelha(P)

$v \leftarrow P$
repita
    $u \leftarrow prev[v]$
    $w \leftarrow prox[v]$
    $orelha[v] \leftarrow$ Diagonal(P, u, w)
    $v \leftarrow w$
até que $v = P$

Diagonal também teria que ser reescrita para listas ligadas.

Interseção de segmentos

Uma coleção de segmentos do plano é dada por dois vetores $e[1\,..\,n], d[1\,..\,n]$ de pontos.

Interseção de segmentos

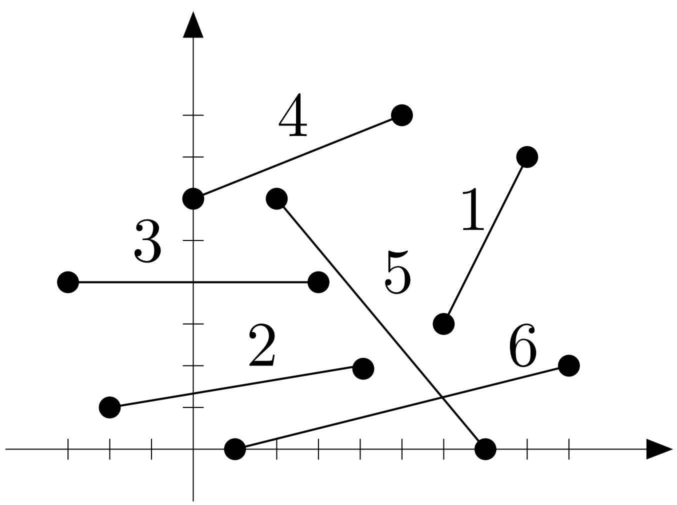
Uma coleção de segmentos do plano é dada por dois vetores $e[1\,..\,n], d[1\,..\,n]$ de pontos.

A coordenada do ponto $e[i]$ é $(e_X[i], e_Y[i])$.
A coordenada do ponto $d[i]$ é $(d_X[i], d_Y[i])$.

Interseção de segmentos

Uma coleção de segmentos do plano é dada por dois vetores $e[1\,..\,n], d[1\,..\,n]$ de pontos.

A coordenada do ponto $e[i]$ é $(e_X[i], e_Y[i])$.
A coordenada do ponto $d[i]$ é $(d_X[i], d_Y[i])$.

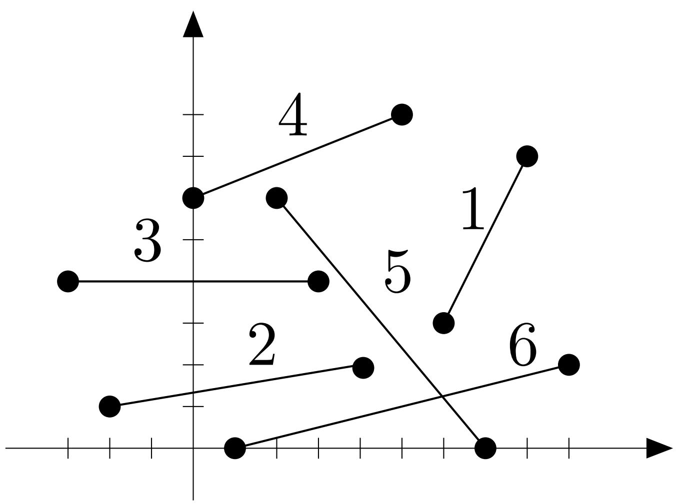


	1	2	3	4	5	6
e_X	6	−2	−3	0	3	4
e_Y	3	1	4	6	5	1

	1	2	3	4	5	6
d_X	8	1	2	5	7	9
d_Y	7	3	4	8	0	2

Interseção de segmentos

Problema: Dada uma coleção de segmentos no plano, decidir se existem dois segmentos na coleção que se intersectam.

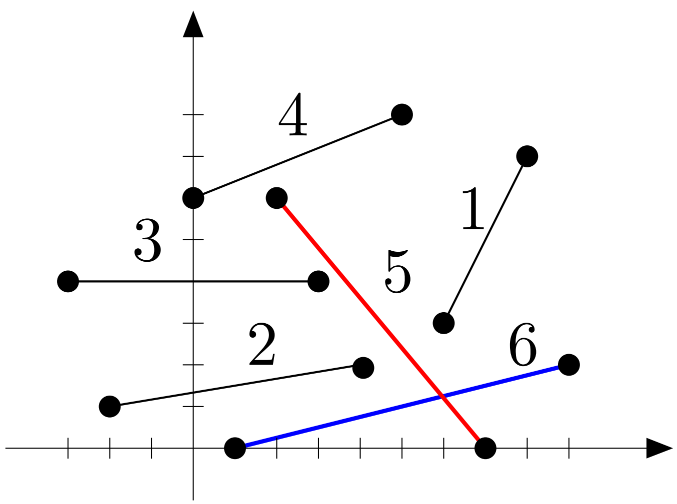


e_X	6	-2	-3	0	3	4
e_Y	3	1	4	6	5	1
	1	2	3	4	5	6

d_X	8	1	2	5	7	9
d_Y	7	3	4	8	0	2
	1	2	3	4	5	6

Interseção de segmentos

Problema: Dada uma coleção de segmentos no plano, decidir se existem dois segmentos na coleção que se intersectam.

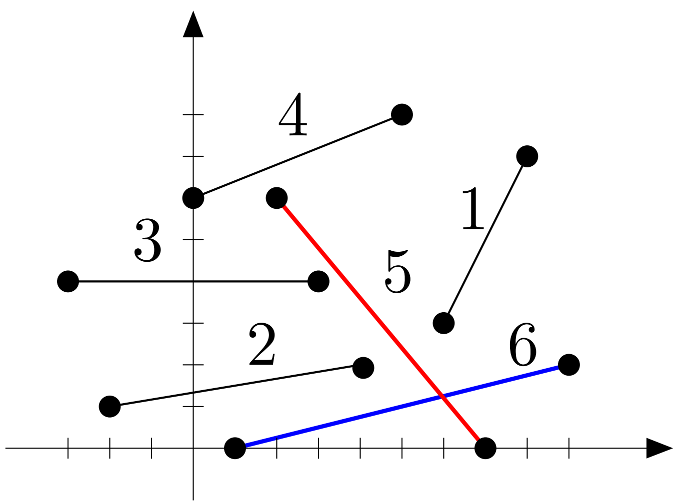


e_X	6	-2	-3	0	3	4
e_Y	3	1	4	6	5	1
	1	2	3	4	5	6

d_X	8	1	2	5	7	9
d_Y	7	3	4	8	0	2
	1	2	3	4	5	6

Interseção de segmentos

Problema: Dada uma coleção de segmentos no plano, decidir se existem dois segmentos na coleção que se intersectam.



	1	2	3	4	5	6
e_X	6	-2	-3	0	3	4
e_Y	3	1	4	6	5	1

	1	2	3	4	5	6
d_X	8	1	2	5	7	9
d_Y	7	3	4	8	0	2

Resposta: sim, existem dois segmentos com interseção.

Interseção de dois segmentos

Das aulas passadas...

Interseção entre ab e cd

Intersecta(a, b, c, d)
1 se IntersectaProp(a, b, c, d)
2 então devolva VERDADE
3 devolva Entre(a, b, c) ou Entre(a, b, d)
ou Entre(c, d, a) ou Entre(c, d, b)

Interseção de dois segmentos

Das aulas passadas...

Interseção entre ab e cd

Intersecta(a, b, c, d)
1 se IntersectaProp(a, b, c, d)
2 então devolva VERDADE
3 devolva Entre(a, b, c) ou Entre(a, b, d)
ou Entre(c, d, a) ou Entre(c, d, b)

Abreviatura:

INTER(e, d, i, j)
1 devolva Intersecta$(e[i], d[i], e[j], d[j])$

Interseção de segmentos

Solução quadrática:

IntersectaQuad(e, d, n)
para $i \leftarrow 1$ até $n-1$ faça
  para $j \leftarrow i+1$ até n faça
    se INTER (e, d, i, j)
      então devolva VERDADE
devolva FALSO

Interseção de segmentos

Solução quadrática:

IntersectaQuad(e, d, n)
1 para $i \leftarrow 1$ até $n-1$ faça
2 para $j \leftarrow i+1$ até n faça
3 se INTER (e, d, i, j)
4 então devolva VERDADE
5 devolva FALSO

Consumo de tempo: $\Theta(n^2)$.

Interseção de segmentos

Solução quadrática:

IntersectaQuad(e, d, n)

```
para i ← 1 até n−1 faça
    para j ← i+1 até n faça
        se INTER (e, d, i, j)
            então devolva VERDADE
devolva FALSO
```

Consumo de tempo: $\Theta(n^2)$.

Conseguimos fazer melhor que isso?

Interseção de intervalos

Este é o caso na reta.

Interseção de intervalos

Este é o caso na reta.

Um segmento na reta é um intervalo.

Interseção de intervalos

Este é o caso na reta.

Um segmento na reta é um intervalo.

Os vetores $e_X[1\,.\,.\,n]$ e $d_X[1\,.\,.\,n]$ representam os intervalos $[e_X[1]\,.\,.\,d_X[1]], \ldots, [e_X[n]\,.\,.\,d_X[n]]$.

Interseção de intervalos

Este é o caso na reta.

Um segmento na reta é um intervalo.

Os vetores $e_X[1\,.\,.\,n]$ e $d_X[1\,.\,.\,n]$ representam os intervalos $[e_X[1]\,.\,.\,d_X[1]], \ldots, [e_X[n]\,.\,.\,d_X[n]]$.

Se ordenarmos os pontos extremos dos intervalos, é fácil decidir se há interseção ou não, percorrendo os pontos na ordem obtida.

Interseção de intervalos

Este é o caso na reta.

Um segmento na reta é um intervalo.

Os vetores $e_X[1\,..\,n]$ e $d_X[1\,..\,n]$ representam os intervalos $[e_X[1]\,..\,d_X[1]], \ldots, [e_X[n]\,..\,d_X[n]]$.

Se ordenarmos os pontos extremos dos intervalos, é fácil decidir se há interseção ou não, percorrendo os pontos na ordem obtida.

Basta contar quantos intervalos estão "abertos". Se houver mais do que um aberto num momento, há interseção.

Interseção de intervalos

VARREDURA(e, d, n)

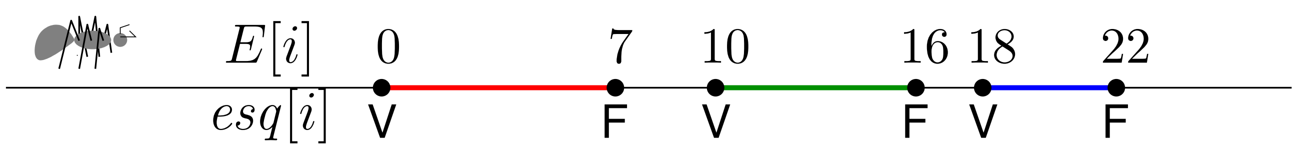
para $i \leftarrow 1$ até n faça ▷ para cada intervalo marca

  $E[i] \leftarrow e_X[i]$ $esq[i] \leftarrow$ VERDADE ▷ extremo esquerdo

  $E[i+n] \leftarrow d_X[i]$ $esq[i+n] \leftarrow$ FALSO ▷ extremo direito

MERGESORT$(E, esq, 1, 2n)$ ▷ ordena os extremos

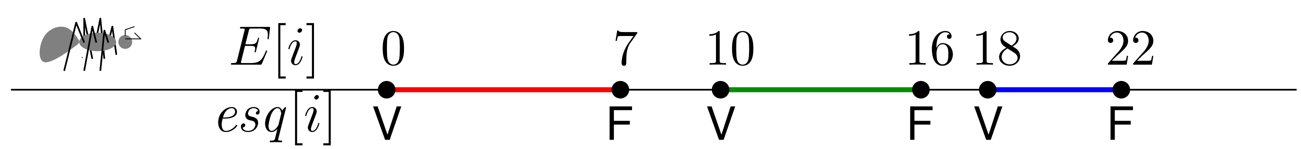
e_X	10	0	18
d_X	16	7	22
	1	2	3



Interseção de intervalos

```
VARREDURA(e, d, n)
  para i ← 1 até n faça                      ▷ para cada intervalo marca
      E[i] ← e_X[i]      esq[i] ← VERDADE    ▷ extremo esquerdo
      E[i+n] ← d_X[i]    esq[i+n] ← FALSO    ▷ extremo direito
  MERGESORT(E, esq, 1, 2n)                   ▷ ordena os extremos
  cont ← 0     resp ← FALSO
  para p ← 1 até 2n faça                     ▷ para cada ponto extremo
      se esq[p]                              ▷ se extremo esquerdo
            então cont ← cont+1
                  se cont = 2 então resp ← VERDADE
            senão cont ← cont−1
  devolva resp
```

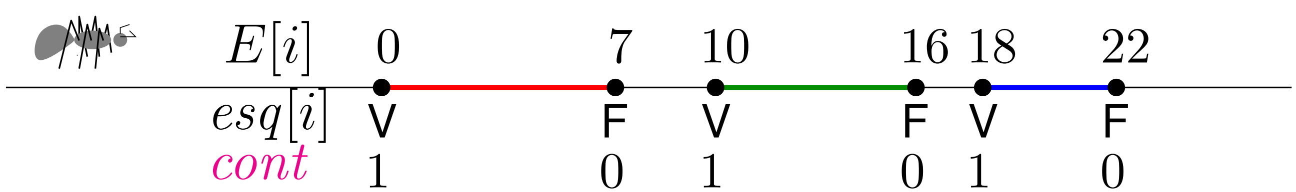
	1	2	3
e_X	10	0	18
d_X	16	7	22



Interseção de intervalos

```
VARREDURA(e, d, n)
para i ← 1 até n faça                                 ▷ para cada intervalo marca
    E[i] ← e_X[i]          esq[i] ← VERDADE           ▷ extremo esquerdo
    E[i+n] ← d_X[i]        esq[i+n] ← FALSO           ▷ extremo direito
MERGESORT(E, esq, 1, 2n)                              ▷ ordena os extremos
cont ← 0      resp ← FALSO
para p ← 1 até 2n faça                                ▷ para cada ponto extremo
    se esq[p]                                         ▷ se extremo esquerdo
        então cont ← cont+1
                  se cont = 2 então resp ← VERDADE
        senão cont ← cont−1
devolva resp
```

e_X	10	0	18
d_X	16	7	22
	1	2	3



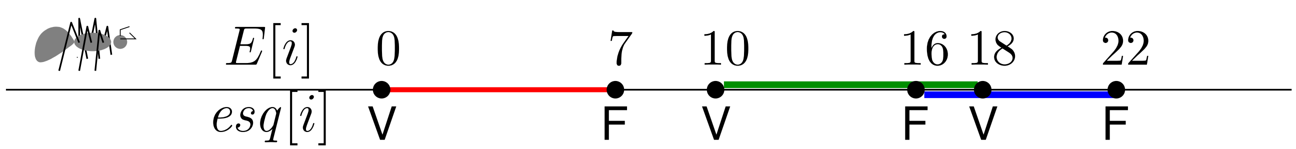
VARREDURA$(e_X, d_X, 3) =$ FALSO

Interseção de intervalos

VARREDURA(e, d, n)

para $i \leftarrow 1$ até n faça ▷ para cada intervalo marca
  $E[i] \leftarrow e_X[i]$ $esq[i] \leftarrow$ VERDADE ▷ extremo esquerdo
  $E[i+n] \leftarrow d_X[i]$ $esq[i+n] \leftarrow$ FALSO ▷ extremo direito
MERGESORT$(E, esq, 1, 2n)$ ▷ ordena os extremos
$cont \leftarrow 0$ $resp \leftarrow$ FALSO
para $p \leftarrow 1$ até $2n$ faça ▷ para cada ponto extremo
  se $esq[p]$ ▷ se extremo esquerdo
    então $cont \leftarrow cont+1$
      se $cont = 2$ então $resp \leftarrow$ VERDADE
    senão $cont \leftarrow cont-1$
devolva $resp$

e_X	10	0	16
d_X	18	7	22
	1	2	3

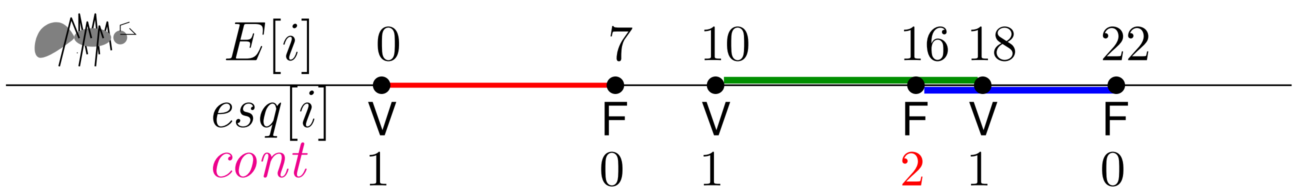


Interseção de intervalos

VARREDURA(e, d, n)

para $i \leftarrow 1$ até n faça ▷ para cada intervalo marca
    $E[i] \leftarrow e_X[i]$ $esq[i] \leftarrow$ VERDADE ▷ extremo esquerdo
    $E[i+n] \leftarrow d_X[i]$ $esq[i+n] \leftarrow$ FALSO ▷ extremo direito
MERGESORT$(E, esq, 1, 2n)$ ▷ ordena os extremos
$cont \leftarrow 0$ $resp \leftarrow$ FALSO
para $p \leftarrow 1$ até $2n$ faça ▷ para cada ponto extremo
    se $esq[p]$ ▷ se extremo esquerdo
        então $cont \leftarrow cont+1$
            se $cont = 2$ então $resp \leftarrow$ VERDADE
        senão $cont \leftarrow cont-1$
devolva $resp$

e_X	10	0	16
d_X	18	7	22
	1	2	3



VARREDURA$(e_X, d_X, 3) =$ VERDADE

Interseção de intervalos

```
VARREDURA(e, d, n)
  para i ← 1 até n faça                      ▷ para cada intervalo marca
     E[i] ← e_X[i]      esq[i] ← VERDADE      ▷ extremo esquerdo
     E[i+n] ← d_X[i]    esq[i+n] ← FALSO      ▷ extremo direito
  MERGESORT(E, esq, 1, 2n)                   ▷ ordena os extremos
  cont ← 0    resp ← FALSO
  para p ← 1 até 2n faça                     ▷ para cada ponto extremo
     se esq[p]                               ▷ se extremo esquerdo
          então cont ← cont+1
                se cont = 2 então resp ← VERDADE
          senão cont ← cont−1
  devolva resp
```

Consumo de tempo: $\Theta(n \lg n)$.

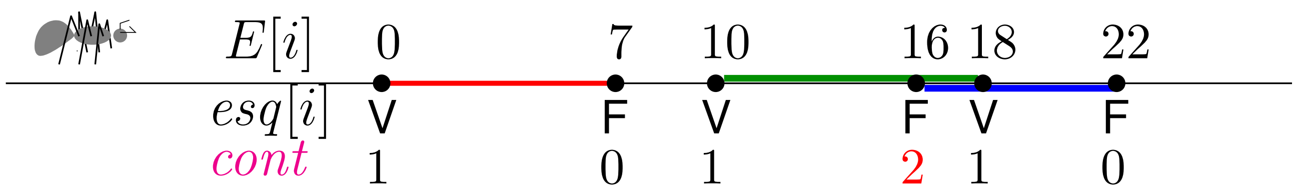
Método da linha de varredura

Ideia: reduzir um problema estático bidimensional
a um problema dinâmico unidimensional

Método da linha de varredura

Ideia: reduzir um problema estático bidimensional
a um problema dinâmico unidimensional

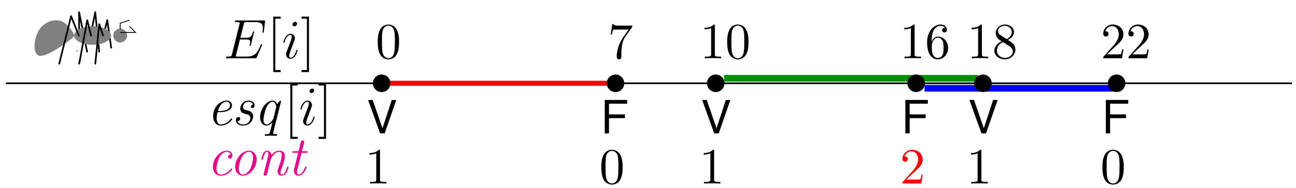
Uma linha imaginária move-se da esquerda para a direita.



Método da linha de varredura

Ideia: reduzir um problema estático bidimensional
a um problema dinâmico unidimensional

Uma linha imaginária move-se da esquerda para a direita.

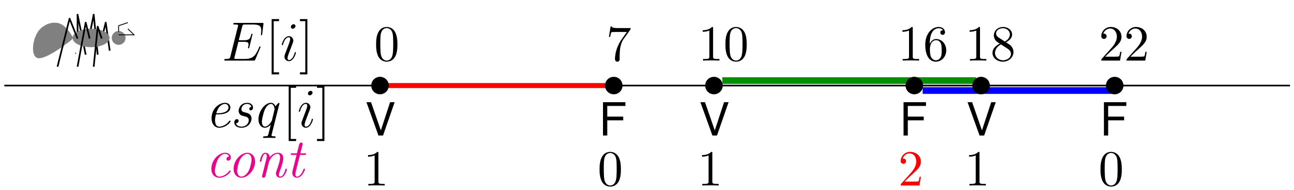


À medida que ela move,
o problema restrito à esquerda dela é resolvido.

Método da linha de varredura

Ideia: reduzir um problema estático bidimensional
a um problema dinâmico unidimensional

Uma linha imaginária move-se da esquerda para a direita.



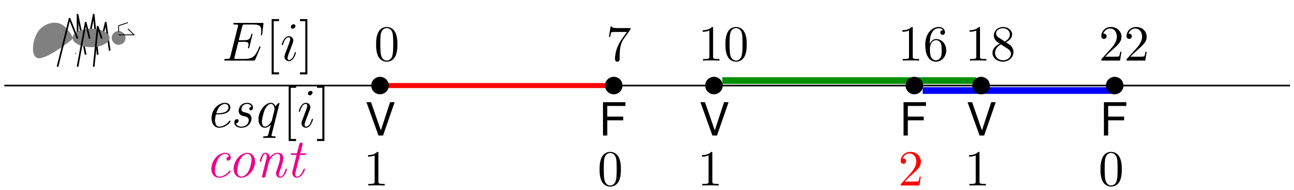
À medida que ela move,
o problema restrito à esquerda dela é resolvido.

Informação necessária para estender a solução parcial é mantida numa descrição combinatória da linha.

Método da linha de varredura

Ideia: reduzir um problema estático bidimensional
a um problema dinâmico unidimensional

Uma linha imaginária move-se da esquerda para a direita.



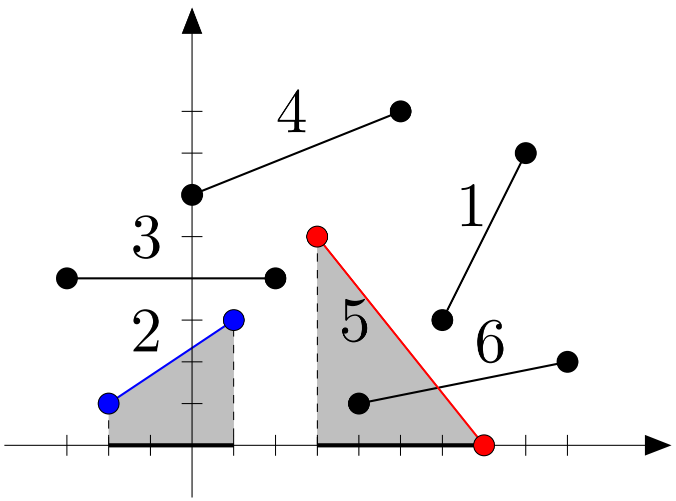
À medida que ela move,
o problema restrito à esquerda dela é resolvido.

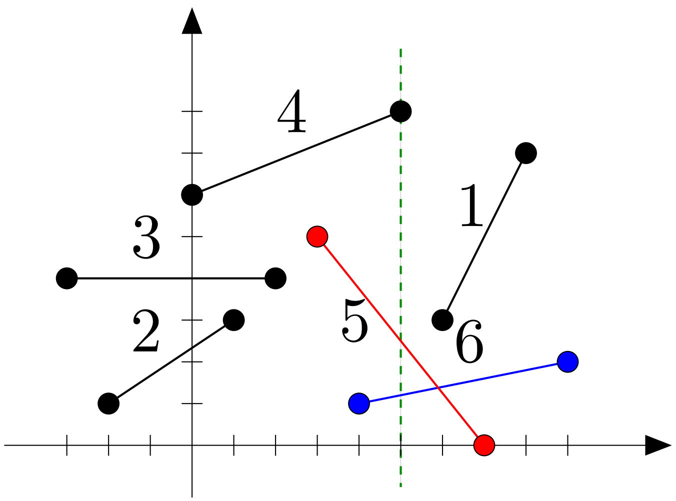
Informação necessária para estender a solução parcial é mantida numa descrição combinatória da linha.

Muda apenas em posições chaves: os pontos eventos.

Algoritmo de Shamos e Hoey

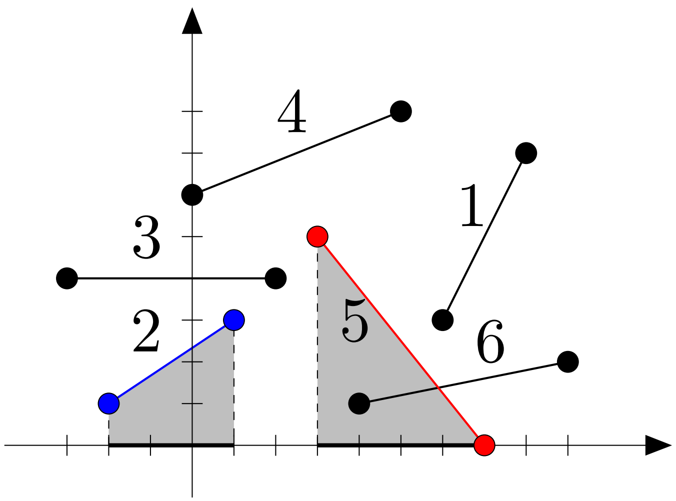
Ideia: Dois segmentos cuja projeção no eixo X sejam disjuntas não se intersectam.

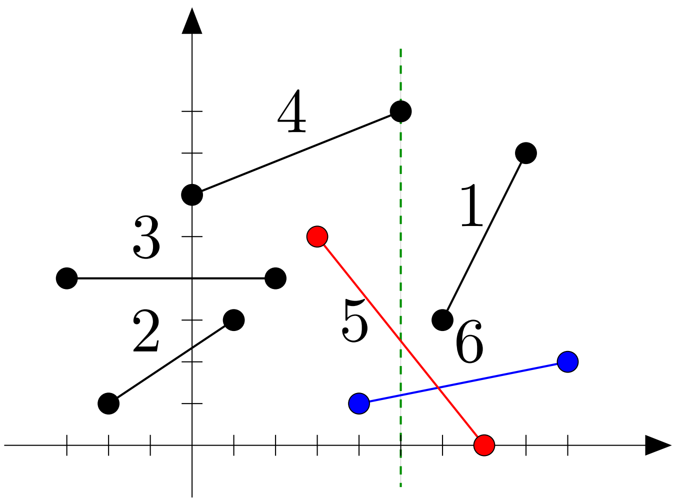




Algoritmo de Shamos e Hoey

Ideia: Dois segmentos cuja projeção no eixo X sejam disjuntas não se intersectam.

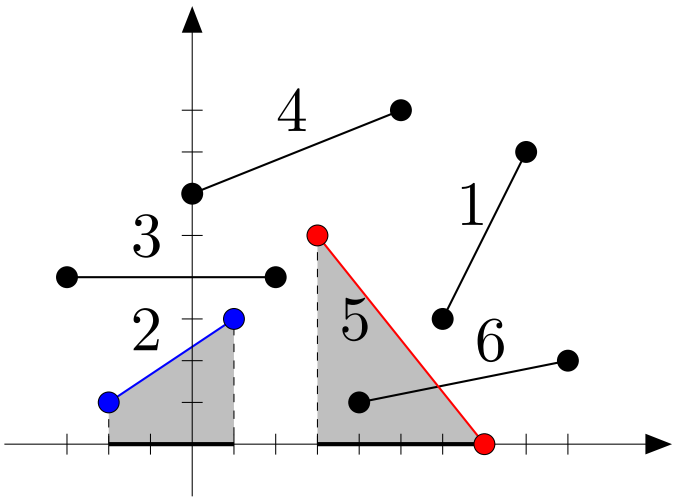


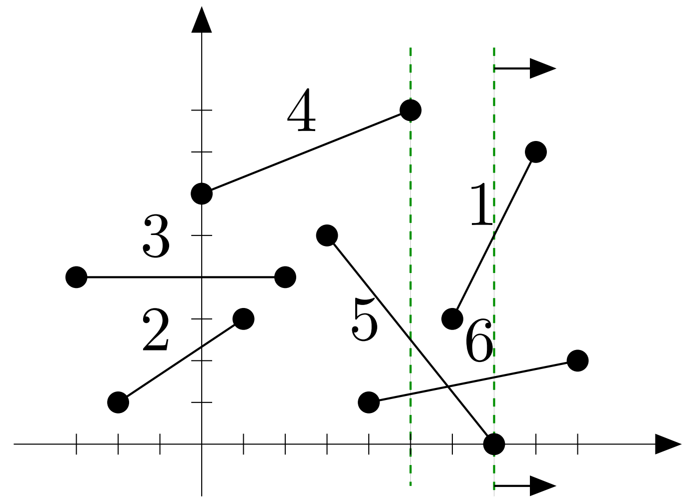


Se a projeção no eixo X de dois segmentos tem interseção, então há uma linha vertical que intersecta ambos.

Algoritmo de Shamos e Hoey

Ideia: Dois segmentos cuja projeção no eixo X sejam disjuntas não se intersectam.

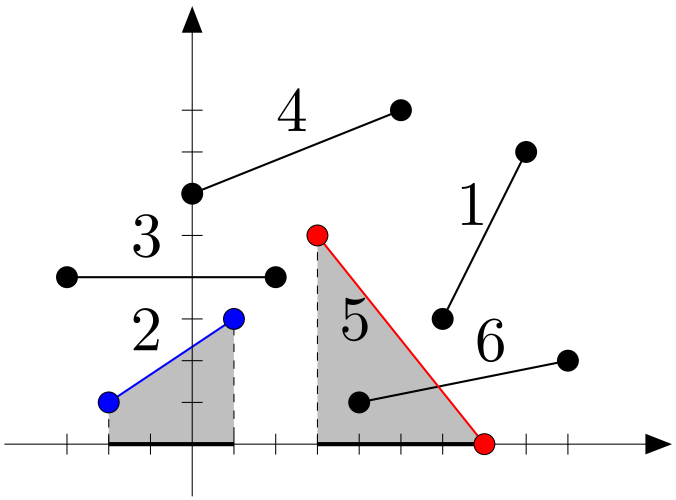


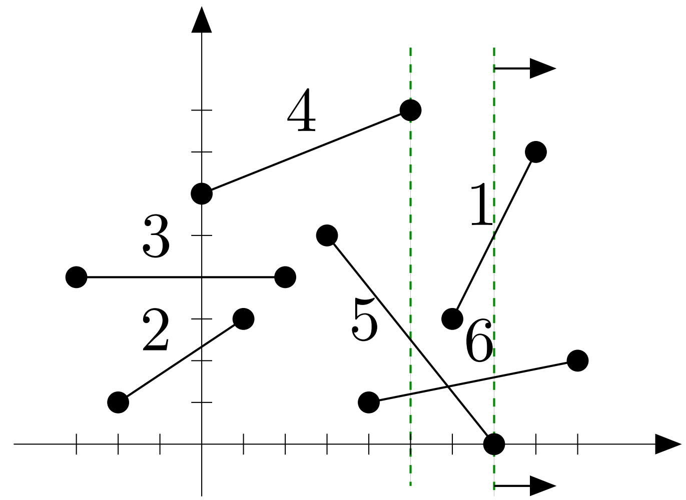


Imagine esta linha vertical varrendo o plano da esquerda para a direita...

Algoritmo de Shamos e Hoey

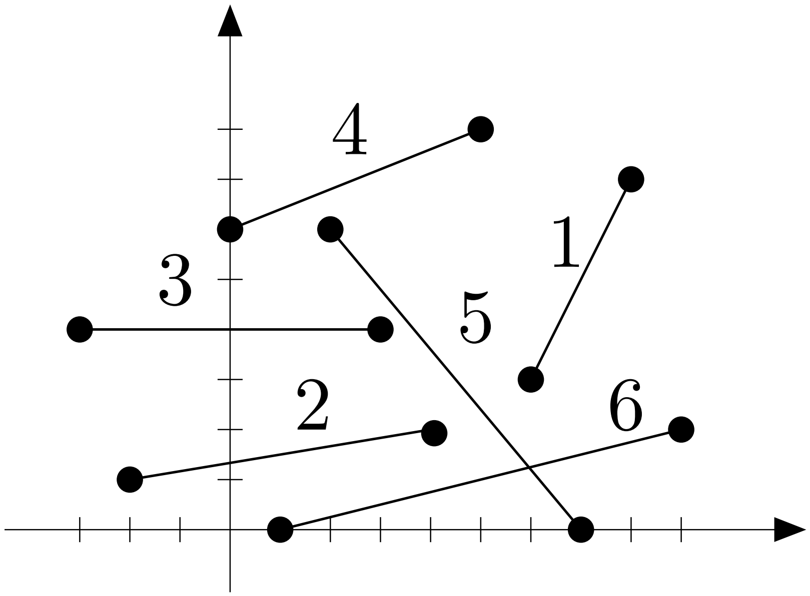
Ideia: Dois segmentos cuja projeção no eixo X sejam disjuntas não se intersectam.





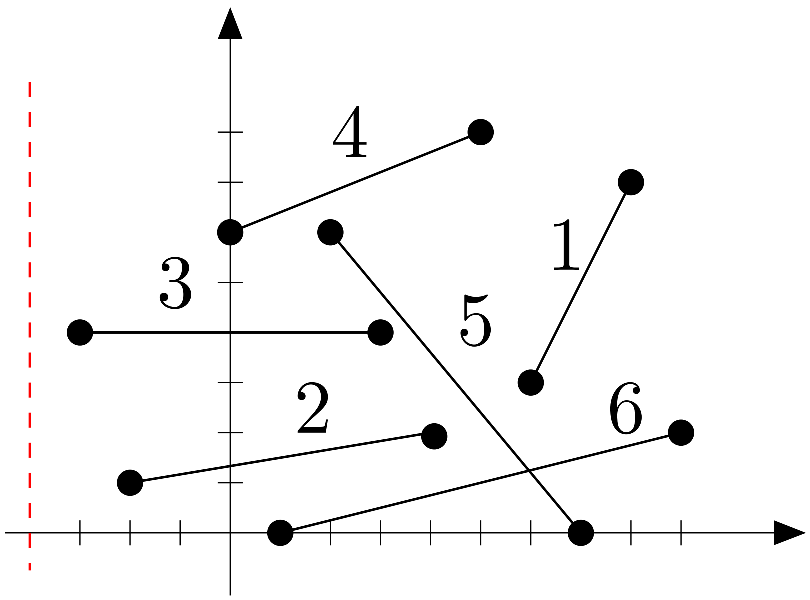
Imagine esta linha vertical varrendo o plano da esquerda para a direita...
Enquanto a linha varre o plano, mantemos os segmentos intersectados por ela na descrição combinatória da linha.

Descrição combinatória da linha



$x < -3$	$\emptyset$
$-3 \leq x < -2$	$\{3\}$
$-2 \leq x < 0$	$\{2, 3\}$
$0 \leq x < 1$	$\{2, 3, 4\}$
$1 \leq x \leq 2$	$\{2, 3, 4, 6\}$
$2 < x < 3$	$\{2, 3, 4, 5, 6\}$
$3 \leq x < 4$	$\{2, 4, 5, 6\}$
$4 \leq x \leq 5$	$\{4, 5, 6\}$
$5 < x < 6$	$\{5, 6\}$
$6 \leq x \leq 7$	$\{1, 5, 6\}$
$7 < x \leq 8$	$\{1, 6\}$
$8 < x \leq 9$	$\{6\}$
$9 < x$	$\emptyset$

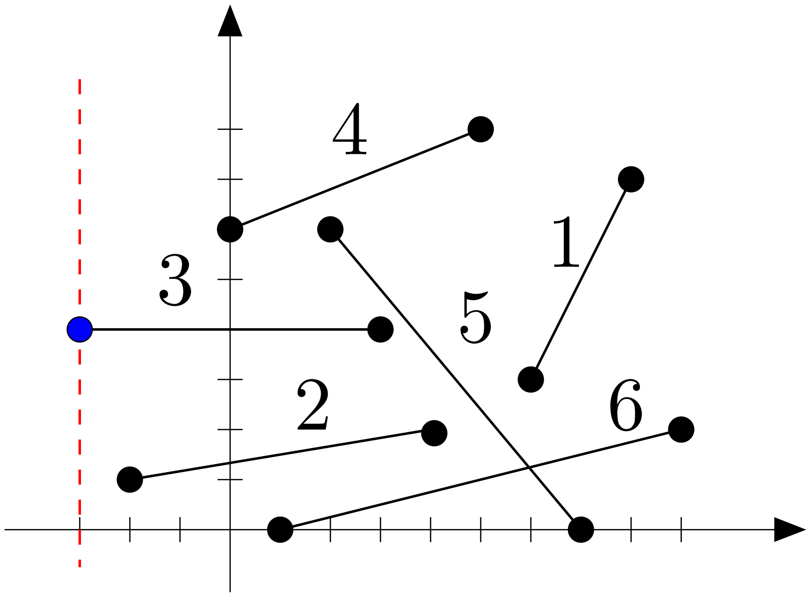
Descrição combinatória da linha



$x < -3$	$\emptyset$
$-3 \leq x < -2$	$\{3\}$
$-2 \leq x < 0$	$\{2,3\}$
$0 \leq x < 1$	$\{2,3,4\}$
$1 \leq x \leq 2$	$\{2,3,4,6\}$
$2 < x < 3$	$\{2,3,4,5,6\}$
$3 \leq x < 4$	$\{2,4,5,6\}$
$4 \leq x \leq 5$	$\{4,5,6\}$
$5 < x < 6$	$\{5,6\}$
$6 \leq x \leq 7$	$\{1,5,6\}$
$7 < x \leq 8$	$\{1,6\}$
$8 < x \leq 9$	$\{6\}$
$9 < x$	$\emptyset$

Alterações ocorrem
nos extremos dos segmentos.

Descrição combinatória da linha

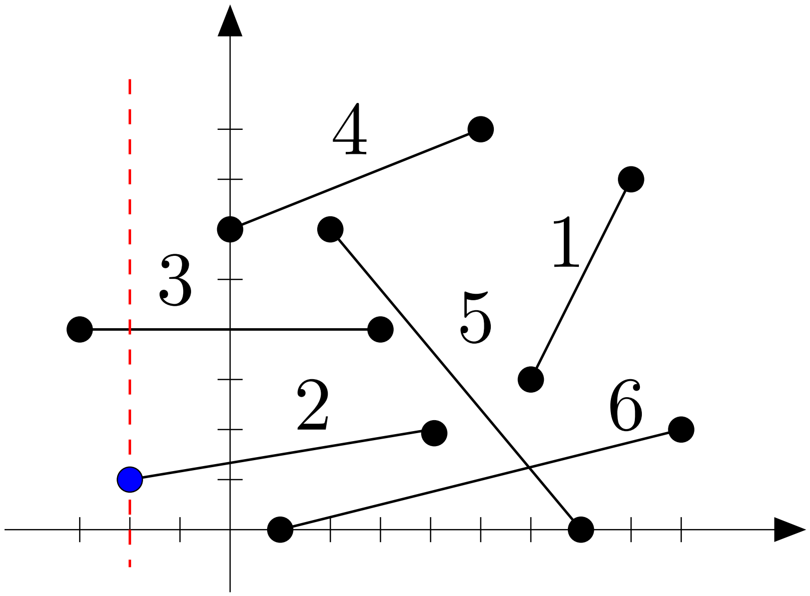


$x < -3$	$\emptyset$
$-3 \leq x < -2$	$\{3\}$
$-2 \leq x < 0$	$\{2, 3\}$
$0 \leq x < 1$	$\{2, 3, 4\}$
$1 \leq x \leq 2$	$\{2, 3, 4, 6\}$
$2 < x < 3$	$\{2, 3, 4, 5, 6\}$
$3 \leq x < 4$	$\{2, 4, 5, 6\}$
$4 \leq x \leq 5$	$\{4, 5, 6\}$
$5 < x < 6$	$\{5, 6\}$
$6 \leq x \leq 7$	$\{1, 5, 6\}$
$7 < x \leq 8$	$\{1, 6\}$
$8 < x \leq 9$	$\{6\}$
$9 < x$	$\emptyset$

Alterações ocorrem
nos extremos dos segmentos.

Estes são
os pontos eventos.

Descrição combinatória da linha

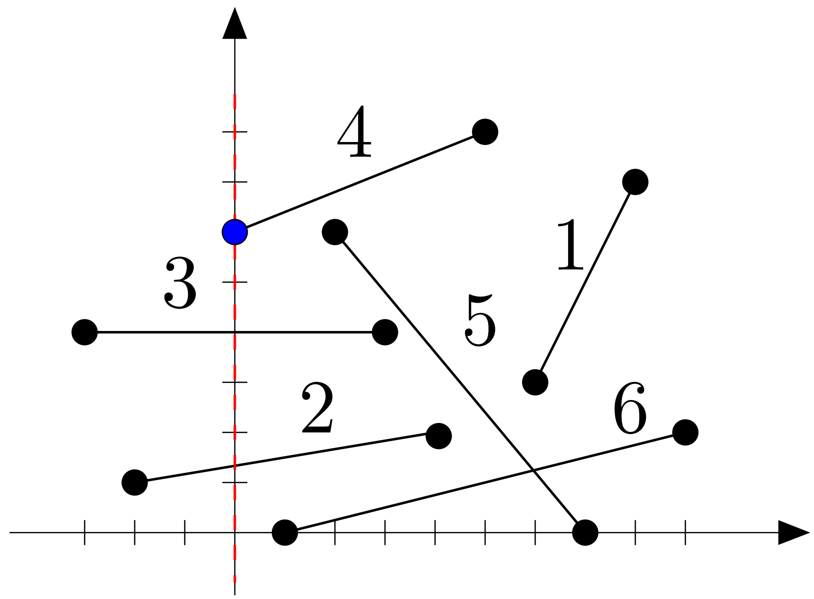


$x < -3$	$\emptyset$
$-3 \leq x < -2$	$\{3\}$
$-2 \leq x < 0$	$\{2, 3\}$
$0 \leq x < 1$	$\{2, 3, 4\}$
$1 \leq x \leq 2$	$\{2, 3, 4, 6\}$
$2 < x < 3$	$\{2, 3, 4, 5, 6\}$
$3 \leq x < 4$	$\{2, 4, 5, 6\}$
$4 \leq x \leq 5$	$\{4, 5, 6\}$
$5 < x < 6$	$\{5, 6\}$
$6 \leq x \leq 7$	$\{1, 5, 6\}$
$7 < x \leq 8$	$\{1, 6\}$
$8 < x \leq 9$	$\{6\}$
$9 < x$	$\emptyset$

Alterações ocorrem nos extremos dos segmentos.

Estes são os pontos eventos.

Descrição combinatória da linha

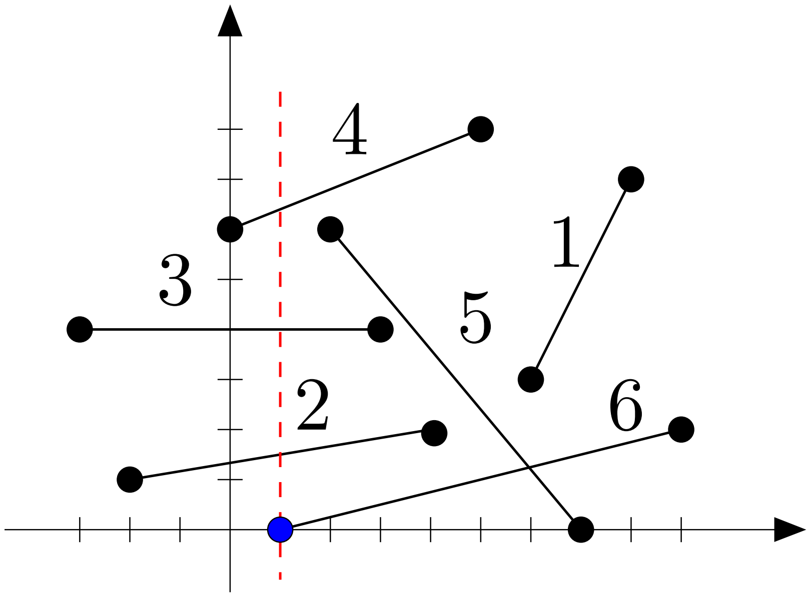


$x < -3$	$\emptyset$
$-3 \leq x < -2$	$\{3\}$
$-2 \leq x < 0$	$\{2, 3\}$
$0 \leq x < 1$	$\{2, 3, 4\}$
$1 \leq x \leq 2$	$\{2, 3, 4, 6\}$
$2 < x < 3$	$\{2, 3, 4, 5, 6\}$
$3 \leq x < 4$	$\{2, 4, 5, 6\}$
$4 \leq x \leq 5$	$\{4, 5, 6\}$
$5 < x < 6$	$\{5, 6\}$
$6 \leq x \leq 7$	$\{1, 5, 6\}$
$7 < x \leq 8$	$\{1, 6\}$
$8 < x \leq 9$	$\{6\}$
$9 < x$	$\emptyset$

Alterações ocorrem
nos extremos dos segmentos.

Estes são
os pontos eventos.

Descrição combinatória da linha

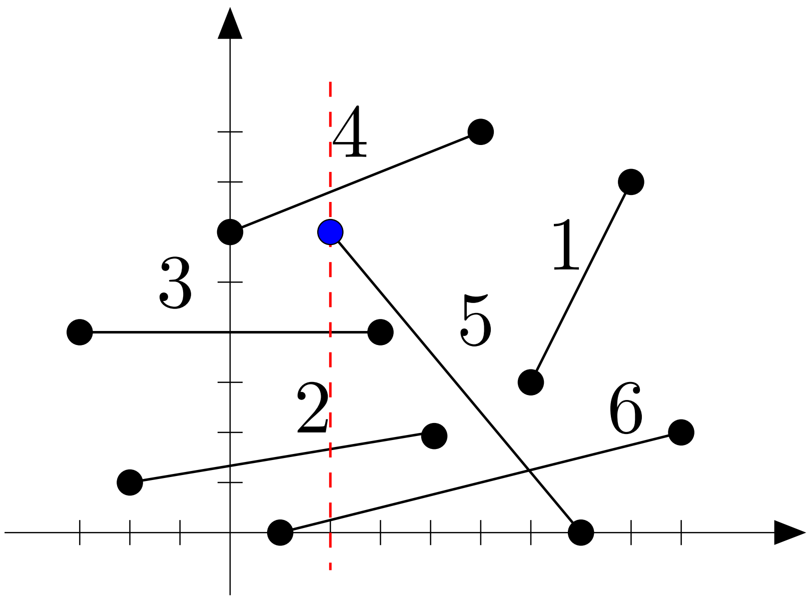


$x < -3$	$\emptyset$
$-3 \le x < -2$	$\{3\}$
$-2 \le x < 0$	$\{2,3\}$
$0 \le x < 1$	$\{2,3,4\}$
$1 \le x \le 2$	$\{2,3,4,6\}$
$2 < x < 3$	$\{2,3,4,5,6\}$
$3 \le x < 4$	$\{2,4,5,6\}$
$4 \le x \le 5$	$\{4,5,6\}$
$5 < x < 6$	$\{5,6\}$
$6 \le x \le 7$	$\{1,5,6\}$
$7 < x \le 8$	$\{1,6\}$
$8 < x \le 9$	$\{6\}$
$9 < x$	$\emptyset$

Alterações ocorrem nos extremos dos segmentos.

Estes são os pontos eventos.

Descrição combinatória da linha

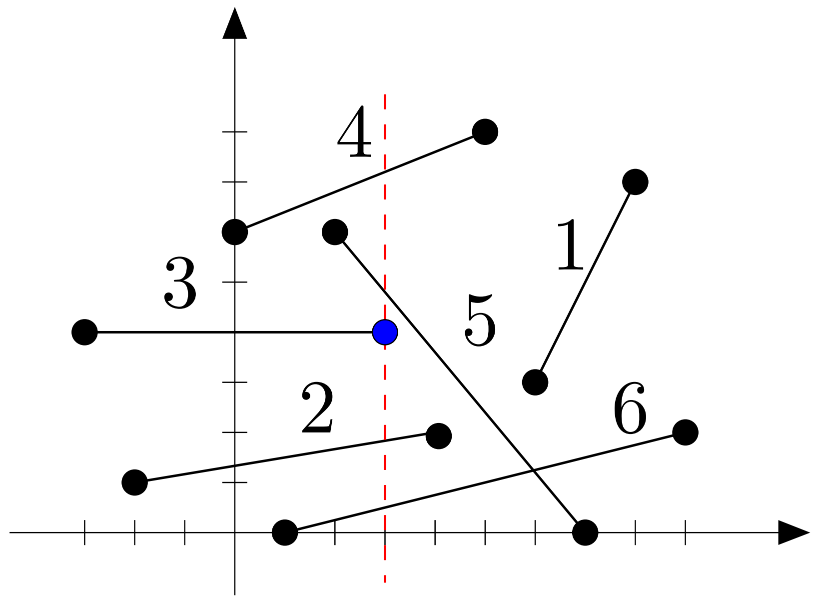


$x < -3$	$\emptyset$
$-3 \leq x < -2$	$\{3\}$
$-2 \leq x < 0$	$\{2, 3\}$
$0 \leq x < 1$	$\{2, 3, 4\}$
$1 \leq x \leq 2$	$\{2, 3, 4, 6\}$
$2 < x < 3$	$\{2, 3, 4, 5, 6\}$
$3 \leq x < 4$	$\{2, 4, 5, 6\}$
$4 \leq x \leq 5$	$\{4, 5, 6\}$
$5 < x < 6$	$\{5, 6\}$
$6 \leq x \leq 7$	$\{1, 5, 6\}$
$7 < x \leq 8$	$\{1, 6\}$
$8 < x \leq 9$	$\{6\}$
$9 < x$	$\emptyset$

Alterações ocorrem
nos extremos dos segmentos.

Estes são
os pontos eventos.

Descrição combinatória da linha

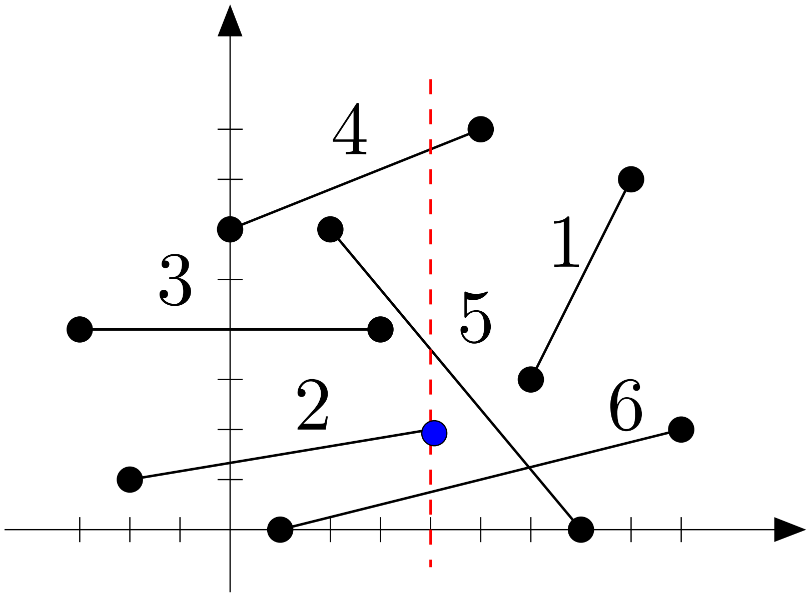


$x < -3$	$\emptyset$
$-3 \leq x < -2$	$\{3\}$
$-2 \leq x < 0$	$\{2, 3\}$
$0 \leq x < 1$	$\{2, 3, 4\}$
$1 \leq x \leq 2$	$\{2, 3, 4, 6\}$
$2 < x < 3$	$\{2, 3, 4, 5, 6\}$
$3 \leq x < 4$	$\{2, 4, 5, 6\}$
$4 \leq x \leq 5$	$\{4, 5, 6\}$
$5 < x < 6$	$\{5, 6\}$
$6 \leq x \leq 7$	$\{1, 5, 6\}$
$7 < x \leq 8$	$\{1, 6\}$
$8 < x \leq 9$	$\{6\}$
$9 < x$	$\emptyset$

Alterações ocorrem nos extremos dos segmentos.

Estes são os pontos eventos.

Descrição combinatória da linha

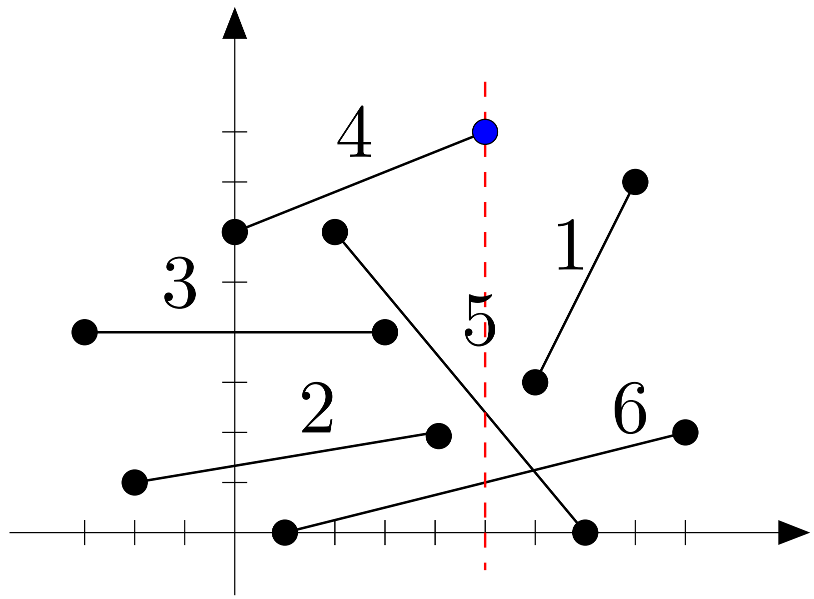


$x < -3$	$\emptyset$
$-3 \leq x < -2$	$\{3\}$
$-2 \leq x < 0$	$\{2,3\}$
$0 \leq x < 1$	$\{2,3,4\}$
$1 \leq x \leq 2$	$\{2,3,4,6\}$
$2 < x < 3$	$\{2,3,4,5,6\}$
$3 \leq x < 4$	$\{2,4,5,6\}$
$4 \leq x \leq 5$	$\{4,5,6\}$
$5 < x < 6$	$\{5,6\}$
$6 \leq x \leq 7$	$\{1,5,6\}$
$7 < x \leq 8$	$\{1,6\}$
$8 < x \leq 9$	$\{6\}$
$9 < x$	$\emptyset$

Alterações ocorrem
nos extremos dos segmentos.

Estes são
os pontos eventos.

Descrição combinatória da linha

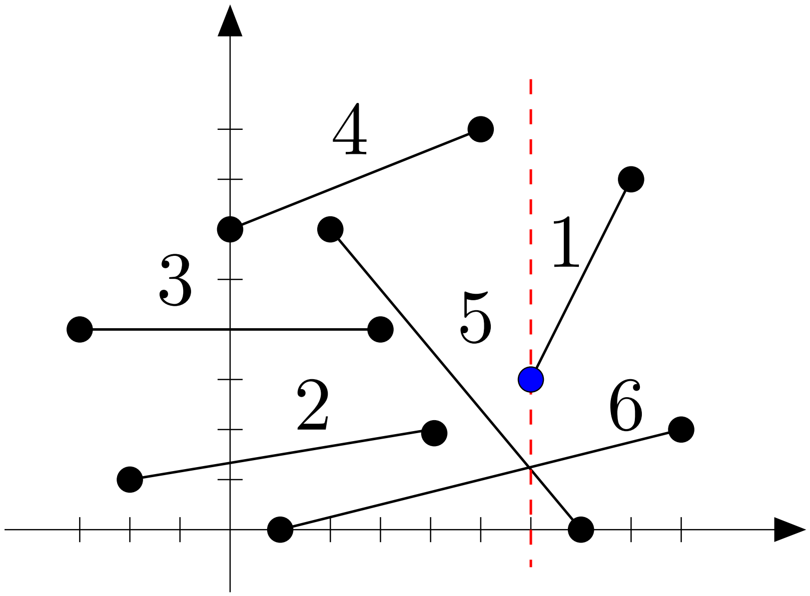


Alterações ocorrem
nos extremos dos segmentos.

Estes são
os pontos eventos.

$x < -3$	$\emptyset$
$-3 \leq x < -2$	$\{3\}$
$-2 \leq x < 0$	$\{2, 3\}$
$0 \leq x < 1$	$\{2, 3, 4\}$
$1 \leq x \leq 2$	$\{2, 3, 4, 6\}$
$2 < x < 3$	$\{2, 3, 4, 5, 6\}$
$3 \leq x < 4$	$\{2, 4, 5, 6\}$
$4 \leq x \leq 5$	$\{4, 5, 6\}$
$5 < x < 6$	$\{5, 6\}$
$6 \leq x \leq 7$	$\{1, 5, 6\}$
$7 < x \leq 8$	$\{1, 6\}$
$8 < x \leq 9$	$\{6\}$
$9 < x$	$\emptyset$

Descrição combinatória da linha

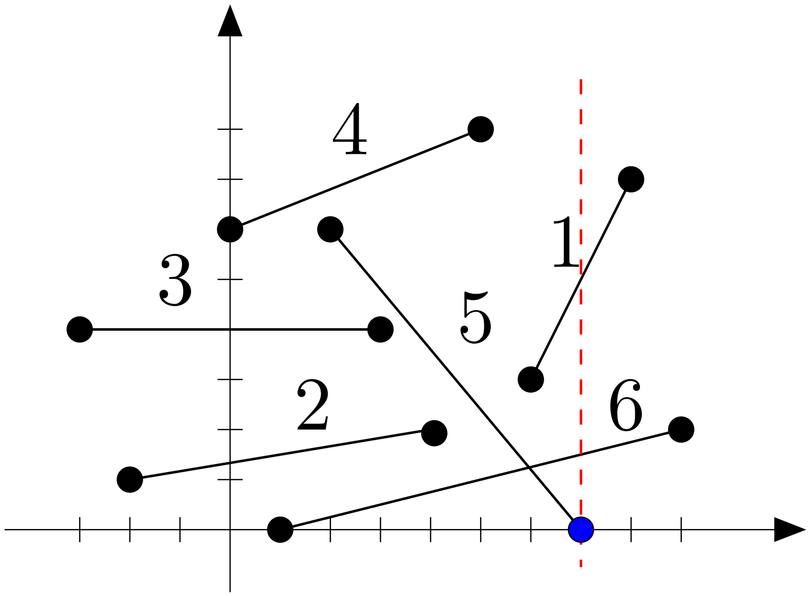


Alterações ocorrem
nos extremos dos segmentos.

Estes são
os pontos eventos.

$x < -3$	$\emptyset$
$-3 \leq x < -2$	$\{3\}$
$-2 \leq x < 0$	$\{2,3\}$
$0 \leq x < 1$	$\{2,3,4\}$
$1 \leq x \leq 2$	$\{2,3,4,6\}$
$2 < x < 3$	$\{2,3,4,5,6\}$
$3 \leq x < 4$	$\{2,4,5,6\}$
$4 \leq x \leq 5$	$\{4,5,6\}$
$5 < x < 6$	$\{5,6\}$
$6 \leq x \leq 7$	$\{1,5,6\}$
$7 < x \leq 8$	$\{1,6\}$
$8 < x \leq 9$	$\{6\}$
$9 < x$	$\emptyset$

Descrição combinatória da linha

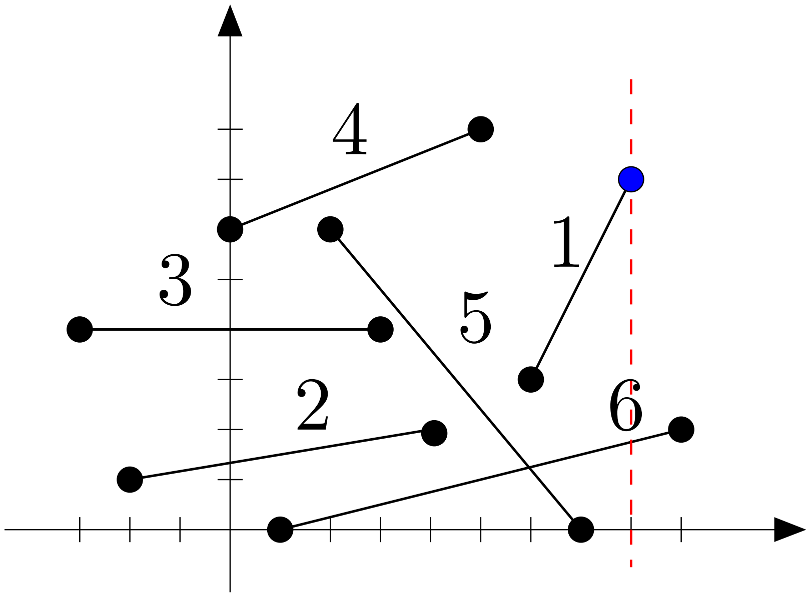


$x < -3$	$\emptyset$
$-3 \leq x < -2$	$\{3\}$
$-2 \leq x < 0$	$\{2, 3\}$
$0 \leq x < 1$	$\{2, 3, 4\}$
$1 \leq x \leq 2$	$\{2, 3, 4, 6\}$
$2 < x < 3$	$\{2, 3, 4, 5, 6\}$
$3 \leq x < 4$	$\{2, 4, 5, 6\}$
$4 \leq x \leq 5$	$\{4, 5, 6\}$
$5 < x < 6$	$\{5, 6\}$
$6 \leq x \leq 7$	$\{1, 5, 6\}$
$7 < x \leq 8$	$\{1, 6\}$
$8 < x \leq 9$	$\{6\}$
$9 < x$	$\emptyset$

Alterações ocorrem
nos extremos dos segmentos.

Estes são
os pontos eventos.

Descrição combinatória da linha

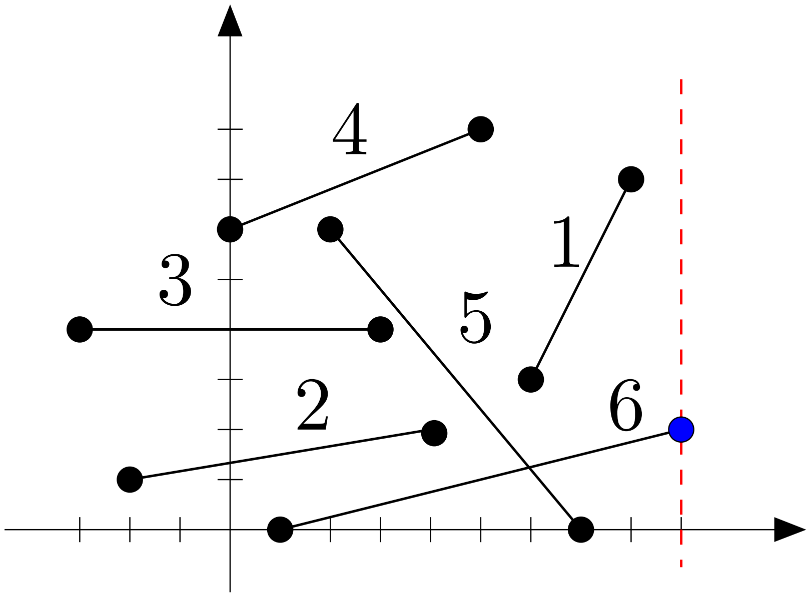


Alterações ocorrem
nos extremos dos segmentos.

Estes são
os pontos eventos.

$x < -3$	$\emptyset$
$-3 \leq x < -2$	$\{3\}$
$-2 \leq x < 0$	$\{2,3\}$
$0 \leq x < 1$	$\{2,3,4\}$
$1 \leq x \leq 2$	$\{2,3,4,6\}$
$2 < x < 3$	$\{2,3,4,5,6\}$
$3 \leq x < 4$	$\{2,4,5,6\}$
$4 \leq x \leq 5$	$\{4,5,6\}$
$5 < x < 6$	$\{5,6\}$
$6 \leq x \leq 7$	$\{1,5,6\}$
$7 < x \leq 8$	$\{1,6\}$
$8 < x \leq 9$	$\{6\}$
$9 < x$	$\emptyset$

Descrição combinatória da linha

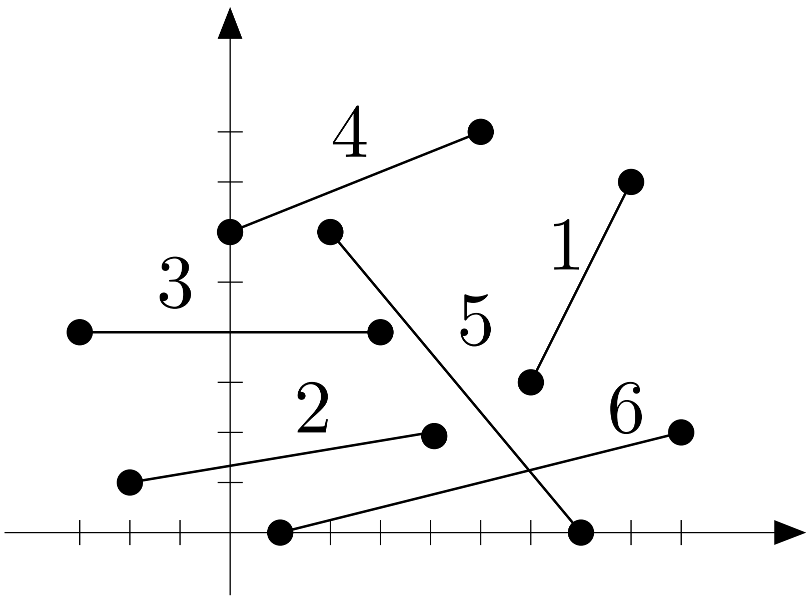


$x < -3$	$\emptyset$
$-3 \leq x < -2$	$\{3\}$
$-2 \leq x < 0$	$\{2, 3\}$
$0 \leq x < 1$	$\{2, 3, 4\}$
$1 \leq x \leq 2$	$\{2, 3, 4, 6\}$
$2 < x < 3$	$\{2, 3, 4, 5, 6\}$
$3 \leq x < 4$	$\{2, 4, 5, 6\}$
$4 \leq x \leq 5$	$\{4, 5, 6\}$
$5 < x < 6$	$\{5, 6\}$
$6 \leq x \leq 7$	$\{1, 5, 6\}$
$7 < x \leq 8$	$\{1, 6\}$
$8 < x \leq 9$	$\{6\}$
$9 < x$	$\emptyset$

Alterações ocorrem
nos extremos dos segmentos.

Estes são
os pontos eventos.

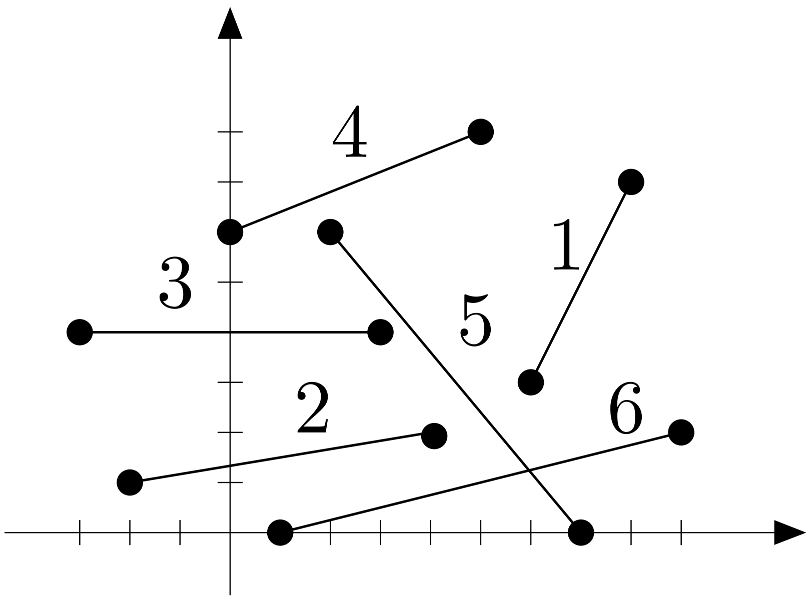
Descrição combinatória da linha



Como guardar
um destes conjuntos?

$x < -3$	$\emptyset$
$-3 \leq x < -2$	$\{3\}$
$-2 \leq x < 0$	$\{2, 3\}$
$0 \leq x < 1$	$\{2, 3, 4\}$
$1 \leq x \leq 2$	$\{2, 3, 4, 6\}$
$2 < x < 3$	$\{2, 3, 4, 5, 6\}$
$3 \leq x < 4$	$\{2, 4, 5, 6\}$
$4 \leq x \leq 5$	$\{4, 5, 6\}$
$5 < x < 6$	$\{5, 6\}$
$6 \leq x \leq 7$	$\{1, 5, 6\}$
$7 < x \leq 8$	$\{1, 6\}$
$8 < x \leq 9$	$\{6\}$
$9 < x$	$\emptyset$

Descrição combinatória da linha



$x < -3$	$\emptyset$
$-3 \leq x < -2$	$\{3\}$
$-2 \leq x < 0$	$\{2, 3\}$
$0 \leq x < 1$	$\{2, 3, 4\}$
$1 \leq x \leq 2$	$\{2, 3, 4, 6\}$
$2 < x < 3$	$\{2, 3, 4, 5, 6\}$
$3 \leq x < 4$	$\{2, 4, 5, 6\}$
$4 \leq x \leq 5$	$\{4, 5, 6\}$
$5 < x < 6$	$\{5, 6\}$
$6 \leq x \leq 7$	$\{1, 5, 6\}$
$7 < x \leq 8$	$\{1, 6\}$
$8 < x \leq 9$	$\{6\}$
$9 < x$	$\emptyset$

Como guardar
um destes conjuntos?

Que operações ele sofre?

Descrição combinatória da linha

O conjunto dos segmentos na linha sofre inserções e remoções.

Descrição combinatória da linha

O conjunto dos segmentos na linha sofre
inserções e remoções.

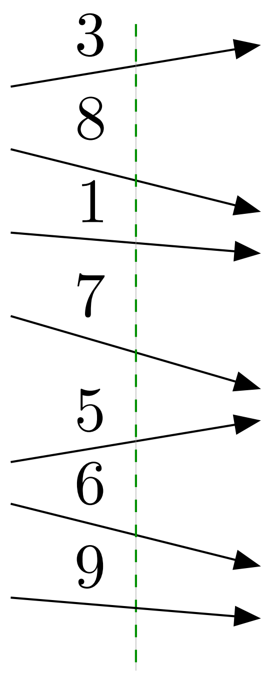
Como a linha vai nos ajudar a detectar interseção?

Descrição combinatória da linha

O conjunto dos segmentos na linha sofre
inserções e remoções.

Como a linha vai nos ajudar a detectar interseção?

Ideia: testar interseção
apenas entre segmentos
"vizinhos na linha".



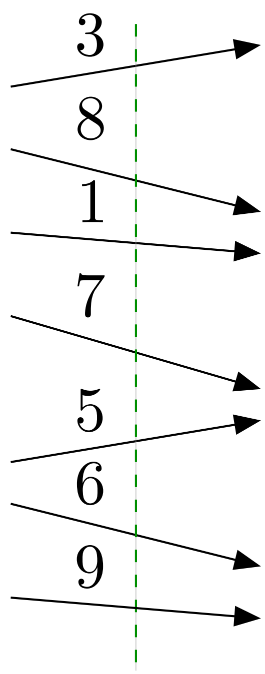
Descrição combinatória da linha

O conjunto dos segmentos na linha sofre inserções e remoções.

Como a linha vai nos ajudar a detectar interseção?

Ideia: testar interseção apenas entre segmentos “vizinhos na linha”.

Para isso, mantemos os segmentos na linha ordenados.



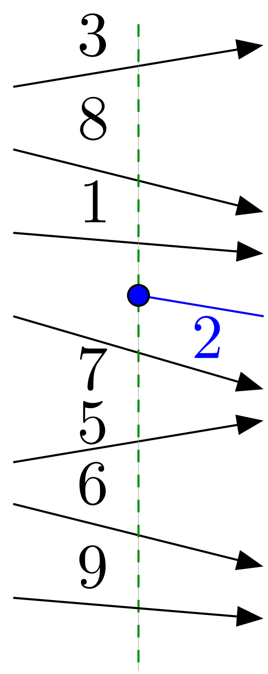
Descrição combinatória da linha

Os segmentos ficam na ordem em que intersectam a linha.

Descrição combinatória da linha

Os segmentos ficam na ordem em que intersectam a linha.

$$3 \prec 8 \prec 1 \prec 2 \prec 7 \prec 5 \prec 6 \prec 9$$

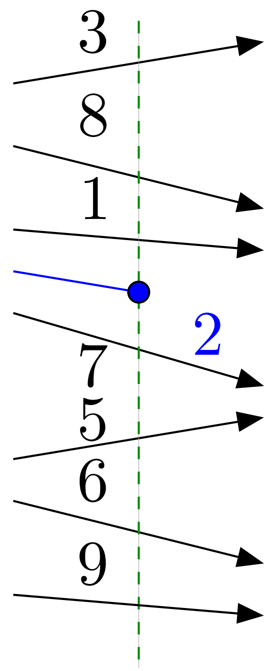


Ao inserimos um segmento, testamos a interseção dele com seu predecessor e com seu sucessor na ordem.

Descrição combinatória da linha

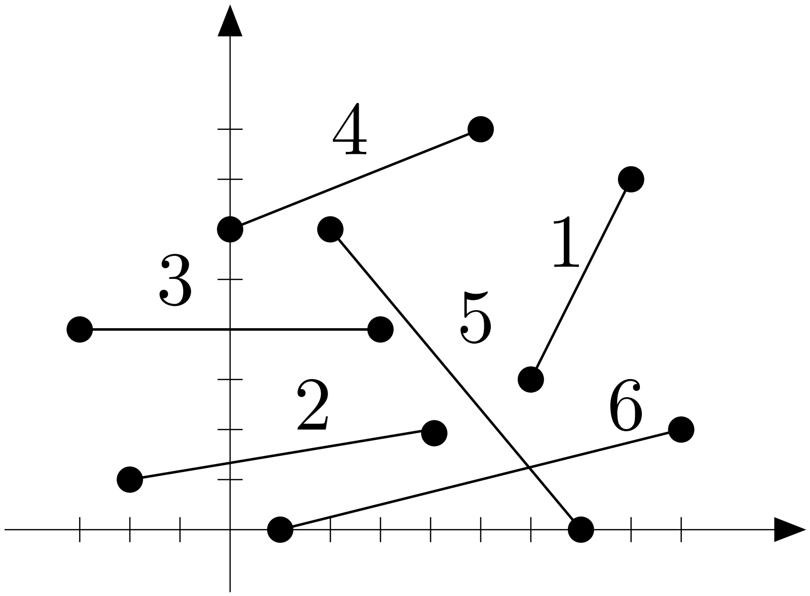
Os segmentos ficam na ordem em que intersectam a linha.

$$3 \prec 8 \prec 1 \prec 2 \prec 7 \prec 5 \prec 6 \prec 9$$



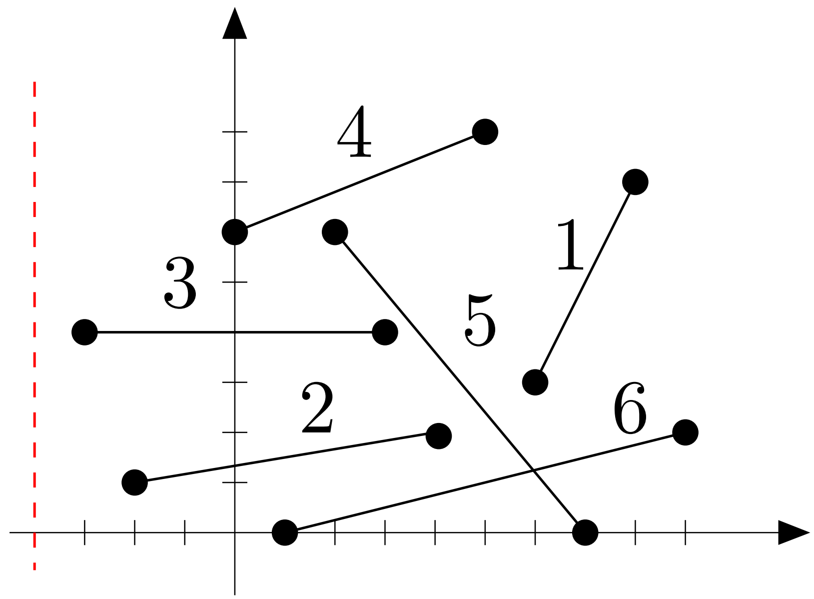
Ao removermos um segmento, testamos a interseção de seu predecessor e com seu sucessor na ordem.

Algoritmo de Shamos e Hoey



$-\infty$	
-3	3
-2	$3 \prec 2$
0	$4 \prec 3 \prec 2$
1	$4 \prec 3 \prec 2 \prec 6$
2	$4 \prec 5 \prec 3 \prec 2 \prec 6$
3	$4 \prec 5 \prec 2 \prec 6$
4	$4 \prec 5 \prec 6$
5	$\cdots$
6	
7	
8	

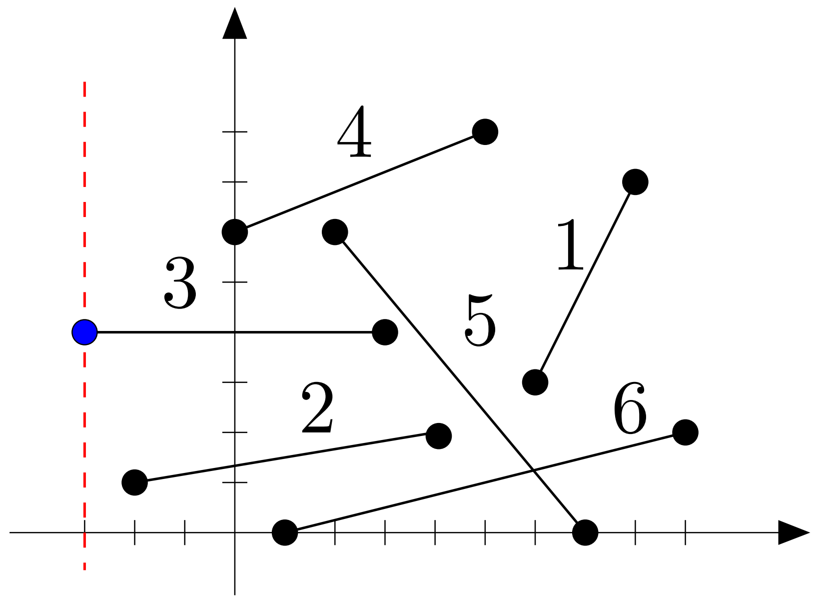
Algoritmo de Shamos e Hoey



$-\infty$	
-3	3
-2	$3 \prec 2$
0	$4 \prec 3 \prec 2$
1	$4 \prec 3 \prec 2 \prec 6$
2	$4 \prec 5 \prec 3 \prec 2 \prec 6$
3	$4 \prec 5 \prec 2 \prec 6$
4	$4 \prec 5 \prec 6$
5	
6	
7	
8	

Alterações ocorrem
nos extremos dos segmentos.

Algoritmo de Shamos e Hoey

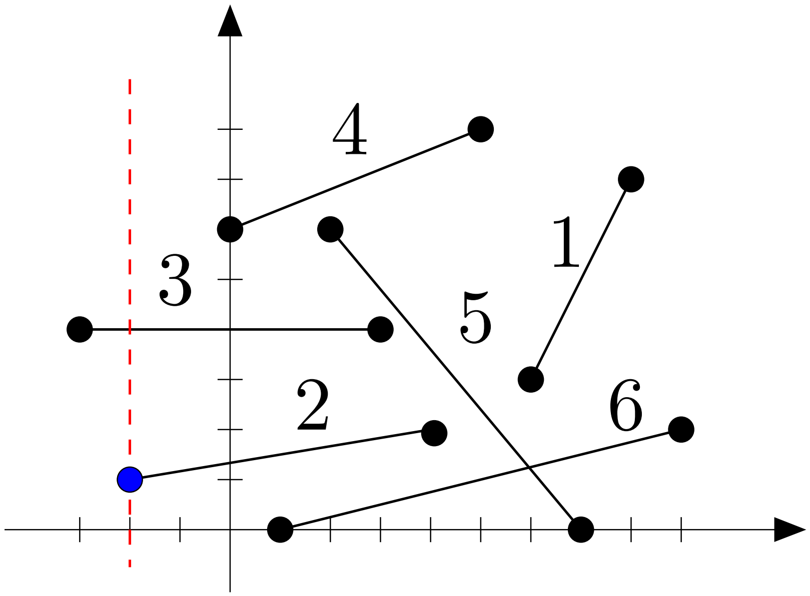


$-\infty$	
-3	3
-2	$3 \prec 2$
0	$4 \prec 3 \prec 2$
1	$4 \prec 3 \prec 2 \prec 6$
2	$4 \prec 5 \prec 3 \prec 2 \prec 6$
3	$4 \prec 5 \prec 2 \prec 6$
4	$4 \prec 5 \prec 6$
5	
6	
7	
8	

Alterações ocorrem
nos extremos dos segmentos.

Estes são
os pontos eventos.

Algoritmo de Shamos e Hoey

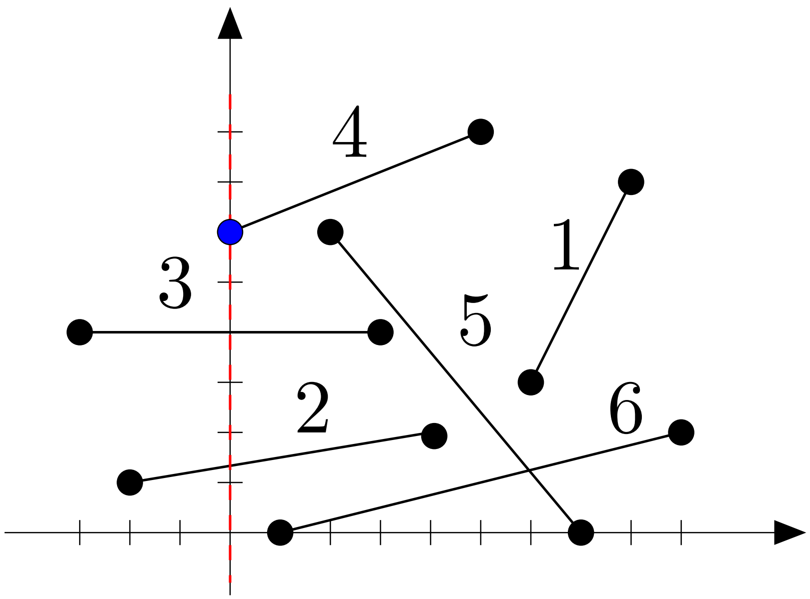


$-\infty$	
-3	3
-2	$3 \prec 2$
0	$4 \prec 3 \prec 2$
1	$4 \prec 3 \prec 2 \prec 6$
2	$4 \prec 5 \prec 3 \prec 2 \prec 6$
3	$4 \prec 5 \prec 2 \prec 6$
4	$4 \prec 5 \prec 6$
5	$5 \prec 6$
6	$\cdots$
7	
8	

Alterações ocorrem
nos extremos dos segmentos.

Estes são
os pontos eventos.

Algoritmo de Shamos e Hoey

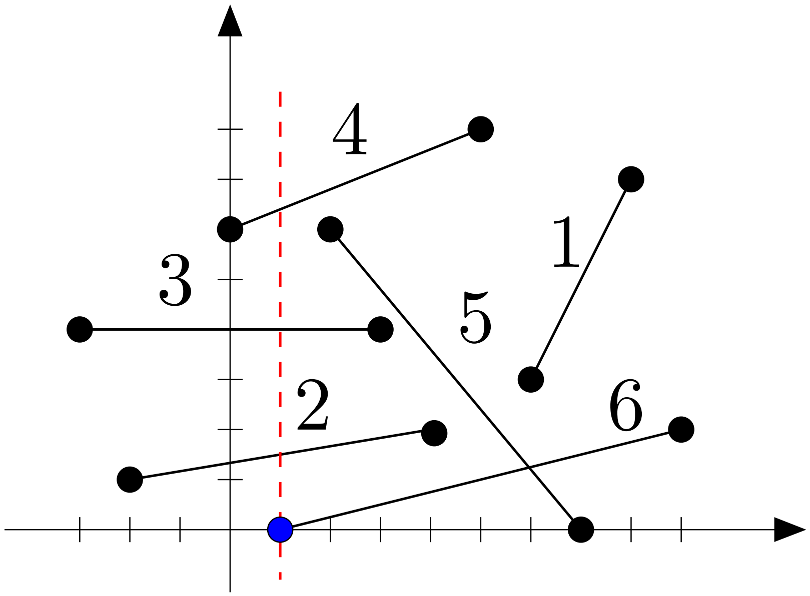


$-\infty$	
-3	3
-2	$3 \prec 2$
0	$4 \prec 3 \prec 2$
1	$4 \prec 3 \prec 2 \prec 6$
2	$4 \prec 5 \prec 3 \prec 2 \prec 6$
3	$4 \prec 5 \prec 2 \prec 6$
4	$4 \prec 5 \prec 6$
5	
6	
7	
8	

Alterações ocorrem
nos extremos dos segmentos.

Estes são
os pontos eventos.

Algoritmo de Shamos e Hoey

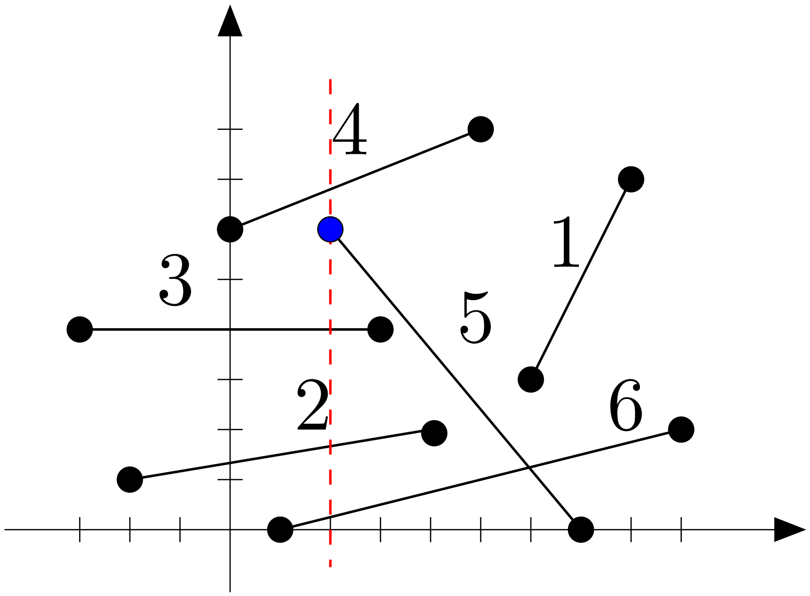


$-\infty$	
-3	3
-2	$3 \prec 2$
0	$4 \prec 3 \prec 2$
1	$4 \prec 3 \prec 2 \prec 6$
2	$4 \prec 5 \prec 3 \prec 2 \prec 6$
3	$4 \prec 5 \prec 2 \prec 6$
4	$4 \prec 5 \prec 6$
5	
6	
7	
8	

Alterações ocorrem nos extremos dos segmentos.

Estes são os pontos eventos.

Algoritmo de Shamos e Hoey

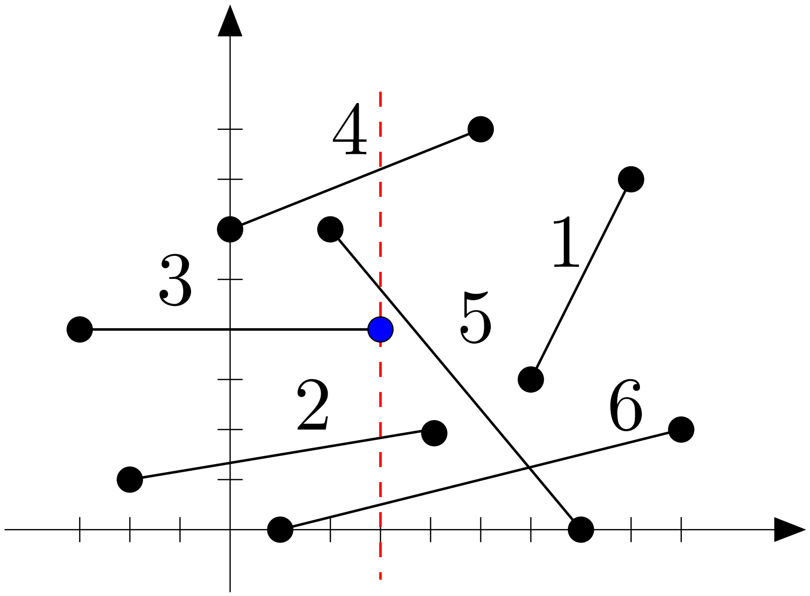


$-\infty$	
-3	3
-2	$3 \prec 2$
0	$4 \prec 3 \prec 2$
1	$4 \prec 3 \prec 2 \prec 6$
2	$4 \prec 5 \prec 3 \prec 2 \prec 6$
3	$4 \prec 5 \prec 2 \prec 6$
4	$4 \prec 5 \prec 6$
5	
6	
7	
8	

Alterações ocorrem
nos extremos dos segmentos.

Estes são
os pontos eventos.

Algoritmo de Shamos e Hoey

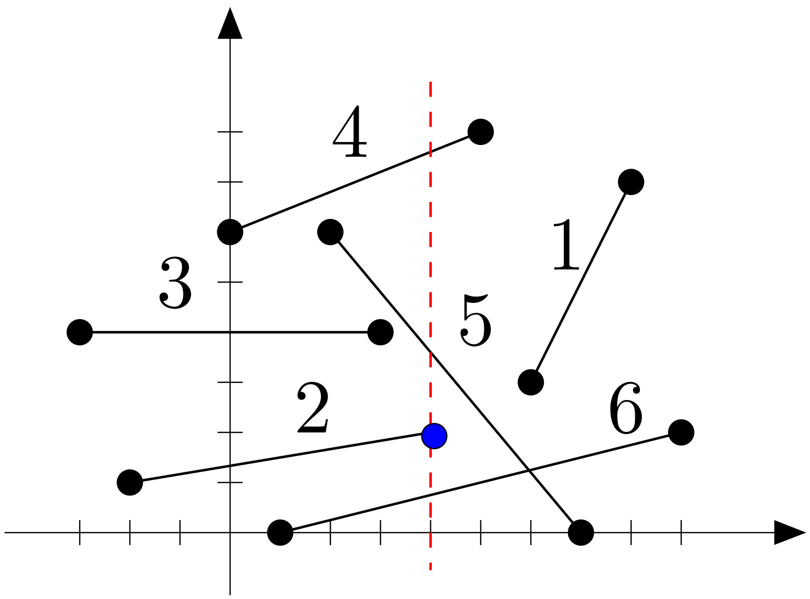


$-\infty$	
-3	3
-2	$3 \prec 2$
0	$4 \prec 3 \prec 2$
1	$4 \prec 3 \prec 2 \prec 6$
2	$4 \prec 5 \prec 3 \prec 2 \prec 6$
3	$4 \prec 5 \prec 2 \prec 6$
4	$4 \prec 5 \prec 6$
5	
6	
7	
8	

Alterações ocorrem
nos extremos dos segmentos.

Estes são
os pontos eventos.

Algoritmo de Shamos e Hoey

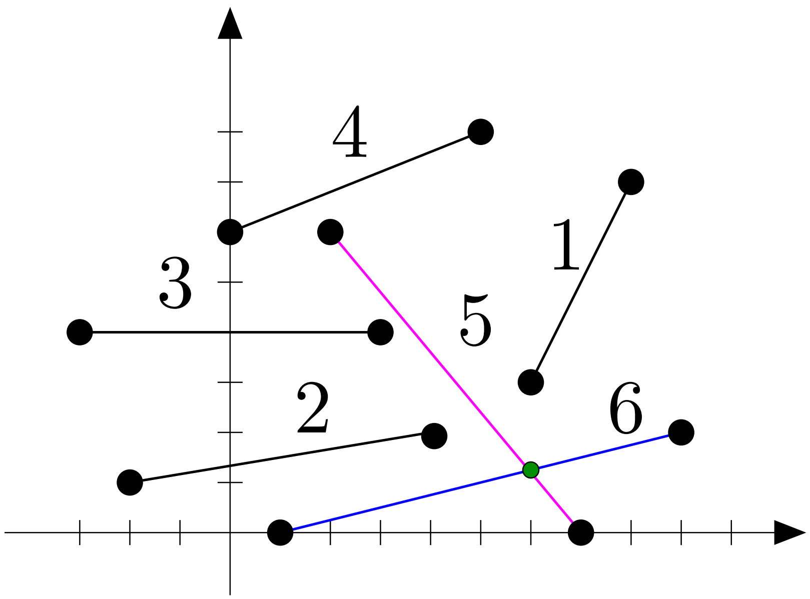


$-\infty$	
-3	3
-2	$3 \prec 2$
0	$4 \prec 3 \prec 2$
1	$4 \prec 3 \prec 2 \prec 6$
2	$4 \prec 5 \prec 3 \prec 2 \prec 6$
3	$4 \prec 5 \prec 2 \prec 6$
4	$4 \prec 5 \prec 6$
5	
6	
7	
8	

Alterações ocorrem
nos extremos dos segmentos.

Estes são
os pontos eventos.

Algoritmo de Shamos e Hoey



$-\infty$	
-3	3
-2	$3 \prec 2$
0	$4 \prec 3 \prec 2$
1	$4 \prec 3 \prec 2 \prec 6$
2	$4 \prec 5 \prec 3 \prec 2 \prec 6$
3	$4 \prec 5 \prec 2 \prec 6$
4	$4 \prec 5 \prec 6$
5	
6	
7	
8	

Alterações ocorrem
nos extremos dos segmentos.

Estes são
os pontos eventos.

Encontrou uma interseção!