

Diagrama de Voronoi

TRATA-EVENTO-PONTO (q, Q, T, Vor)

```
1  se  $T = \emptyset$ 
2    então insere  $q$  na ABBB  $T$ 
3    senão seja  $\alpha$  o arco em  $T$  acima de  $q$ 
4        seja  $p$  o ponto de  $P$  associado a  $\alpha$ 
5        se evento-circulo $[\alpha] \neq \text{NIL}$ 
6            então REMOVE(evento-circulo( $\alpha$ ),  $Q$ )
7            troca( $T, q, p$ )
8            acrescente uma nova aresta a  $V$ 
9            atualiza-eventos( $q, Q, T$ )
```

Há uma face para cada célula de $Vor(P)$, ou seja, as faces são associadas aos pontos de P .

A nova aresta inserida em $Vor(P)$ na linha 8 separa a face $V(p)$ da face $V(q)$.

Diagrama de Voronoi

troca(T, q, p):

Substitui a folha de T contendo q
por uma árvore com três folhas:

a do meio com q (representando o novo arco),
e as outras duas com p (representando os dois
trechos do arco α em que α foi “quebrado”
pela inserção do novo arco).

Os dois nós internos dessa árvore com três folhas
contém (p, q) e (q, p) ,

onde p é o ponto de P correspondente ao arco α .

Diagrama de Voronoi

atualiza-eventos(q, Q, T):

Determina os eventos-círculos das novas triplas de arcos consecutivos em T , envolvendo o novo arco associado a q

Para os novos eventos-círculos que estiverem abaixo da linha de varredura ajuste o campo *evento-circulo* dos arcos correspondentes e os insira em Q

Obs: O campo *evento-circulo* $[\alpha]$ guarda um apontador para o evento-círculo associado a α e seus dois arcos adjacentes na linha da praia, se tal evento-círculo for posterior a linha de varredura; senão guarda NIL.

Diagrama de Voronoi

TRATA-EVENTO-CÍRCULO (q, Q, T, Vor)

- 1 seja α o arco correspondente ao evento-círculo q
- 2 remova q de T ,
ajustando o *evento-círculo* dos arcos adjacentes
- 3 adicione a V o vértice que fica
no centro do círculo associado a q
- 4 ponha o novo vértice como ponta das duas arestas
associadas aos pontos de quebra de α
e os arcos adjacentes
- 5 crie uma nova aresta, associada
ao ponto de quebra dos arcos adjacentes a α

Verifique que **TRATA-EVENTO-PONTO** e **TRATA-EVENTO-CÍRCULO** consomem tempo $O(\lg n)$, e portanto o algoritmo de Fortune consome tempo $O(n \lg n)$.