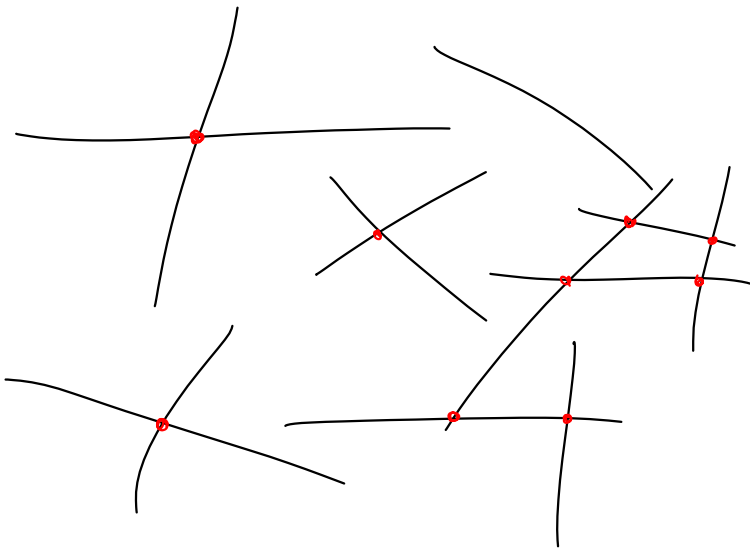


Line segment intersection reporting



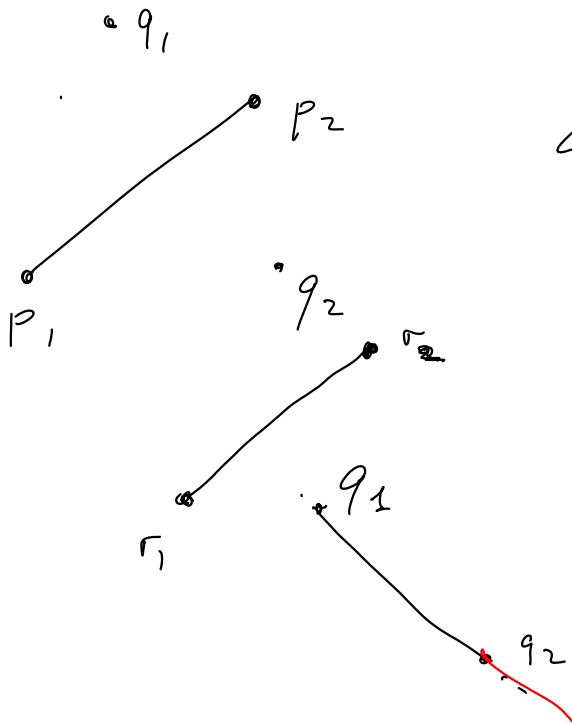
General position

Two line segments
 (p_1, p_2) & (q_1, q_2)
intersect iff

line $\overline{(p_1, p_2)}$ straddles

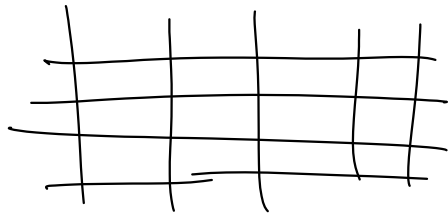
(q_1, q_2)

& line (q_1, q_2) straddles
 (p_1, p_2)



Trivial algorithm:

$O(n^2)$ [for every pair of line segments s_i, s_j
check if they intersect & compute
the coordinates of the intersection

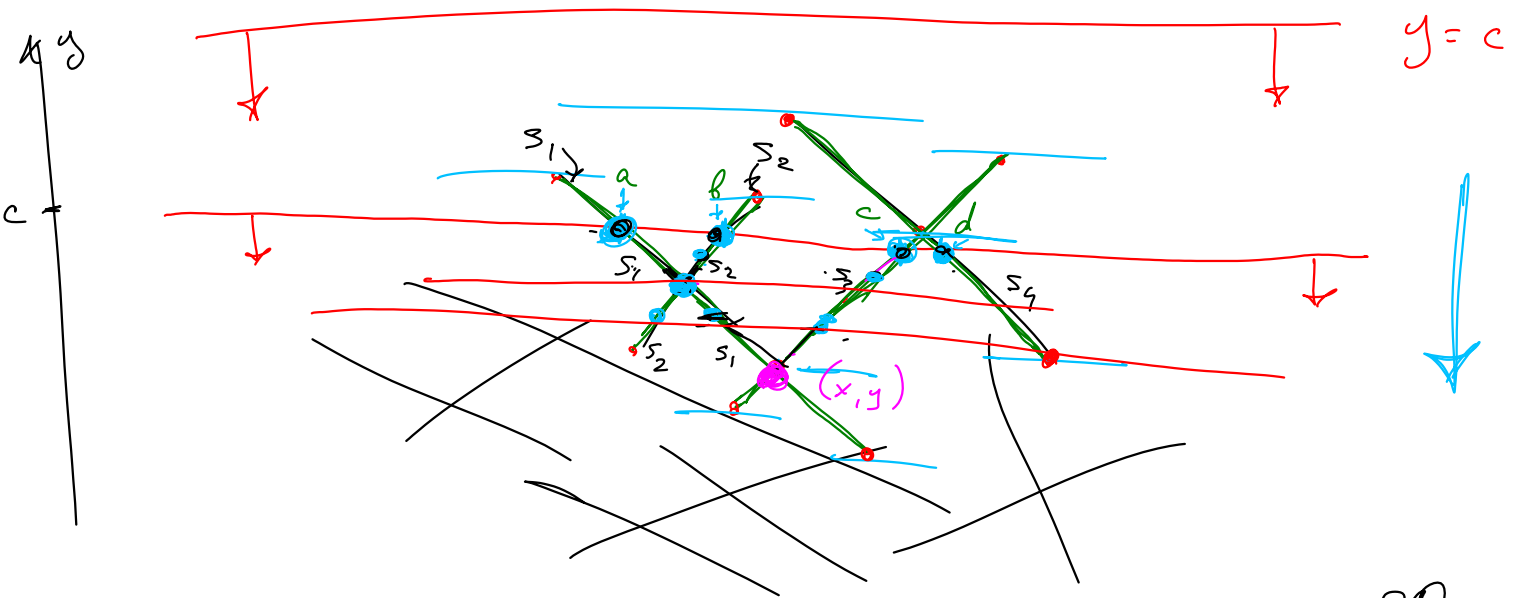


$$\frac{n}{2} \times \frac{n}{2} = \frac{n^2}{4} = O(n^2)$$

The best algorithm runs in $O(n \log n + k)$

\uparrow # of input segments \uparrow # of intersections
 \uparrow

Today: $O((n+k) \log n)$ -time algorithm



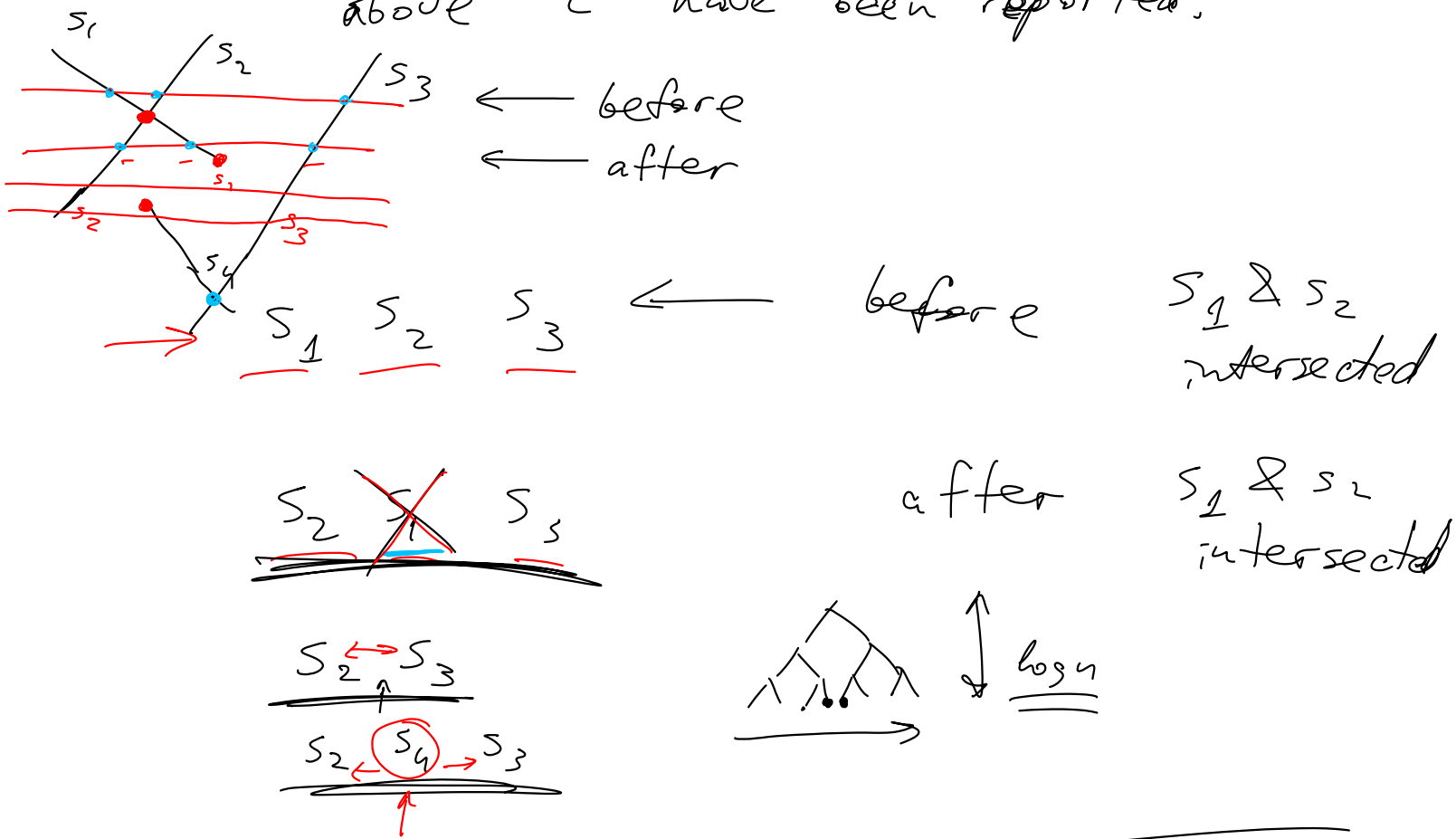
Events of interest:

- top endpoint of a segment
- bottom endpoint of a segment
- intersection of segments.

max-PQ, with keys that are y-coordinates of the events

Invariant: At a time when sweep line is defined

by equation $y = c$, all intersections above c have been reported.



→ Insert all endpoints into max-PQ Q by y -coordinate
 while Q is not empty
 $e = Q.\text{extractMax}()$ ← event of interest
 process Event(e)

process Event(e)

switch(e)
 case

top endpoint;

insert e into BST T by x -coord. of e

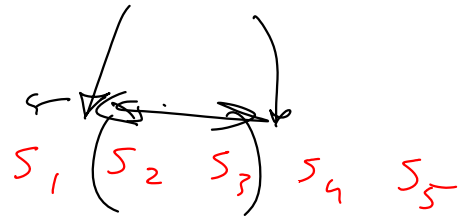
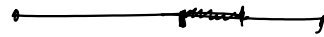
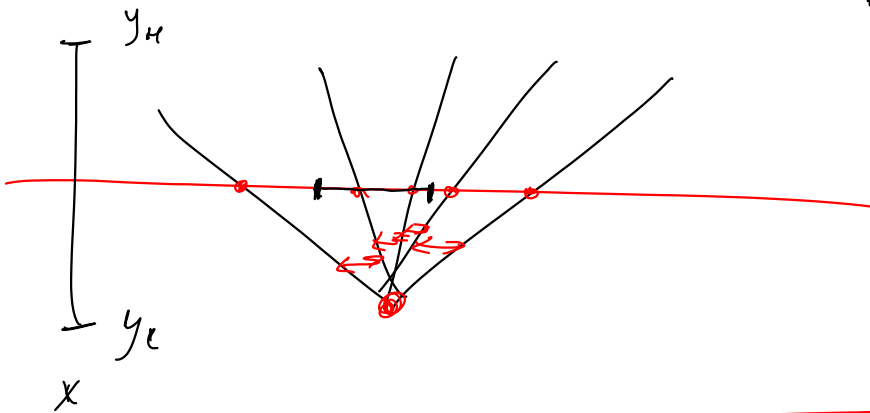
bottom endpoint:

remove e from BST T

intersection:

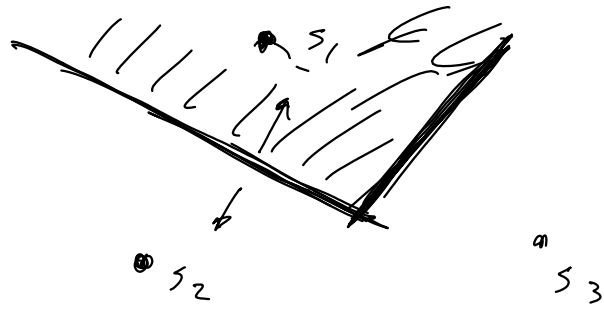
swap keys of segments that cause e

check: if newly neighboring segments cause intersections with their (new) neighbors below sweepline & if so, add it (intersection) event into max PQ Q .



Voronoi Diagrams

Post Office Problem



1

2

3

<http://www.raymondhill.net/voronoi/rhill-voronoi.html>