

Segment Intersection Sweep Line Algorithm

Main Module: Find Intersections

Algorithm: Find Intersections

Input: S : Set of line segments in a 2D plane.

Output: R : Set of intersection points within all segments of S , and for each of the points, the segments that intersect.

- 1 **Initialize** the Event Queue Q by inserting all end points (start and end) of all segments.
When an upper end point is inserted, its corresponding segment must be stored.
 - 2 **Initialize** the Sweep Line T , initially empty.
 - 3 **While** Q is not empty :
 - 4 | **Obtain and Delete** the next Event p from Q .
 - 5 | $processEvent(p)$
 - 6 **Return** R
-

Module: Process Event

Algorithm: Process Event

Input: Event p

- 1 Let $U(p)$ be the set of segments from S that have their upper end point at p . In case of a horizontal segment the upper end point is the left-most point.
 - 2 **Find** all segments in T that contain p ; they must be adjacent in T . Let $L(p)$ be a sub-set of segments found whose lower end point is p , and $C(p)$ a sub-set of segments that contain p within themselves.
 - 3 **if** $|L \cup U \cup C| > 1$ **then**
 - 4 | **Report** p as an intersection point with all segments of L, U, C .
 - 5 **end**
 - 6 **Delete** the segments $L \cup C$ from T .
 - 7 **Insert** the segments $U \cup C$ in T . The order in T must correspond to the order in which the segments intersect the Sweep Line just below p . If there is a horizontal segment, insert it at the end.
 - 8 **if** $U \cup C = \emptyset$ **then**
 - 9 | Let s_l and s_r the left and right neighbours of p over T , $findEvents(s_l, s_p, p)$.
 - 10 **end**
 - 11 **else**
 - 12 | Let s' be the left-most segment of $U \cup C$ in T .
 - 13 | Let s_l be the left neighbour of s' in T .
 - 14 | $findEvents(s_l, s', p)$.
 - 15 | Let s'' be the right-most segment of $U \cup C$ in T .
 - 16 | Let s_r be the right neighbour of s'' in T .
 - 17 | $findEvents(s'', s_r, p)$.
 - 18 **end**
-

Module: Find Events

Algorithm: Find Events

Input: s_l, s_r, p

```
1 if  $s_l$  and  $s_r$  intersect under the Sweep Line, or right at it but at the left side of current  
   Event  $p$ , and the intersection is not yet an Event in  $Q$  then  
2   | Insert the intersection point as a new event in  $Q$ .  
3 end
```
