CMPS 3130 Computational Geometry – Spring 20

2/13/20

5. Homework (undergrad)

Due **2/20/20** before class

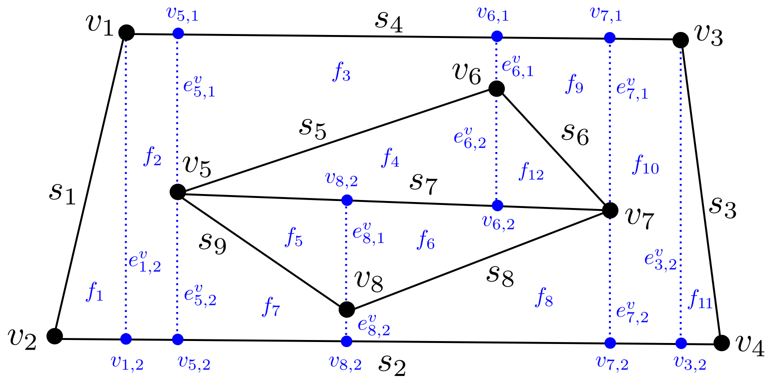
Please justify all your answers. Often it helps to draw pictures.

1. **Sweepline Trapezoidal Map (10 points)**
   For this problem we consider the sweepline algorithm below for constructing a trapezoidal map of n non-crossing line segments which runs in $O(n \log n)$ time. (Note, this algorithm is not randomized and does not compute the associated DAG for point location.) The goal is to construct a DCEL for the trapezoidal map, but the algorithm description is higher-level and refers to "constructing a trapezoid", which means the trapezoid and all its half-edges, vertices, and associated pointers are added to the DCEL.

   - **Input:** Set of line segments $S = \{s_1, \ldots, s_n\}$. All segments are non-vertical and do not intersect in the interior.
   - **Output:** A DCEL of the trapezoidal map of this set of line segments.
   - **Cleanliness property:** All trapezoids that lie entirely to the left of the sweep-line have been constructed.
   - **Sweep-line status:** (1) All segments intersecting the sweep line in order along the sweep-line, stored in a balanced binary search tree. (Exactly like in the line-segment intersection sweep.)
     (2) In addition, each interior node of the tree is considered to lie between two leaves/segments, and each node stores the partially constructed trapezoid that lies between those segments. (See example on the next page.)
   - **Events:** (a) The events are all segment endpoints.
     (b) The event queue is computed by sorting all the $2n$ endpoints in $O(n \log n)$ time by x-coordinate.
     (c) Event handling: Remove any segments ending in the current event point v from the tree. Complete the construction of the trapezoids ending at v (which were stored in some interior tree nodes) by adding a vertical edge through v. Finally, insert any segments starting at v into the tree, and initialize new trapezoids starting at v and store them in interior tree nodes.

   Consider the example on the next page, which shows the input line segments in black and the additional trapezoidal map edges and vertices in blue. (For simplicity we consider v_1, v_2, v_3, v_4 to be the bounding box.)

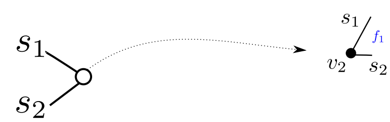
Flip over to next page $\Longrightarrow$



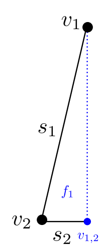
For this homework problem, you should draw for each event point in this example (1) the trapezoids that have been completed while handling this event, and (2) the sweep line status after this event has been handled. Please follow the format shown for the first few events below. You can arbitrarily choose the location of interior nodes and children of the binary search tree as long as it is balanced (so, don't worry about tree insertion algorithms or rotations).

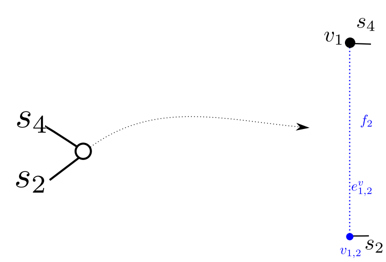
Event | Completed Trapezoids | Tree | Trapezoids

v_2

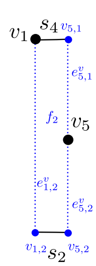


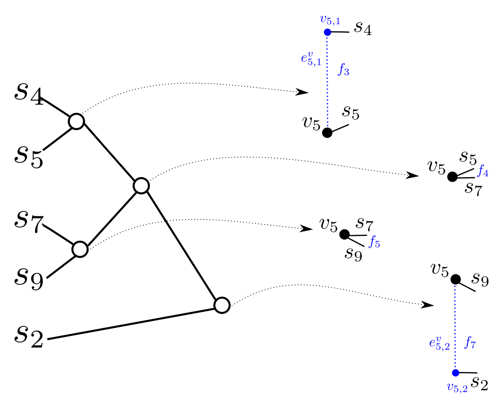
v_1





v_5





Flip over to next page $\Longrightarrow$

2. **Trapezoidal Map (10 points)**
Consider the following instance of the trapezoidal map point location data structure. The left side shows the map, and the right side shows the corresponding DAG. Describe the resulting trapezoidal map and DAG after segment $\overline{xy}$ has been added.