CIS677
Comments on homework 2

1. *Problem 2.14 of the text.*

   (Sweeping about a point p to find segments visible to p.)

   Since all endpoint events are known in advance, no event queue is necessary if the appropriate sorting by polar angle is done.

   Since a segment that is behind $n/2$ other segments can still be visible eventually, a heap is needed to maintain the status of the sweep. A linked list is not enough, and a binary tree is OK, but overkill.

   Few people noticed that initialization of the status is needed: unlike the usual sweep, there may be no position with an empty status.

2. *Problem 6.2 of the text.*

   Note that the worst-case behavior can occur even with line segments that do not intersect.

3. *Problem 6.3 of the text.*

   One way to handle degeneracy, that no one mentioned, is to say that a bounding edge crosses ρ just when one endpoint is above or on the ray ρ and the other endpoint is strictly below ρ. This would imply that:

   - an edge contained in ρ would be viewed as "not intersecting",
   - two edges with a common endpoint on ρ would either have

     two intersections, when the two other endpoints are strictly below ρ, or

     no intersections, when the others are on or above.

     one intersection, when one of the others is endpoint is above or on and the other is strictly below.

   The algorithm then simply adds up the number of intersecting edges: an odd number implies in, an even number implies out.

   Note that this does not require the sequence of edges to be used, or any decisions to require considering more than one edge at a time.

4. *Problem 6.12 of the text.*

5. *Suppose a triangulation with n vertices is given; included in this input is the adjacency information of the triangulation. A point location data structure for the triangulation can be built by building the trapezoidal diagram of its edges using the randomized incremental algorithm. Show that this data structure can be built in $O(n)$ expected time.*

   As I should have emphasized or said more clearly, the problem here is build "this data structure" in $O(n)$ time, where "this data structure" means one built by the randomized incremental procedure, exactly as output by such a procedure. The algorithm I had in mind was this: rather than build the trapezoidal diagram by adding segments one at a time, *start out* with the trapezoidal diagram, which is easily obtained from the triangulation (as some of you showed), and then delete segments at random.

   Suppose $TD(R_i)$ would be maintained by the usual randomized incremental algorithm. When a segment s_{i+1} is deleted by the "backwards" algorithm, some splits and merges of trapezoids take place, and $TD(R_i)$ is found. Now make pointers as in the usual algorithm, from trapezoids of $TD(R_i)$ that meet s_{i+1} to those of $TD(R_{i+1})$ that are unions of pieces of trapezoids from $TD(R_i)$.

   The work is only expected $O(1)$ in these backwards steps, because *no searching needs to be done:* the trapezoids incident to s_{i+1} are already known, and the trapezoids incident to each segment still in the random subset can be maintained to keep this information correct.

6. *Let S be a set of line segments in the plane, with no three having a common crossing point, and no endpoints or crossing points with the same y coordinate. Let R be a random subset of S of size i. For trapezoid T of $TD(R)$, let m_T be the number of segments of S that meet T. Let M be the sum of m_T over all trapezoids of $TD(R)$. Show that the expected value of M is $O(n + Ai/n)$, where A is the number of crossing segments of S. It will be helpful to know that the expected number of crossing points is exactly $Ai(i-1)/n(n-1)$.*

   (That should be same x coordinate, of course.)

   This was intended to be an extension of the corresponding Lemma 2 of Lecture 4, the case $A = 0$.

   A segment meets a trapezoid if one of the following is true:

   (a) it is contained in the trapezoid;

   (b) it crosses a bounding segment of the trapezoid.

   (c) it crosses a visibility edge of the trapezoid;

   The segments contained in trapezoids contribute n to M.

   Suppose a segment s crosses A_s segments of S. Then the expected number of segments of R that s crosses is $A_s i/n$, and the total number of crossings contributed to M is $\sum_{s \in S} A_s i/n = 2Ai/n$.

To account for the last category of segment/trapezoid meetings, adapt the proof of Lemma 2 of Lecture 9:

Let s be a random segment of $S \setminus R$, and let $R' \equiv R \cup \{s\}$, so that R' is a random subset of S, and s is a random member of R'.

We will bound the probability that s crosses a visibility edge of $TD(R)$; multiplying that by $n-i$ is a bound for the segment/visibility edge crossings.

Let

V denote the visibility edges of $TD(R)$, and

V' denote the visibility edges of $TD(R')$.

Then

$V \setminus V'$ is the visibility edges of $TD(R)$ crossed by s, and

$V' \setminus V$ is the visibility edges of $TD(R')$ incident to s.

We have

$$|V \setminus V'| + |V'| = |V' \setminus V| + |V|,$$

and so

$$E|V \setminus V'| = E|V' \setminus V| + E|V| - E|V'|,$$

but $E|V| = 4i + 2Ai(i-1)/n(n-1)$, and $E|V'| = 4(i+1) + 2Ai(i+1)/n(n-1)$, so the difference of the last two is $-4 - 4Ai/n(n-1)$, and $E|V' \setminus V| \leq 8 + 6Ai/n(n-1)$, since there are $4(i+1) + 2Ai(i+1)/n(n-1)$ visibility edges, with 2 (respectively 3) segments of R' incident to each one.

Hence

$$E|V \setminus V'| \leq 8 + 6Ai/n(n-1) - (4 + 4Ai/n(n-1)),$$

and multiplying by $n-i$ yields the result.