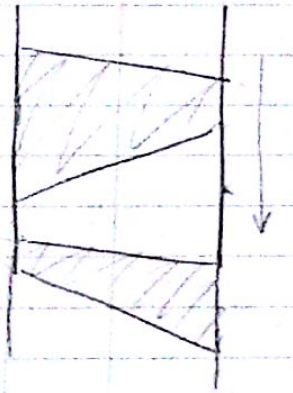
- We'll run the line intersection algorithm for the input first. These intersection points and triangle vertices will be our event points.

- Then we'll sort those event points by x-coordinate and start sweeping.

- At any given time, we will be sweeping from above. Status structure holds two line segments from "active" triangles. (which are intersecting with the sweep line). (I guess)

- Our goal is to find the area spanned by triangles "between" two event lines. Looks like this:



- Note that it is guarenteed that we only have line segments between these two sweep lines.

☆ We start from the top segment. If the segment at hand is an "upper edge" of a triangle, we increment the counter. If it is a "lower edge" we decrement the counter. Whenever counter hits zero, it means we found an area to be calculated. (spanned by triangles). We calculate the trapezoid area and move on below.

- "Upper" and "lower" edges can be determined by edge order or right hand rule.

- And we sum all the partial areas we found.

- Beginning of an edge will mean its insertion to status structure, end of an edge will mean deletion.

Complexity: At each event iteration, at the worst case we move over n triangles. Since the iteration count is of $O(n+1)$, total cost would be $O(n \cdot (n+1)) = O(n^2 + 1n)$