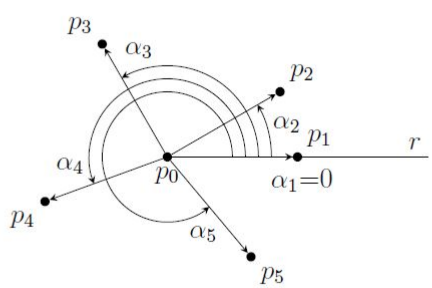
FTP_Alg_Week 7: Exercises and solutions

jungkyu.canci@hslu.ch

28 October 2024

Exercise 1 *The polar angle of a point p_i with respect of an origin p_0 is the angle from the semi horizontal straight line r (see picture below) and the vector $\vec{p_0 p_i}$. The positive direction of the angle is the counterclockwise one. Furthermore angle amplitude are taken in the interval $[0, 2\pi)$. In the picture below you find some examples of polar angle.*



Write a pseudocode, which orders n points $q_1, \ldots, q_n$ according their polar angles, in increasing order. The algorithm should have $O(n \log n)$ running time.

A possible answer. Strategy: Let $p_0 = [x_0, y_0]$ and $q_i = [x_i, y_i]$ express in coordinates. We divide the set of points in $A = \{q_1, \ldots, q_n\}$ in two groups. A_1 is the subset of A containing the points with coordinates $[x, y]$ such that $y \geq y_0$ and A_2 the one such that $y < y_0$. (this costs $O(n)$). The polar angles of the points in A_1 are in the interval $[0, \pi]$ and the polar angles of the points in A_2 are in the interval $(\pi, 2\pi)$. Concretely we assume that A is a vector whose entries are the points q_i. So

$$A = [q_1, \ldots, q_n]$$

We denote by $y.A[k] = y_k$ for all index $1 \leq k \leq n$. The following Algorithm give a partition of A where the fist i elements are the points in A_1 and the remianing are elements in A_2. It returns also the first index where we have an element of A_2.

Algorithm 1

procedure PARTITIONANGLE(A, y_0)
  $i = 0$
  for $j = 1$ `to` n **do**
    if $y.A[j] \geq y_0$ **then**
      $i = i + 1$
      `exchange` $A[i]$ `with` $A[j]$
  return $i + 1$

Now we consider a procedure similar to MERGE(A, p, q, r) (see slide 7/28 of Lecture 3 and 4) where the comparison $L[i] \leq R[j]$ is not the comparison of numbers. Let $L[i] = (x_i, y_i)$ and $R[j] = (x_j, y_j)$, then we replace the condition $L[i] \leq R[j]$ in the pseudocode in slide 7/28 of Lecture 3 and 4, with the following condition about the cross product

$$(x_i - x_0, y_i - y_0) \times (x_j - x_0, y_j - y_0) \geq 0$$

Let's denote the procedure from slide 7/28 of Lecture 3 and 4 with the above change with ANGLEMERGE(A, p, q, r)

Nun we define the following algorithm.

Algorithm 2

procedure ANGLEMERGE-SORT(A, p, r)
  if $p < r$ **then**
    $q = \lfloor (p + r)/2 \rfloor$
    ANGLEMERGE–SORT(A, p, q)
    ANGLEMERGE-SORT$(A, q + 1, r)$
    ANGLEMERGE(A, p, q, r)

This algorithm ANGLEMERGE-SORT(A, p, r) will be used to sort the in increasing order the sets A_1 and (separatly) the set A_2. Finally the last algorithm give the solution to the problem

Algorithm 3

procedure ANGLE-SORT(A)
  if $A \neq NIL$ **then**
    $n = length.A$
    $q =$PARTITIONANGLE(A, x_0)
    ANGLEMERGE–SORT$(A, 1, q - 1)$
    ANGLEMERGE–SORT(A, q, n)

We have created the two sets A_1 and A_2, beacuse with the cross-product we could have some troubles, because the peridocity 2π of the angles. For example in the above pictures, by using the cross-product, the point p_5 would be considered predecessor of p_1 and clearly we do not wnat this. With the creation of the sets A_1 and A_2 we avoid this problem and we have, for example, that p_5 is the last element in the order and so is not the predecessor of p_1.

Algorithm 1 costs $O(n)$. Algorithm 2 costs $O(m \log m)$ with $m = r - p$. Thus Algorithm 3 costs $O(n \log n)$.

Exercise 2 (*) *Given two segments a and b that are comparable at $\tilde{x}$, show how to determine in $O(1)$ time which of $a \succcurlyeq_{\tilde{x}} b$ or $b \succcurlyeq_{\tilde{x}} a$ holds. Assume that neither segment is vertical. (Hint: If a and b do not intersect, you can just use cross products. If a and b intersect—which you can of course determine using only cross products—you can still use only addition, subtraction, and multiplication, avoiding division. Of course, in the application of the $\succcurlyeq_{\tilde{x}}$ relation used here, if a and b intersect, we can just stop and declare that we have found an intersection.)*

A possible answer. We consider two cases:

1) The two segments intersect.

2) The two segments do not intersect.

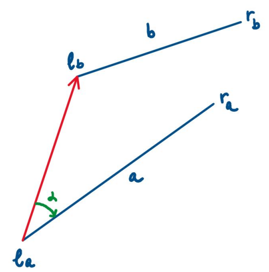
Let us denote by ℓ_a and r_a the left point and the right point of a respectively. In the same way we donote by ℓ_b and r_b the left point and the right point of b. We express the coordinates of these points in the following way

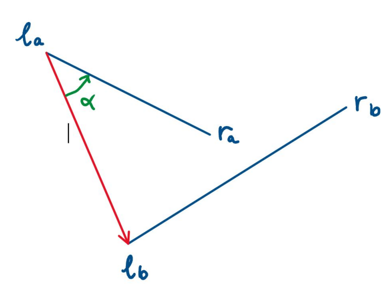
$$\ell_a = (x_1, y_1) \ , \ r_a = (x_2, y_2) \ , \ \ell_b = (x_3, y_3) \ , \ r_b = (x_4, y_4)$$

Case 1) Suppose that the segments a and b do not intersect. Without loss of generality suppose that $x_1 \leq x_3$, that is the left point of a is left of the left point of b. Otherwise in the following formulas swap a with b. Consider the vector $\ell_b - \ell_a$ and the cross-product

$$cp = (\ell_b - \ell_a) \times (r_a - \ell_a).$$

Note that the sweep line at x is so that $x_3 \leq \tilde{x} \leq \min\{y_2, y_4\}$. We are in one of the following situations:



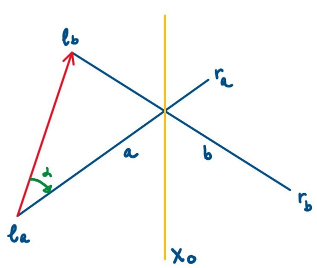


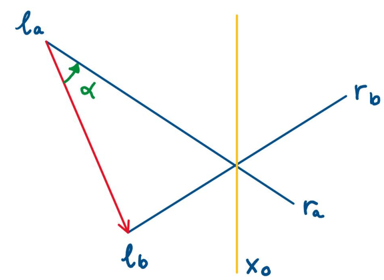
In both above cases the angle α has a magnitude between 0 and π. In the case left, α turns in the clockwise direction so the cross-product cp is negative. In the case right α turns in the counterclockwise direction so the cross-product cp is positive.

We summarize the above reasoning in the case where a and b do not intersect (under the hypothesis that $x_1 \leq x_3$, otherwise swap in the following a with b):

- If $cp = (\ell_b - \ell_a) \times (r_a - \ell_a) < 0$, then is $b \succcurlyeq_{\tilde{x}} a$ for any sweep line at $\tilde{x}$ where a and b are comparable.
- If $cp = (\ell_b - \ell_a) \times (r_a - \ell_a) > 0$, then is $a \succcurlyeq_{\tilde{x}} b$ for any sweep line at $\tilde{x}$ where a and b are comparable.

Case 2) Suppose that the segments a and b intersect in a point with coordinates (x_0, y_0). As before, WLOG, we assume that $x_1 \leq x_3$. Therefore one of the two situation is possible





Therefore is it important to know if $\tilde{x}$ (which gives the sweep line) is $< x_0$ or $\geq x_0$.

The straight line containing the segment a, and so the points ℓ_a and r_a, has equation

$$y = \frac{y_2 - y_1}{x_2 - x_1}(x - x_1) + y_1$$

The straight line containing the segment b, and so the points ℓ_b and r_b, has equation

$$y = \frac{y_4 - y_3}{x_4 - x_3}(x - x_3) + y_3$$

Thus (x_0, y_0) is the solution of the system of the two above equations, in particular we have that

$$\frac{y_2 - y_1}{x_2 - x_1}(x_0 - x_1) + y_1 = \frac{y_4 - y_3}{x_4 - x_3}(x_0 - x_3) + y_3$$

thus

$$\begin{aligned}(y_2 - y_1)(x_4 - x_3)(x_0 - x_1) + y_1(x_2 - x_1)(x_4 - x_3) = \\ = (y_4 - y_3)(x_2 - x_1)(x_0 - x_3) + y_3(x_2 - x_1)(x_4 - x_3)\end{aligned}$$

By solving x_0 in the above equation we obtain

$$x_0 = \frac{\alpha}{\beta}$$

with α the number

$$y_3(x_2-x_1)(x_4-x_3)-y_1(x_2-x_1)(x_4-x_3)-x_3(y_4-y_3)(x_2-x_1)+x_1(y_2-y_1)(x_4-x_3)$$

and

$$\beta = (y_2 - y_1)(x_4 - x_3) - (y_4 - y_3)(x_2 - x_1)$$

Now remark that $\tilde{x} < x_0$ if and only if $\beta\tilde{x} < \alpha$ and $\tilde{x} \geq x_0$ if and only if $\beta\tilde{x} \geq \alpha$.

Therefore we can summarize the case 2 as follow:

- If $cp = (\ell_b - \ell_a) \times (r_a - \ell_a) < 0$, then is $b \succcurlyeq_{\tilde{x}} a$ for any sweep line at $\tilde{x}$ where a and b are comparable and $\beta\tilde{x} < \alpha$.
- If $cp = (\ell_b - \ell_a) \times (r_a - \ell_a) < 0$, then is $a \succcurlyeq_{\tilde{x}} b$ for any sweep line at $\tilde{x}$ where a and b are comparable and $\beta\tilde{x} \geq \alpha$.
- If $cp = (\ell_b - \ell_a) \times (r_a - \ell_a) > 0$, then is $a \succcurlyeq_{\tilde{x}} b$ for any sweep line at $\tilde{x}$ where a and b are comparable and $\beta\tilde{x} < \alpha$.
- If $cp = (\ell_b - \ell_a) \times (r_a - \ell_a) > 0$, then is $b \succcurlyeq_{\tilde{x}} a$ for any sweep line at $\tilde{x}$ where a and b are comparable and $\beta\tilde{x} \geq \alpha$.

Note that the first two cases represent the left part situation in the above picture and the last two cases the right part in the above picture. The conditions to be verified in case 1) and case 2) only need sums (subtractions) and multiplications and do not need divisions as requested in the text of the exercise.

Exercise 3 *Argue that ANY-SEGMENTS-INTERSECT works correctly even if three or more segments intersect at the same point.*

A possible answer. We have to show that if three segment (that means at least three segments) intersect at a point, then ANY-SEGMENTS-INTERSECT returns true. Recall that ANY-SEGMENTS-INTERSECT returns true, when one of these two case occur:

- At a point stop we insert a segment, which intersects the above segment or the below segment, with respect the order given by the sweep line (see lines 6 and 7 of ANY-SEGMENTS-INTERSECT).
- At a point stop we delete a segment such that its above and below segments meets in a point(see lines 9 and 10 of ANY-SEGMENTS-INTERSECT).

Let p be the first intersection point of n segments $a_1, \ldots, a_n$ with $n \geq 3$. If there is a point q to the left of p, where in q meets exactly 2 segments, then the algorithm returns true, so there is nothing to prove.

Thus we can assume that p is the first event point, where we have an intersection of segments. Let x_0 the x coordinate of the first sweep line, for which two segments a_i and a_j (of the above set) become consecutive in the list of the partial order given by the sweep line at x_0. Therefore only one of the two cases are possible

- The event point p is the left point of a segment a_i or a_j. WLOG we can assume that a_i is already in the preorder at the event point prior to p, therefore we add a_j, and the two a_i and a_j become consecutive in the preorder at p. Thus 6 and 7 of ANY-SEGMENTS-INTERSECT return true.

- The event point p is the right point of a segment b, therefore we eliminate b, which has as above and below segment the two a_i and a_j. Thus 9 and 10 of ANY-SEGMENTS-INTERSECT return true.

Exercise 4 *Show that ANY-SEGMENTS-INTERSECT works correctly in the presence of vertical segments if we treat the bottom endpoint of a vertical segment as if it were a left endpoint and the top endpoint as if it were a right endpoint. How does your answer to above Exercise 3 change if we allow vertical segments?*

A possible answer. First of all we observe that by using the rule written in the text of the exercise for sorting the event points, we apply the usual rule, that is, we insert in the list before the left end points with smaller y coordinate, then right end points with smaller y coordinate. After the event points "left point" v_s of a vertical segment, where we insert in the preorder that vertical segment, we test if its above segment intersect the vertical line. If yes lines 6 and 7 of ANY-SEGMENTS-INTERSECT return true. If not, there is no intersection with the vertical line and segments in the preorder given by the sweep line at v_s, Then we consider event points that are right points of segments which meets the segments, thus 9 and 10 of ANY-SEGMENTS-INTERSECT return true, if there are such segments. If not, next event point is the right point v_r of the vertical segment and so ANY-SEGMENTS-INTERSECT continues in a *normal* way.

If in the previous Exercise 3 we assume that in the intersection of more than two segments intersect, one of the segments is vertical, nothing changes. By using the same notation of Exercise 3, if the two segments a_i and a_j are non vertical, clearly nothing changes. If one of the segments a_i and a_j is vertical, then all segments are collinear at the same point except at most one. The algorithm ANY-SEGMENTS-INTERSECT works because it consider the event points according to the y coordinate and SEGMENTS-INTERSECT also works for the intersection of vertical segments.