CS/ECE 374 A (Spring 2024)
Homework 5 (due Feb 29 Thursday at 10am)

Instructions: As in previous homeworks.

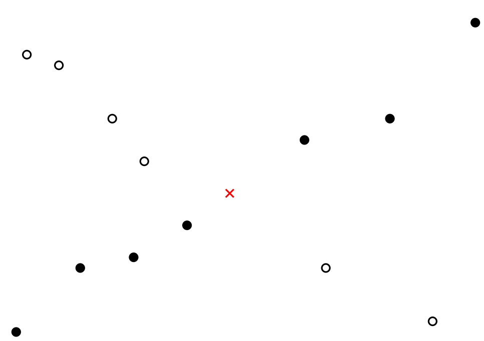
In algorithm design problems, an algorithm is usually best described using pseudocode (not actual code!), together with an explanation of the ideas behind the algorithm and/or justification of correctness (if correctness is not obvious), and accompanied by an analysis of the running time. See `https://courses.engr.illinois.edu/cs374al1/sp2024/hw_policies.html#content`.

Problem 5.1: A *point* p in 2D is specified by its x-coordinate $p.x$ and y-coordinate $p.y$.

We say that a set P of points in 2D is *increasing* if for every two points $p_i, p_j \in P$ with $p_i.x < p_j.x$, we have $p_i.y < p_j.y$. A set Q of points in 2D is *decreasing* if for every two points $q_i, q_j \in Q$ with $q_i.x < q_j.x$, we have $q_i.y > q_j.y$. Given an increasing point set P and a decreasing point set Q, a *crossing point* is a point s such that $P \cup \{s\}$ remains increasing and $Q \cup \{s\}$ remains decreasing; here, s may or may not be in $P \cup Q$. (A crossing point always exists between an increasing and a decreasing point set—you may use this fact without proof. You may assume that no two points have the same x- or y-coordinates.)

For example: for the increasing point set $P = \{(1,0),(4,1),(5,4),(8,5),(13,7)\}$ and the decreasing point set $Q = \{(0,11),(2,10),(7,6),(9,2),(11,1)\}$, a crossing point is $(7.5, 4.5)$.

See below for another example, where P is drawn in black, Q in white, and a crossing point in red.



(a) (50 pts) Let P be an increasing set of points, and Q be a decreasing set of points. Suppose that P is given in sorted x-order, and Q is given in sorted x-order. Design and analyze an $O(\log|P|\log|Q|)$-time algorithm to find a crossing point.

(Hint: Pick the "middle" point p_m of P. Let q be the rightmost point in Q with $q.x < p_m.x$, and q' be the leftmost point in Q with $q'.x > p_m.x$ (how fast can we find q and q'?). By comparing $p_m.y$ with $q.y$ and $q'.y$, try to eliminate roughly half of P.)

(Bonus: A faster correct solution achieving $O(\log|P| + \log|Q|)$ running time may receive up to 10 extra points.)

(b) (50 pts) Suppose instead that P and Q are given in arbitrary order (i.e., not necessarily sorted). Design and analyze an $O(n)$-time algorithm to find a crossing point, where $n = |P| + |Q|$.

(Hint: the hint from (a) may still be helpful here, though you may consider picking the point with the median x-coordinate of $P \cup Q$ instead of P. You may use the linear-time algorithm for median finding or selection, from class, as a subroutine.)

Problem 5.2:

(a) (85 pts) Consider the following problem: given a sequence of integers $b_0, \ldots, b_{n-1}$ all lying between $-M$ and M, compute the following rational number exactly:

$$X = \sum_{i=0}^{n-1} \frac{16^{n-1-i}}{b_i} = \frac{16^{n-1}}{b_0} + \frac{16^{n-2}}{b_1} + \cdots + \frac{16^0}{b_{n-1}}.$$

More precisely, we want to compute the binary representation of some integers A and B (the numerator and denominator) such that $X = \frac{A}{B}$ (where $|B| \leq M^n$ and $|A| < 16^n M^n$). Design an efficient divide-and-conquer algorithm to solve this problem. You may use Karatsuba's multiplication algorithm as a subroutine. Analyze the running time of your algorithm, which should be bounded by $O((n \log M)^{\log_2 3})$.

(b) (15 pts) Define the sequence

$$\pi_n = \sum_{i=0}^{n-1} \frac{1}{16^i} \left(\frac{4}{8i+1} - \frac{2}{8i+4} - \frac{1}{8i+5} - \frac{1}{8i+6} \right).$$

By applying part (a)[1], show how to compute π_n for a given n. More precisely, we want to compute the binary representation of some integers A and B such that $\pi_n = \frac{A}{B}$. Analyze the running time as a function of n.

(Note: as you might have guessed, π_n converges to π; this result was due to Plouffe (1995).)

[1] Generally, in a multi-part problem like this, if you are unable to solve (a), you can still do (b) under the assumption that (a) has been solved.