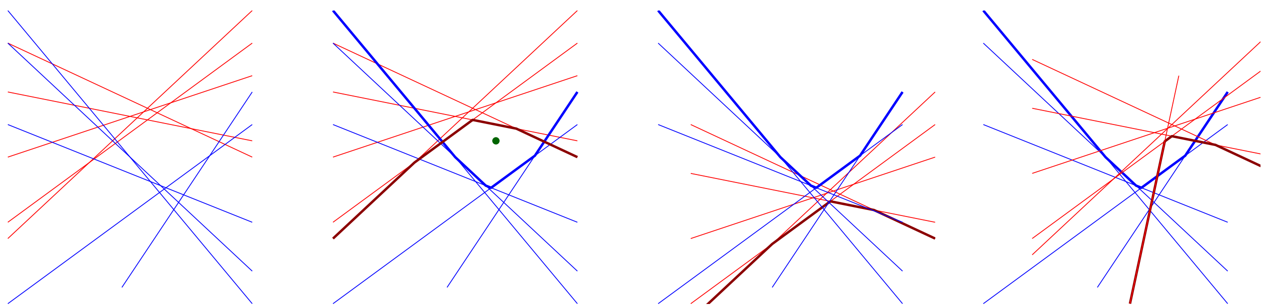
HW 3 (due Wednesday, at noon, September 19, 2018)

CS 473: Algorithms, Fall 2018 Version: **1.01**

Submission guidelines and policies as in homework 1.

1 (100 PTS.) Sandwich point.

Let T and B be two sets of lines in the plane, in general position, such that $|T| + |B| = n$. The lines of T are top, and the lines of B are bottom. Here, general position implies that no two lines of $T \cup B$ are parallel, no line of $T \cup B$ is vertical or horizontal, no three lines of $T \cup B$ meets in a point, and no two vertices have the same x coordinate (a vertex is a point of intersection of two lines). Here, we are interested in computing in linear time a point that is above all the bottom lines of B and below all the top lines of T. Such a point is ***feasible***. (Here, a point that is on a line, can be considered to be either above or below the line, as convenient.) Here are a few examples:



Input feasible point Infeasible input Another feasible input.

We interpret a line $\ell \equiv y = ax + b \in T \cup B$ as a function from $\mathbb{R}$ to $\mathbb{R}$. In particular, for $z \in \mathbb{R}$, let $\ell(z)$ be the y coordinate of $\ell \cap (x = z)$ – that is, $\ell(z) = az + b$.

1.A. (10 PTS.) For $x \in \mathbb{R}$, let $U_B(x) = \max_{\ell \in B} \ell(x)$ be the ***upper envelope*** of B. Similarly, let $L_T(x) = \min_{\ell \in T} \ell(B)$ be the ***lower envelope*** of T. Prove that $U_B(x)$ is a convex function, and $L_T(x)$ is a concave function.

1.B. (10 PTS.) Consider the function $Z(x) = L_T(x) - U_B(x)$. Argue that $Z(x)$ is concave, and show how to compute its value at point x in linear time.

1.C. (10 PTS.) Prove that $Z(x) \geq 0$ for some $x \iff$ there a point above all the bottom lines, and below all the top lines.

1.D. (20 PTS.) Let x^* be the value where $Z(\cdot)$ achieves its maximum. Let $X = \{x_1, \ldots, x_m\}$ be a set of m numbers, and let $z_1 = \mathit{Select}(X, \lfloor m/2 \rfloor - 1)$, $z_2 = \mathit{Select}(X, \lfloor m/2 \rfloor)$, and $z_3 = \mathit{Select}(X, \lfloor m/2 \rfloor + 1)$, where $\mathit{Select}(X, i)$ is the ith rank element in X.

Show, how in linear time, one can decide if $x^* \in [z_1, \infty]$ or $x^* \in [-\infty, z_3]$,

Note, that this implies that for roughly half the values of X, this decides that the maximum of $Z(\cdot)$ is either bigger or smaller than them.

1.E. (20 PTS.) For the case that $|T| \geq |B|$, show how, in linear time, either

(i) compute a feasible point of T and B, or

(ii) compute a set $T' \subseteq T$, such that the maximum of $Z_{T,B}(\cdot)$ and $Z_{T',B}(\cdot)$ is the same, where $Z_{T',B}$ is the function Z described above for the sets T' and B.

(Hint: Follow the algorithm seen in class.)

1.F. (30 PTS.) Given T and B, describe a linear time algorithm for computing a feasible point if it exists. If it does not, the algorithm should output that there is no such point.

2 (100 PTS.) Absolutely not subset sum.

Let $B = \{b_1, \ldots, b_m\} \subseteq [\![U]\!] = \{1, 2, \ldots, U\}$. A number $t \leq U$ is ***n*-representable** by B, if there exists integer numbers $\alpha_i \geq 0$, for $i = 1, \ldots, m$, such that

(i) $\sum_{i=1}^{m} \alpha_i = n$, and

(ii) $\sum_{i=1}^{m} \alpha_i b_i = t$.

Show how to compute, as fast as possible, if t is n-representable by B by an algorithm with running time close to linear in m and U (the dependency of the running time on n should be polylogarithmic in n).

[Hint: Use FFT.]

To make life easy for you, I broke it into three steps:

2.A. (30 PTS.) Show how to solve the case $n = 2$.

2.B. (20 PTS.) Show how to solve the case that n is a power of 2.

2.C. (50 PTS.) Show how to solve the general problem.

(As usual, the solutions to (A) and (B) are much simpler than (C), and are useful in solving (C).)

3 (100 PTS.) Computing Polynomials Quickly

In the following, assume that given two polynomials $p(x), q(x)$ of degree at most n, one can compute the polynomial remainder of $p(x) \bmod q(x)$ in $O(n \log n)$ time. The ***remainder*** of $r(x) = p(x) \bmod q(x)$ is the unique polynomial of degree smaller than this of $q(x)$, such that $p(x) = q(x) * d(x) + r(x)$, where $d(x)$ is a polynomial.

Let $p(x) = \sum_{i=0}^{n-1} a_i x^i$ be a given polynomial.

3.A. (25 PTS.) Prove that $p(x) \bmod (x - z) = p(z)$, for all z.

3.B. (25 PTS.) We want to evaluate $p(\cdot)$ on the points $x_0, x_1, \ldots, x_{n-1}$. Let

$$P_{ij}(x) = \prod_{k=i}^{j} (x - x_k)$$

and

$$Q_{ij}(x) = p(x) \bmod P_{ij}(x).$$

Observe that the degree of Q_{ij} is at most $j - i$.

Prove that, for all x, $Q_{kk}(x) = p(x_k)$ and $Q_{0,n-1}(x) = p(x)$.

3.C. (25 PTS.) Prove that for $i \leq k \leq j$, we have

$$\forall x \quad Q_{ik}(x) = Q_{ij}(x) \bmod P_{ik}(x)$$

and

$$\forall x \quad Q_{kj}(x) = Q_{ij}(x) \bmod P_{kj}(x).$$

3.D. (25 PTS.) Given an $O(n \log^2 n)$ time algorithm to evaluate $p(x_0), \ldots, p(x_{n-1})$. Here $x_0, \ldots, x_{n-1}$ are n given real numbers.