Please answer the following 3 questions, some of which have multiple parts. If asked for an $O(T(n))$ time algorithm, you may give a randomized algorithm with expected time $O(T(n))$.

1. (From Mount) This is a query variant of Problem 2(b) from Homework 2. You are given two sets $P = \{p_1, \ldots, p_m\}$ and $Q = \{q_1, \ldots, q_n\}$, which you may assume lie in the positive $x, y$-quadrant. The cannon is located at the origin, and its projectile function is restricted so that $a = 0$. In other words, given $b$ and $c$, the projectile travels along the curve $y = bx - cx^2$. The objective is to preprocess the sets $P$ and $Q$ so that the following queries can be answered efficiently. Given a pair $(b, c)$, determine whether the projectile passes above all the points of $P$ and below all the points of $Q$.

   ![Figure 1. The example scenario for Problem 1.](image)

Let $N = m + n$ be the total input size. The preprocessing time for your data structure should be at most $O(N \log N)$, your data structure should use space $O(N)$, and queries should be answered in time $O(\log N)$. Partial credit will be given if the space or preprocessing time is a bit higher, but the query time must be $O(\log N)$.

2. (From Mount) Given a set of $n$ points in the plane, we define a subdivision of the plane into rectangular regions by the following rule. We assume that all the points are contained within a bounding rectangle. Imagine that the points are sorted in increasing order of $y$-coordinate. For each point in this order, shoot a bullet to the left, to the right, and up until each bullet hits an existing segment, and then add these three bullet-path segments to the subdivision (see Figure 2(a)).

   (a) Show that the resulting subdivision has size $O(n)$ (including vertices, edges, and faces).
(b) Consider the following algorithm that adds a new point \( p \) to the subdivision and restores the proper subdivision structure. We determine the rectangle containing \( p \) and subdivide it by shooting bullets to the left and right of \( p \). Then, we iteratively following a bullet going up from \( p \) to the top of the bounding rectangle, trimming back each horizontal segment crossed by the bullet-path (see Figure 2(b)). Prove that if the \( n \) points are added in random order in this way, then the expected number of structural changes to the subdivision with each insertion is \( O(1) \).

3. (From Mount) This problem arises from computational biology. You are given a protein molecule \( P \) that consists of \( n \) atoms. Each atom is represented by a circular disk in the plane of radius \( r \) (see Figure 3(b)). Let \( P = \{ p_1, \ldots, p_n \} \) denote the centers of these atoms.

The protein molecule lives in a solution of water. A water molecule is represented by a circular disk of radius \( r' > r \) (see Figure 3(a)). Water molecule and protein molecules cannot overlap. We say that an atom of \( P \) is solvent-accessible if there exists a placement of a water molecule that touches this atom, but does not intersect any of the other atoms of \( P \). (In Figure 3(c), the two shaded atoms are the only ones that are not solvent-accessible.)

(a) Consider the Voronoi diagram of the center points of the atoms of \( P \). Show that an atom \( p_i \in P \) is solvent-accessible if and only if there exists a point within \( p_i \)'s Voronoi cell that is at distance at least \( r + r' \) from \( p_i \).

(b) Using (a), show that it is possible to determine all the atoms that are solvent-accessible in \( O(n \log n) \) time.