Computational Geometry - Homework III

K. Subramani
LCSEE, West Virginia University, Morgantown, WV
{ksmani@csee.wvu.edu}

1 Instructions

1. The homework is due on April 27, in class. Each question is worth 5 points.
2. Attempt as many problems as you can. You will be given partial credit, as per the policy discussed in class.

2 Problems

1. **Duality:** Let \( R \) be a set of \( n \) red points in the plane and \( B \) denote a set of \( n \) blue points. A line \( l \) is said to be a separator for \( R \) and \( B \), if \( l \) has all the red points on one side and all the blue points on the other. Describe a randomized, linear-time algorithm that decides whether the given sets \( R \) and \( B \) have a separator.

2. **Line Arrangements:** Let \( L \) denote a set of \( n \) lines in the plane and let \( A(L) \) denote their arrangement. Describe an \( O(n \cdot \log n) \) algorithm to compute an axis-parallel rectangle that contains all the vertices of \( A(L) \).

3. **Delaunay Triangulation:** Given a set \( P \) of \( n \) points in the plane, the Euclidean Minimum Spanning Tree (EMST) is the tree of total edge length connecting all the points in \( P \). Describe an \( O(n \cdot \log n) \) algorithm to compute the EMST for \( P \).

4. **Delaunay Triangulation:** Given a set \( P \) of \( n \) points, the Gabriel graph \( G(P) \) of \( P \) is defined as follows: Two points \( p \) and \( q \) are connected by an edge in \( G(P) \) if and only if the circle with diameter \( \overline{pq} \) does not contain any other point of \( P \) in its interior.
   (a) Prove that \( \mathcal{DG}(P) \) contains \( G(P) \), where \( \mathcal{DG}(P) \) denotes the Delaunay graph of \( P \).
   (b) Describe an \( O(n \cdot \log n) \) algorithm to compute \( G(P) \).

5. **Geometric Data Structures:** Let \( I \) denote a set of \( n \) intervals in the plane. Given an interval \( r = [x : x'] \), we are interested in all the intervals \( p \in I \), such that \( p \) is completely contained in \( r \). Describe a data structure that uses \( O(n \cdot \log n) \) storage and answers these queries in \( O(\log n + k) \) time, where \( k \) is the number of answers.