Objective

- In this lab, you will implement the Guibas and Stolfi’s Delaunay Triangulation algorithm.

Detailed Requirements

In the previous two labs, you created and strengthened a DCEL implementation. This lab, you will use your data structure to implement the divide-and-conquer Delaunay Triangulation algorithm (DT). The input is a list of points and the output is a DCEL structure that represents the Delaunay Triangulation of the input points. Although inefficient, you are allowed to use recursion in your implementation.

DCEL gsdet(const std::vector<Point>& p);

*Hint: You might want to extend the DCEL structure to maintain the convex hull of the points (vertices) in the DCEL. This will help you in the merge step.*