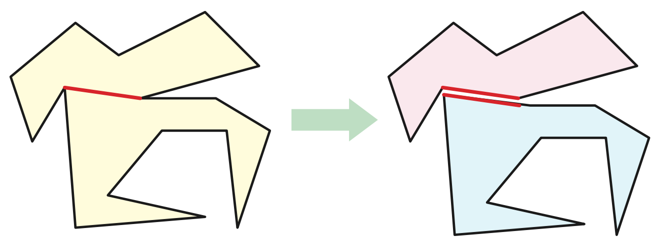
CS 498 TC ✧ Computational Geometry ✧ Spring 2022

Homework 1

Due Thursday, February 24, 2022 at 8pm

---

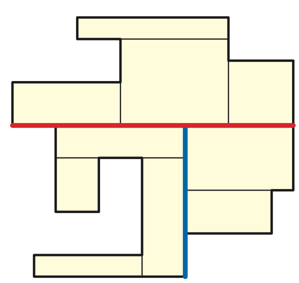
1. An interior diagonal of a simple polygon P with $n \geq 4$ vertices is a *balanced separator* if it subdivides P into two smaller polygons, each with at least $\lceil n/3 \rceil + 1$ vertices. Describe and analyze an algorithm to find a balanced separator in a given simple polygon P (with at least four vertices). *[Hint: Prove that a balanced separator always exists!]*



The red diagonal is a balanced separator of this polygon.

2. A simple polygon is *orthogonal* if its edges alternate between horizontal and vertical. An orthogonal polygon is *generic* if no pair of edges lies on a common horizontal or vertical line. Let P be an arbitrary generic simple orthogonal polygon with n vertices.

   A *rectangulation* R of P is a partition of the interior of P into axis-aligned rectangles. (These rectangles do not necessarily meet edge-to-edge.) A *bar* in a rectangulation R is a maximal connected union of collinear edges in R; every edge of R is contained in exactly one bar. A rectangulation R of P is *proper* if (1) no vertex has degree 4 and (2) every bar in R contains an edge of the polygon P.



A proper rectangulation of a generic orthogonal polygon; the red and blue segments are bars.

   (a) Prove that P has exactly $n/2 - 2$ reflex vertices.

   (b) Prove that every proper rectangulation of P has exactly $n/2 - 1$ rectangles.

   (c) Describe an algorithm to construct a proper rectangulation of P. (In particular, this algorithm proves that a proper rectangulation always exists!)

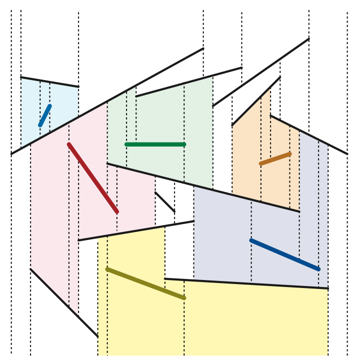
   (d) Prove or disprove: In every proper rectangulation R of P, every rectangle in R touches the boundary of P.

3. Let $S = \{s_1, s_2, \ldots, s_n\}$ be a set of n disjoint line segments, and let $T(S)$ denote the trapezoidal decomposition of S. Each trapezoid $\Delta \in T(S)$ is *incident* to at most four segments in S: One containing the ceiling of Δ, one containing the floor of Δ, one with an endpoint defining the left wall of Δ, and one with an endpoint defining the right wall of Δ. (Some of these segments may coincide; others may may not exist at all.) For each segment $s \in S$, let $\mathbf{deg}(s)$ denote the number of trapezoids incident to s.

   (a) Prove that $\sum_{i=1}^{n} \deg(s_i) \leq \alpha n$ for some constant α.

   (b) Let α be the constant derived in your solution to part (a). We say that a segment $s \in S$ is *long* if $\deg(s) \geq 2\alpha$ and *short* otherwise. Prove that the number of short segments in S is at least βn, for some constant β (which may depend on α).

   (c) An *independent set* of segments is any subset $I \subseteq S$ such that each trapezoid in $T(S)$ is incident to at most one segment in I. Prove that S contains an independent set of at least γn *short* segments, for some constant γ (which may depend on α and β).



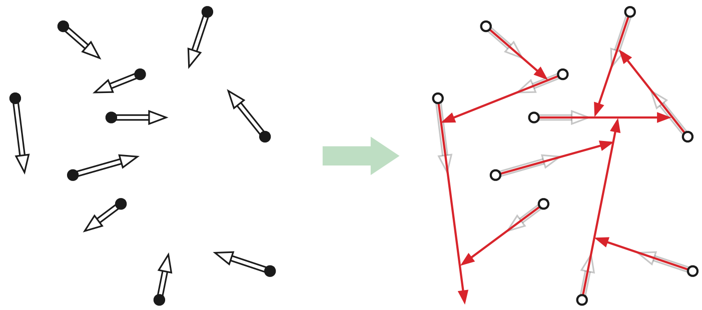
An independent set of 6 segments, each with degree at most 6.

4. Suppose we are given a set M of n moving points in the plane, each with an initial position (x_i, y_i) and a fixed velocity (u_i, v_i). Under normal circumstances, at any time $t > 0$, the ith point in M has coordinates $p_i(t) = (x_i + t \cdot u_i, y_i + t \cdot v_i)$.

   However, if a moving point p_i ever reaches a location that another point has previously occupied, p_i stops moving forever. Similarly, if two moving points collide, by occupying the same location at the same time, they both stop forever. (But assuming general position, this never actually happens.)

   Eventually, each moving point either stops (by running into another points trail) or escapes off to infinity. The complete pattern of trails left by the points is called the *motorcycle graph* of M.[1]

[1]Motorcycle graphs were inspired by the "light cycles" in the 1982 Disney movie *Tron*, except that the moving points can't turn, brake, or accelerate, and their jetwalls remain up forever, even after they derez.



Ten moving points and their motorcycle graph.

(a) Describe an algorithm to compute the motorcycle graph of n moving points in $O(n^2 \log n)$ time.

(b) Describe an algorithm to compute the motorcycle graph of n moving points in $O(n \log n)$ time when $x_i = 0$ for every index i; that is, all n points start on the y-axis.