1. (a) Let PH be a polygon with h holes, with a total of n vertices. How many triangles does every frugal triangulation of PH have, as a function of n and h? Prove your answer is correct.

Solution: The number of triangles is $\boldsymbol{n+2h-2}$.

Let PH be an arbitrary polygon with $n \geq 3$ vertices and $h \geq 0$ holes. We argue primarily by induction on h and secondarily by induction on n. If PH is a triangle ($n = 3$ and $h = 0$), the only frugal triangulation of PH trivially has $n + 2h - 2 = 1$ triangle. Otherwise, fix a frugal triangulation T of PH, and let pq be any diagonal in T. There are two cases to consider.

- If p and q lie on the same polygon, then pq splits PH into two smaller polygons with holes PH_1 and PH_2. Suppose PH_1 has n_1 vertices and h_1 holes, and PH_2 has n_2 vertices and h_2 holes. The induction hypothesis implies that the induced triangulations of PH_1 and PH_2 have $n_1 + 2h_1 - 2$ and $n_2 - 2h_2 - 2$ holes, respectively. By construction, we have $n_1 + n_2 = n + 2$ and $h_2 = h_2 = h$.
- Suppose p and q lie on different polygons. Then cutting PH along pq yields a polygon PH' with $n+2$ vertices (because p and q appear twice) and $h-1$ holes. The induction hypothesis implies that the induced triangulation of PH' has $(n+2)+2(h-1)-2 = n+2h-2$ triangles.

In both cases, we conclude that T has $n + 2h - 2$ triangles. ■

Rubric: 6 points.

(b) Sketch an algorithm to compute a frugal triangulation of PH in $O(n \log n)$ time. You only need to describe any necessary changes from the algorithm described in class (and its analysis) for triangulating simple polygons. (Your algorithm will prove that every polygon with holes has a frugal triangulation!)

Solution: Recall that the faster algorithm for triangulating simple polygons *without* holes has three stages:

i. Compute a trapezoidal decomposition of the input polygon using a sweep-line algorithm.
ii. Partition the input polygon into a collection of monotone mountains, by cutting along the diagonal in each interior boring trapezoid.
iii. Triangulate each monotone mountain using three pennies.

The first phase runs in $O(n \log n)$ time; the other two phases run in $O(n)$ time.

This algorithm also triangulates polygons *with* holes, still in $O(n \log n)$ time, ***with no modifications whatsoever.*** No part of algorithm assumes that the input is a simple polygon without holes; it only assumes that the input is a planar straight-line graph in which every vertex has degree 2. ■

Rubric: 4 points. This is more detail than necessary for full credit.

2. (a) Describe an algorithm to sort the vertices of a given simple polygon in $O(n \log t)$ time, where n is the number of vertices and t is the number of turn vertices.

Solution: We can recast the problem as follows: Given t sorted arrays of total length n, merge them into a single sorted array. The problem is trivial if $t = 1$. Otherwise, recursively merge any $\lfloor t/2 \rfloor$ input arrays, recursively merge the remaining $\lceil t/2 \rceil$ input arrays, and finally merge the two arrays returned by the recursive calls into a single sorted array in $O(n)$ time. ■

Rubric: 3 points

(b) Sketch an algorithm to triangulate a simple n-gon with t turn vertices in $O(n \log t)$ time. You only need to describe any necessary changes from the algorithm described in class (and its analysis) for triangulating simple polygons.

Solution: I'll use the same algorithm from problem 1, changing only the analysis of the sweep-line construction of the trapezoidal decomposition. Let P be the input polygon. We can compute the sorted sequence of events in $O(n \log t)$ using the algorithm in part (a).

I claim that any vertical line intersects at most t edges of P. This claim implies that at all times, the sweep structure (the balanced binary search tree) contains at most t segments, and therefore supports predecessor queries, successor queries, insertions, and deletions in $O(\log t)$ time. Because the algorithm performs $O(n)$ queries and updates, the overall time for the sweep is $O(n \log t)$.

Fix a vertical line ℓ that does not contain any vertex of P. (Lines through vertices cannot have more intersection points.) Index and direct the edges of P in cyclic order. Let e_i and e_k be two edges of P that cross ℓ, such that no edge e_j such that $i < j < k$ crosses ℓ. Without loss of generality, suppose e_i crosses ℓ from left to right, and therefore e_k crosses ℓ from right to left. Let j be the smallest index greater than i such that e_j is directed from right to left; by construction we have $j \leq k$. The vertex $e_{j-1} \cap e_j$ is a turn vertex. We conclude that P has at least one turn vertex between any two intersections with ℓ. It follows that P crosses ℓ at most t times, as claimed. ■

Rubric: 7 points = 2 for same algorithm + 3 for proving at most t intersections + 2 for time analysis

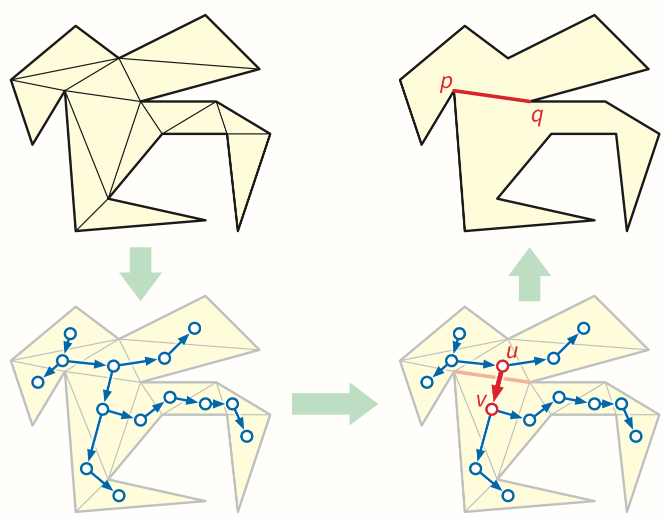
3. An interior diagonal of a simple polygon P with $n \geq 4$ vertices is a *balanced separator* if it subdivides P into two smaller polygons, each with at least $\lceil n/3 \rceil + 1$ vertices. Describe and analyze an algorithm to find a balanced separator in a given simple polygon P (with at least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

Solution: First I'll prove the following stronger theorem: *Every frugal triangulation of P contains a diagonal that is a balanced separator.*

Fix an arbitrary frugal triangulation T of P, and let T^* be its dual tree. T^* has exactly $N = n - 2$ vertices, one for each triangle in the triangulation. Choose an arbitrary leaf of T^* (that is, an ear in T) as its root, so that every vertex in T has at most two children. For any vertex v (that is, any triangle in T), let N_v denote the number of descendants of v in T, including v itself.

Now let u be the deepest vertex in T^* such that $N_u \geq (2N+1)/3$. (This vertex is clearly unique.) If u has only one child, let v be that child; if u has two children, label them v and w so that $N_v \geq N_w$. By construction, we have $N_v \geq \frac{1}{2}((2N+1)/3 - 1) = (N-1)/3$ and $N_v < (2N+1)/3$, which implies $N - N_v > (N-1)/3$. In other words, deleting edge uv splits T^* into two smaller trees, each with at least $(N-1)/3$ vertices.

Let pq be the diagonal that is dual to the tree edge uv. We just showed that cutting along pq splits T into two smaller polygon triangulations, each with at least $(N-1)/3$ triangles, and therefore at least $(N-1)/3 + 2 = n/3 + 1$ vertices. We conclude that pq is a balanced separator. (We get the ceiling for free, because the number of vertices must be an integer!)



Thus, we can find a balanced separator by triangulating P in $O(n \log n)$ time, computing the depth and number of descendants of each vertex in the triangulation's dual tree in $O(n)$ time, and then choosing the tree edge uv as described above. The total running time of the algorithm is $O(n \log n)$. ■

Rubric: 10 points = 7 for proof + 3 for algorithm. Partial credit for weaker separation bounds:

- -1 for $n/3 + c$ for any constant $c < 1$.
- -2 for $\alpha n \pm o(n)$ for any constant $0 < \alpha < 1/3$.

4. Let P be an arbitrary generic simple orthogonal polygon with n vertices.

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

Solution: Orient P counterclockwise (that is, with the interior on the left), so that its total turning angle is 1. Suppose P has r reflex vertices, and therefore $n-r$ convex vertices. Each convex vertex has a turning angle of $+1/4$, and each reflex angle has a turning angle of $-1/4$. It follows that $(n-r)/4-r/4=1$, which immediately implies $r=n/2-2$. ■

Rubric: 2 points.

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

Solution (Euler): Let R be an arbitrary proper rectangulation R of P. I'll compute the number of rectangles in R by counting its vertices and edges and then applying Euler's formula.

Our general position assumption implies that every vertex of R has degree 2 or 3. A vertex of R has degree 2 if and only if it is a convex boundary vertex. Thus part (a) implies that R has exactly $n/2+2$ degree-2 vertices. Counting degree-3 vertices is more complicated.

At each degree-3 vertex v, exactly one incident edge is adjacent to two right angles; call this edge the *stem* of v and the other two edges the *arms* of v. (Think of the vertical stem and horizontal arms of the letter T.) The interior edges of R are exactly covered by orthogonal line segments of three different types:

- Segments that connect an arm of a reflex boundary vertex with the stem of another vertex.
- Segments that connect arms of two reflex vertices. But any such segment must be collinear with two boundary edges, violating our assumption that the polygon P is generic.
- Segments that connect stems of two vertices. But any such segment is a bar that does not contain an edge of P, violating our assumption that the rectangulation R is proper.

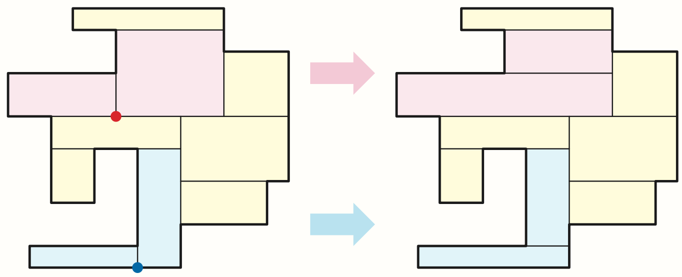
We conclude that the number of degree-3 vertices in R is exactly twice the number of reflex boundary vertices. Part (a) implies that R has exactly $n-4$ degree-3 vertices.

Altogether, R has $3n/2-2$ vertices. The number of edges is $\frac{1}{2}\sum_v \deg(v) = \frac{1}{2}(2\cdot(n/2+2)+3\cdot(n-4)) = 2n-4$. Thus, **Euler's formula** implies that R has $2-V+E = 2-(3n/2-2)+2n-4 = n/2$ faces, *including the outer face*. We conclude that R has $n/2-1$ interior faces, as claimed. ■

Solution (flips): The simplest non-trivial example of an orthogonal polygon is an L-shaped hexagon; this polygon has exactly two proper rectangulations, each with two rectangles.

Let R be any proper rectangulation of any generic orthogonal polygon P. Call a vertex of R *new* if it is not also a vertex of P. Genericity implies that every new vertex v of R has degree 3; moreover the two right angles at v both lie in the interior of P. The union of these two rectangles is orthogonal hexagon.

A *flip* transforms R into another rectangulation R' by finding two adjacent rectangles in R whose union is an orthogonal hexagon L and replacing those two rectangles with the other rectangulation of L. (See the figure on the next page.) Flips do not change the number of vertices, edges, or rectangles. Every new vertex of R corresponds to a possible flip.



Two flips in a rectangulation.

So let R be any proper rectangulation of P with $h > 0$ horizontal edges. Let e be any horizontal edge; let b be the horizontal bar containing R, let v be an endpoint of b that is *not* a vertex of P (which must exist by genericity), and let uv be the edge in b that is incident to v. Then v is a new vertex; the corresponding flip replaces the horizontal edge uv with a vertical edge uw. The resulting rectangulation R' has $h-1$ horizontal edges. It follows by induction that some sequence of h flips transforms R into a rectangulation of P with only vertical edges.

In fact, P has a unique vertical rectangulation $R^|$, which is constructed by the sweepline algorithm in my solution to part (c). Suppose P has at least one reflex vertex, since otherwise the problem is trivial. Let r be a rectangle in $R^|$ whose right wall contains the leftmost reflex vertex of P. (There are at most two such rectangles.) Removing r from $R^|$ yields a smaller rectangulation with $n-2$ boundary vertices and (by the induction hypothesis) $(n-2)/2-1 = n/2-2$ rectangles. It follows that that $R^|$ has $n/2-1$ rectangles. Because flips do not change the number of rectangles, we conclude that every proper rectangulation of P has $n/2-1$ rectangles. ■

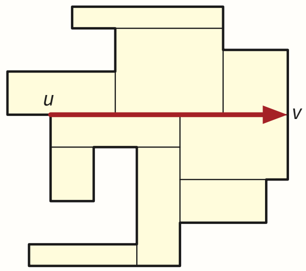
Solution (induction): I'll prove the claim by induction on n. Let R be an arbitrary proper rectangulation of an arbitrary generic orthogonal polygon P, and let n be the number of vertices of P. If P is a rectangle, R has exactly one rectangle and $n = 4$, and therefore $n/2-1 = 4/2-1 = 1$. So assume that $n \geq 6$.

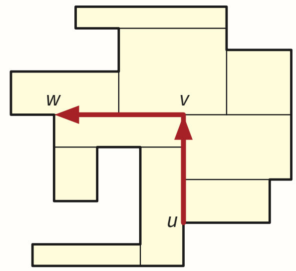
Let b be any bar in R that intersects the interior of P. The definitions of "proper" implies that b contains an edge e of P. Let s be a maximal subsegment of b in the interior of P. (There are at most two such segments.) One endpoint u of s is also an endpoint of e, and therefore a reflex vertex of P. There are two possibilities for the other endpoint v:

- Suppose v lies on the boundary of P. If v were a *vertex* of P, it would lie on an edge collinear with e, violating our assumption that P is generic.

Segment uv partitions R into two smaller rectangulations R_1 and R_2, of two smaller orthogonal polygons P_1 and P_2, respectively.

- Suppose v lies in the interior of P. Then v must lie on another bar b' orthogonal to b. This bar must contain another edge e' of P, and therefore must contain a segment vw, where w is a reflex vertex of P. The L-shaped path uvw partitions R into two smaller rectangulations R_1 and R_2, of two smaller orthogonal polygons P_1 and P_2, respectively.



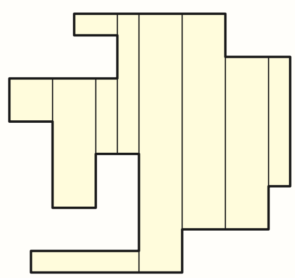


In both cases, v is a vertex of both P_1 and P_2 each vertex of P is a vertex of exactly one of P_1 and P_2, and these are the only vertices of P_1 and P_2. Thus, $n_1 + n_2 = n + 2$, where n_1 and n_2 respectively denote the number of vertices of P_1 and P_2. The induction hypothesis implies that R_1 has $n_1/2 - 1$ rectangles and R_2 has $n_2/2 - 1$ rectangles. We conclude that R has $n_1/2 - 1 + n_2/2 - 1 = (n_1 + n_2)/2 - 1 = (n+2)/2 - 2 = n/2 - 1$ rectangles. ■

Rubric: 4 points. These are not the only correct solutions!!

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

Solution: We construct a degenerate trapezoidal decomposition using a sweep-line algorithm. At each vertical edge, we extend vertical segments through the interior of P, possibly in both directions or neither, until they touch P again.

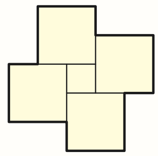


The only differences from the Shamos-Hoey algorithm are (1) the sweep BST can directly use y-coordinates of the horizontal edges as search keys, and (2) at each vertical edge, we perform two update operations on the sweep BST instead of one: either two insertions, two deletions, or one of each. The algorithm still runs in $O(n \log n)$ time. ■

Rubric: 2 points. This is not the only correct solution.

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

Solution: Here is a minimal counterexample:



This example also shows that the second case of the inductive proof of part (b) is actually necessary; not every rectangulation has a *guillotine cut*. ■

Rubric: 2 points

5. Prove that there are constants $0 < \alpha < 1$ and $\Delta > 1$ with the following property: In any planar straight-line graph with n vertices, there is an independent set of size αn vertices, each with degree at most Δ. How small can you make the ratio Δ/α? How quickly can you find such an independent set?

Solution: We follow an argument first published by David Kirkpatrick in 1983. Let G be a planar map with n vertices, and fix an integer $\Delta > 6$. Euler's formula $V - E + F = 2$ implies that G has at most $3n - 6$ edges (with equality if every face of G is a triangle). Thus, the sum of the vertex degrees is at most $6n - 12$. it follows that at least half of the vertices of G have degree less than 12. We can greedily construct an independent set of low-degree vertices in G as follows.

```
GreedyIndepSet(G, Δ):
  S ← ∅
  for every vertex v of G
      if deg(v) < 12 and v is unmarked
          add v to S
          mark v and every neighbor of v in G
  return S
```

The algorithm spends $O(1)$ time at every vertex and therefore runs in $O(n)$ time. By construction, the final set S is an independent set, and every vertex in S has degree at most 11. Moreover, S contains at least $1/12$ of the vertices with degree less than 12, so by our earlier analysis, $|S| > n/24$.

This construction gives us the ratio $\Delta/\alpha = 11 \cdot 24 = 254$. Replacing 12 with 8 everywhere, the same analysis gives us an independent set of at least $n/32$ vertices of degree at most 7, giving us a slightly better ratio $\Delta/\alpha = 224$.

We can improve our analysis slightly by assuming without loss of generality that G is a *maximal* planar graph; every vertex has degree at least 3. Suppose G has k vertices of degree less than 8. Then the sum of vertex degrees in G is least $3k + 8(n - k) = 8n - 5k$ but still less than $6n$, which implies $k > 2n/5$. Our greedy algorithm gives us an independent set of at least $n/20$ vertices of degree at most 7, giving us the ratio $\Delta/\alpha = 140$.

This can be further improved using the infamous Four-Color Theorem. The vertices of G can be colored with at most four colors, so that every edge has endpoints with two different colors. Suppose there are at most k vertices of each color with degree less than 8. Then the sum of vertex degrees is at least $3 \cdot 4k + 8(n - 4k) = 8n - 20k < 6n$, which implies $k > n/10$. Thus, some color class contains an independent set of at least $n/10$ vertices of degree at most 7, giving us the ratio $\Delta/\alpha = 70$.

Unfortunately, the fastest algorithm known to 4-color planar graphs takes $O(n^2)$ time (with a *huge* hidden constant). However, we can compute a 6-coloring in $O(n)$ time using a simple greedy algorithm; this leads to an independent set of at least $n/15$ vertices of degree at most 7, giving us the ratio $\Delta/\alpha = 105$. ■

Rubric: 10 points = 2 for explicit constants α and Δ + 5 for proof + 3 for algorithm. These are not the only correct solutions, or necessarily even the best solutions. No penalty for slower algorithms or larger ratios Δ/α. (I really should have asked about $\Delta/\lg\alpha$.)