1. For a set of objects in 2 dimensions, which of the following examples are degenerate?

Two of the below questions were potentially poorly worded. Given a point AND a line, if the point happens to lie on that line then this is a degenerate case. Given three points, AND a circle... the fact that the three points lie on that circles is degenerate. Given just three points, it is NOT a degenerate case when those three happen to be co-circular (because all triples of points are co-circular).

(a) 3 points on a line. [Degenerate Position]
(b) 3 points all in a box, (ie, all satisfying $0 < x < 1, 0 < y < 1$). [General Position]
(c) 3 points lying on a circle. [General Position], see note above
(d) 3 points lying inside a circle of radius 1. [Degenerate Position]
(e) 2 lines that are parallel. [Degenerate Position]
(f) a point and a line that intersect. [Degenerate Position], see note above
(g) a set of line segments, none of which intersect. [General Position]
(h) a set of line segments whose left endpoints all are the same point. [Degenerate Position]

2. Given a polygon $P$ with $n$ sides and a point $c$ inside that polygon, show how to compute the region that is inside $P$ and visible to $c$ in polynomial time.

Given: a set of line segments $[p_1, q_1], [p_2, q_2], \ldots, [p_n, q_n]$.

**Algorithm Overview:** We will do an angular plan sweep around the point $c$. We will keep track of every line segment that intersects a ray starting at $c$. This set of segments only changes when our ray intersect a segment endpoint.

**Event Queue:** The events occur when our sweep ray touches a new segment endpoint. To create the event queue, we compute $\theta_r$ for every point $r \in \{p_1, p_2, \ldots, q_1, q_2, \ldots\}$ the angle between the vectors $(0, -1), (\vec{r} - \vec{c})$. The sorted list of values $\theta_r$, for all $r \in \{p_1, p_2, \ldots, q_1, q_2, \ldots\}$ is ordered list of events that we will consider. Since these events never change, and we don’t add any new ones, then we can store this list as a linked list.
**Sweep ray initialization**: We start with a sweep ray whose origin is $c$ and which points in direction $(0, -1)$. Compute the intersection of this ray with every segment (some may not intersect). Sort the intersections by the distance of the intersection point from $c$. The sweep ray will maintain a list of segments that it currently intersects, sorted by distance of intersection point.

**Event Processing**: Each event corresponds to the sweep ray moving around until it intersects the endpoint of a segment $s$. There are two cases. If that segment is already in the sweep ray, then this is an “end segment” event. Otherwise, this is a “new segment” event.

- **New Segment** compute the point of intersection of the ray with $s$, and insert this segment into the sweep ray (in the correct place in the ordering by distance from $c$). If this segment is now the closest of all segments, output two new points of the start shaped polygon: first, the intersection point of the ray with the old closest segment, and also the point the sweep line just got to (the intersection of the ray with our newest segment). This takes a total of $(\log n)$ time.

- **End Segment** If $s$ is currently the closest segment, output the intersection of the ray with $s$, and the intersection of $s$ with the next closest segment. Then, delete $s$ from the sweep ray.

3. Define an efficient algorithm for determining the area of a simple polygon containing $n$ vertices. What is the running time of your algorithm?

   Use the triangulation algorithm in class to break the polygon into triangles. Compute the area of each triangle. Sum the results. Polygon triangulation takes $O(n \log n)$ time, and returns $O(n)$ triangles. The area of a triangle can be computed in constant time. So the total time is $O(n \log n)$.

4. **Diameter and width**: Define the diameter of a set of points to be the largest distance between any two points in the set.

   (a) **Prove that the diameter of a set is achieved by two vertices of the convex hull of the set.**

   Proof by contradiction. Let $p$ and $q$ be two points of the set who have the maximum distance apart. Assume, for a contradiction, that $q$ is not on the convex hull. Since $q$ is not on the convex hull, then there are points in every direction from $q$ that are still inside the convex hull. So lets continue the line from $p$ to $q$ until

---

1This is actually a little bit subtle. These distances have changed from the last time we computed this distance at the old position of the sweep ray, BUT the order of these edges hasn’t changed. To insert this element into a balanced binary tree, we may have to re-evaluate the intersection points of up to $\log n$ of the other segments intersecting the ray.
it hits an edge of the convex hull. This line is longer than the line from p to q. Now let's slide this line towards the two endpoints of the convex hull segment it intersects. In at least one direction, this line still grows longer, so the line from p to one of the endpoints of this convex hull segment is longer than p to q; a contradiction of the assumption that p and q have the maximum distance apart.

(b) A line of support to a set is a line L that touches the hull and has all points on or to one side of L. Prove that the diameter of a set is the same as the maximum distance between parallel lines of support for the set.

Proof: Two parts. The max distance between parallel lines of support is at least the diameter. Choose the two points with greatest distance in the set (p,q). By part one, they must be on the convex hull. Not only are they on the hull, but the angle (p,q, either neighbor of q) has to be less that 90 degrees, otherwise the segment from (p to “that neighbor of q”) would be longer than p,q. Therefore, the parallel lines that touch p and q and are perpendicular to the segment p,q are parallel lines of support, and the distance between this pair of lines is the same as the distance between p and q.

Part two: The max distance between a pair of lines is at most the diameter. Suppose not. Then there are two lines of support whose distance greater than the diameter. These lines must contain points from the set (by defn. of line of support). These points must then be more than the diameter apart, which contradicts the defn. of diameter.

(c) Two points a and b are called antipodal if they admit parallel lines of support: there are parallel lines of support through a and b. Develop an algorithm for enumerating (listing) all antipodal pairs of a set of points in two dimensions.

(sketch:) Find the convex hull. On that hull, choose one point. walk around and find its anti-podal point (using a comparison of hull edge slopes). Then walk the pair around the hull, moving whichever one of the points can be moved (why can only one be moved?). Prove that you didn’t miss any anti-podal pairs.

(d) Define the width as the minimum distance between parallel lines of support. Develop an algorithm for computing the width of a set of points in two dimensions.

(sketch:) Compute distance between every antipodal pair defined in the last problem. Prove that the smallest of these distances must be the minimum width.