Exercise 2

Motivation
We extend our study of the convex hull algorithm.

A Third $O(n \log n)$ algorithm
For simplicity, we restrict attention to the upper hull $U$, i.e., the part of the hull visible from $y = +\infty$. We also assume for simplicity that $x$-coordinates are pairwise distinct.
Let $v_1, v_2, \ldots, v_k$ be the vertices of the upper hull in increasing order of $x$-coordinate.

1. Given a point $p$, we want to determine whether $p$ lies above $U$. Show how to solve this problem by binary search in $O(\log k)$ steps.

2. Assume $p$ lies above $U$. We want to determine the tangents of $p$ with respect to $U$. Show how to solve this task in $O(\log k)$ steps using binary search.

3. If the vertices $v_1$ to $v_k$ are stored in an array of size $k$, binary search is easy to realize. However, the addition of $p$ is non-trivial. What data structure is appropriate so that binary search can be carried out efficiently and a new point can be added efficiently to the data structure? You should be able to add a point to an upper hull of $n$ points in time $O(\log n)$.

Examples of Nonrobustness
Study the web-page

1. KM thinks that this web-page is an excellent example of experimental research in CS. Do you agree?

2. Download nonrobust_06.tgz and install it on your machine.¹

3. Run fp_scope ../data/vis_fp_pts_1.bin -o test.ppm and inspect test.ppm in gimp. You should recognize the picture.

¹Should compile on linux with g++. On Windows we encourage to use Cygwin. If there are compile errors (some known for g++-4.3.1), let us know, and we provide a fix.
4. In the last exercise, we proved that

$$\text{sign } \det \begin{pmatrix} 1 & p_x & p_y & p_x^2 + p_y^2 \\ 1 & q_x & q_y & q_x^2 + q_y^2 \\ 1 & r_x & r_y & r_x^2 + r_y^2 \\ 1 & s_x & s_y & s_x^2 + s_y^2 \end{pmatrix}$$

computes the side-of-circle predicate of four points. Start with the program `fp_scope` and design an experiment for the side of circle predicate. Try at least two kinds of data sets:

- The defining points of the circle are nicely spread over the boundary of the circle.
- The defining points of the circle lie close together.

In each case, the query point should range over a region intersecting the circle. Try to explain your experimental findings (along with a pictures).

5. Repeat the previous exercise for the side-of-wedge predicate. Let \( p, q, \) and \( r \) be three points in the plane. A point \( z \) lies in the wedge if \( z \) lies to the left of the line \( \ell(p, q) \) and to the right of the line \( \ell(p, r) \). It lies on the wedge, if it lies on one of the lines, and it lies outside the wedge otherwise.

$$\text{wedge}(p, q, r, z) = \text{orient}(p, q, z) \cdot \text{orient}(p, q, z)$$

$$\text{wedge}(p, q, r, z) = \text{sign} \begin{pmatrix} 1 & p_x & p_y \\ 1 & q_x & q_y \\ 1 & r_x & r_y \\ 1 & s_x & s_y \end{pmatrix} \cdot \begin{pmatrix} 1 & p_x & p_y \\ 1 & r_x & r_y \\ 1 & s_x & s_y \end{pmatrix} \cdot \begin{pmatrix} 1 & q_x & q_y \\ 1 & r_x & r_y \\ 1 & s_x & s_y \end{pmatrix}$$

Try two kinds of query points: points near \( p \) and points near one of the two defining lines. Try to explain your experimental findings (along with pictures).

Have fun with the solution!