INF 4130 Exercise set 12, 6th Nov. 2013 w/proposed solutions

Exercise 1

Go through (in some detail) the process illustrated on slide 15 from the lecture: adding a new vertex and restoring the Delaunay property. Make sure you understand the process.

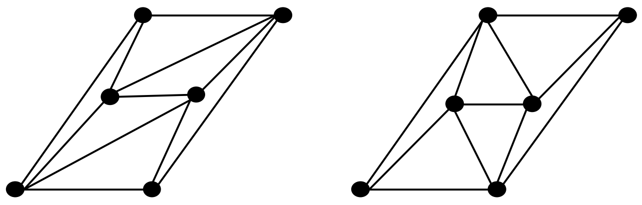
The steps:

1) Add new point p – inside one triangle T.
2) Draw all diagonals from p to corners of T.
3) Check all new triangles, exchange diagonal (opposing corner) if necessary.
4) Repeat 3) until no further change.

Studying it is left to the student.

Exercise 2

We are given six points (in $\mathbf{R}^2$), and two possible triangulations of them, as shown in the figure below.



Answer the following questions:

a) Which of the two triangulations is a Delaunay-triangulation?

Look at all circumcircles, and check that no point lies within. Or argue (more imprecise, perhaps, but one triangulation *is* Delaunay) based on the angles of the triangles.

b) Draw the Voronoi-diagram for the given points, and show that it matches your answer for question a.

Left to the student. Note that all borders are perpendicular to the corresponding edge, and at the midpoint between the points.

c) Show that you can get from the triangulation that is non-Delaunay to one that is, by doing the "Delaunay-trick" one or more times.

Left to the student.

d) Verify that your answer for question a also is correct with the "angel-definition" (max-of-min) of a Delaunay-triangulation.

Left to the student. See also a).

Exercise 3

We are checking whether two adjacent triangles that make up a convex quadrilateral (a convex polygon with four sides) locally has the Delaunay property, or if we have to exchange the diagonal with the opposite one. On the slides it says that we have to calculate a determinant to check this. One could, however, wonder if always choosing the shortest diagonal will work. In most cases it is this one we chose. Find a counterexample where choosing the shortest diagonal does not work.

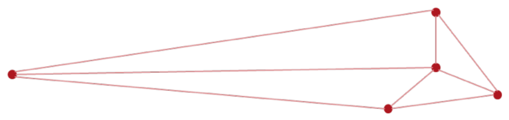
Hint: Start with four points placed somewhat irregularly on a circle, and then move one of the points.

Draw four points along the circumference of a circle. The important point is that the two diagonals should have different lengths. (If one is the diameter, it is easy to convince oneself that the other must be shorter…) Then move one of the points on the shorter diagonal ever so slightly outside the circle. (Its diagonal must remain shorter than the other one.) In the quadrangle defined by these four points we must now choose the long diagonal, the circumcircle for the short one would contain the opposing corner.

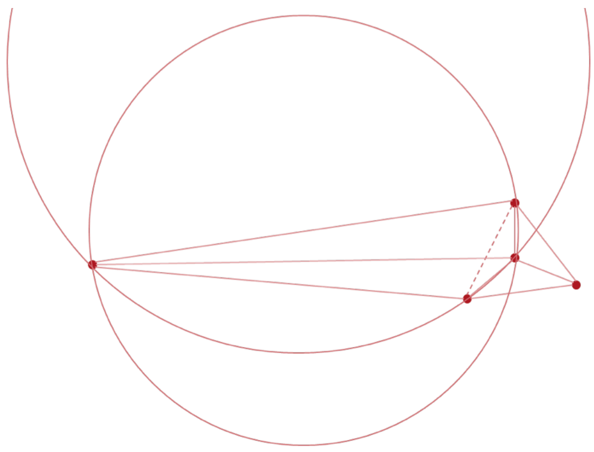
Exercise 4

Assume you have a reasonably gathered set of points *P* in the plane ($\mathbf{R}^2$), and a point *q* far away from *P*. Show that the triangulation of $P \cup \{q\}$ becomes a triangulation of *P* if you remove *q* and all the edges connected to it.

Removing a point could leave us with a triangulation that is illegal, like in the example below, where the triangulation of the remaining points is non-convex (concave).



We therefore restrict our triangulation to be a Delaunay triangulation. The crux of such a proof is that the triangulation of the remaining points must be convex (i.e. all edges in the convex hull must be part of the triangulation. Removal of the dotted edge below leads to a non-Delaunay triangulation. If both its endpoints must be outside the other circumcircle, we get a convex polygon.



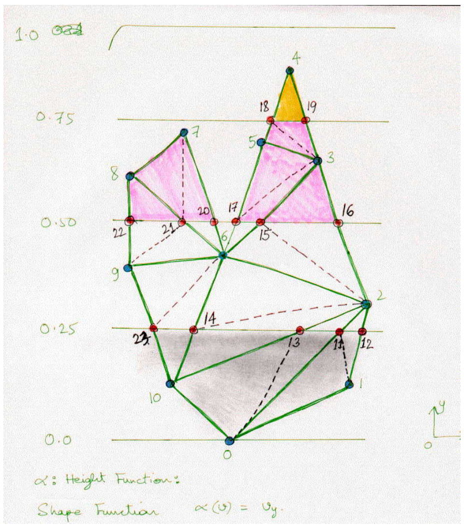
Exercise 5

a) Is the polygon to the right a convex polygon or not? Justify your answer.

No. The line segments 7-4 and 19-22, for instance, do not lie *inside* the polygon.

b) If it isn't convex (it would then be *concave*), can it be divided into a number of convex (sub-)polygons? And if so, what is the smallest number of polygons you need?

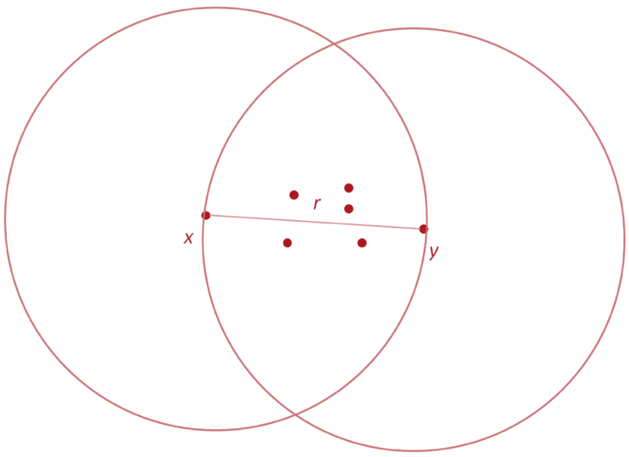
Divide the polygon along the line 6-14-10. Fewer than two sub-polygons is, of course, not possible.



Exercise 6

Given a set of points *Q*, show that the pair of points with the largest pairwise distance *must* be corners in CH(*Q*) – the convex hull of *Q.*

Let *x* and *y* be the points with the largest pairwise distance. All other points must lie within the *luna* describes by the intersection of the two circles around *x* and *y* with radius equal to the distance. Specifically must all projections of the other points onto the line between *x* and *y* lie between *x* and *y,* on the line; the result follows.



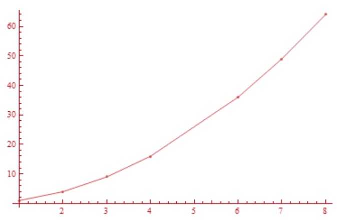
Exercise 7

Solve Exercise 8.32 in the textbook: Show that finding the convex hull of *n* points cannot be done faster than *O*(*n* log *n*). Or more correctly: Show that finding the convex hull is Ω(*n* log *n*).

Hint: Show that sorting *n* values cannot be faster than finding the convex hull of *n* points.

We show this by making a reduction from SORTING to CONVEX HULL. (Usually we transform decision problems, but the principle is the same, with a small interpretation of the answer from the sub-routine, since it isn't simply YES or NO).

The idea is to transform the points to be sorted into points describing a convex polygon, so that a counter clockwise ordering of the points gives us the sorted order. We transform all numbers *n* into points (n, n^2) (describing a part of a parabola, which is convex).



The procedure for SORTING with a procedure for CONVEX HULL as a subroutine would look like this:

```
proc Sorting(N)
{
   P = Transform(N)             // Transform number: n to point (n,n^2)
   H = ConvexHull(P)
   l = <the h in H with lowest y-coordinate>                 // O(n)
   S = <read x-coordinates of corners in H, l first>         // O(n)
   return (S)
}
```

This reduction tells us that we cannot solve CONVEX HULL faster than SORTING. Informally since a weak algorithm cannot solve a hard problem, so CONVEX HULL must be as hard (time consuming) as SORTING.

We have shown SORTING ∝ CONVEX HULL (Time: SORTING $\leq_{pol}$ CONVEX HULL) .

+

Know that SORTING is Ω(*n* log *n*) (Time: *n* log *n* ≤ SORTING , actually C * *n* log *n*).

⇓

CONVEX HULL is Ω(*n* log *n*).

[END]