Chair of Computer Science I
Algorithms and Complexity
University of Würzburg

Würzburg, October 30, 2020

Dr. Philipp Kindermann
Dr. Myroslav Kryven

Homework Assignment #1
Computational Geometry (Winter Semester 2020)

Exercise 1

Recall that a subset A of the plane is called *convex*, if, for all points $u, v \in A$, the line segment $\overline{uv}$ is also contained in A. Let S be a finite set of points. Then we define the set

$$\mathcal{H}(S) := \{\, h \mid h \text{ is a closed half-plane and } S \subseteq h \}$$

of all closed half-planes that contain S. (A *closed* half-plane h contains its own boundary, the line ∂h). The *convex hull* $\mathrm{CH}(S)$ of S can be defined as the intersection $\bigcap \mathcal{H}(S)$ of all these half-planes. Prove that the convex hull fulfills the following properties. (For properties b) and c), we assume that S contains at least three non-collinear points. You may use the property that half-planes are convex.)

a) $\mathrm{CH}(S)$ is convex. **[2 points]**

b) Let $\mathcal{H}_2(S) := \{\, h \mid h \in \mathcal{H}(S) \text{ and } |\partial h \cap S| \geq 2 \}$ be the set of all closed half-planes that contain S and that have at least two points of S on their own boundary. Then

$$\mathrm{CH}(S) = \bigcap \mathcal{H}_2(S)\,.$$

[4 points]

c) $\mathrm{CH}(S)$ is a (convex) polygon of which all corners are points of S. **[4 points]**

Exercise 2

In Exercise 1 we have shown that the convex hull of a finite set of points in the plane is a polygon. We require the output of a convex-hull algorithm to be a list of the vertices of the corresponding polygon in clock-wise order.

a) Prove that every algorithm that computes the convex hull of n points needs a running time of $\Omega(n \log n)$. That means that the algorithm of the lecture is optimal in the sense of the asymptotic running time.

Hint: Use the property that *sorting* n keys (in certain computer models) requires running time $\Omega(n \log n)$. **[3 points]**

b) Let P be a simple, not necessarily convex polygon in the common list representation. (In a *simple* polygon the edges are crossing-free.) Develop an algorithm that computes the convex hull of the vertices of this polygon in $O(n)$ time. Explain why this is not a contradiction to the claim in subexercise a). **[5 points]**

c) Is there also a linear-time algorithm if we drop the requirement of simplicity in subexercise b)? **[2 points]**

This assignment is due on November 11th. The solutions will be made available on November 13th.