Chair of Computer Science I
Algorithms and Complexity
University of Würzburg

Würzburg, June 04, 2020

Dr. Philipp Kindermann
Dr. Myroslav Kryven

Homework Assignment #6
Computational Geometry (Summer Semester 2020)

Exercise 1

Give an example of a set of n line segments with an order on them that makes the algorithm TrapezoidalMap, without the randomization (line 2), create a search structure of size $\Theta(n^2)$ and query time $\Theta(n)$. **[6 points]**

Exercise 2

Describe a randomized, incremental algorithm that calculates the intersection of a set H of n half-planes in $O(n \log n)$ time. The algorithm first chooses a random permutation $\langle h_1, \ldots, h_n \rangle$ of H. In every step $i = 1, \ldots, n$, it calculates the intersection $I_i := h_1 \cap \cdots \cap h_i$ of the first i half-planes in the permutation. Use the following witness structure to find out where to insert a half-plane: For every half-plane h_j with $j > i$, save a witness (one point). The witness is a corner of I_i that does not lie in h_j (If all the corners lie in h_j, then we can ignore h_j). For the sake of simplicity, assume that $h_1 \cap h_2 \cap h_3$ is bounded.

Hint: First, consider the running time that is required to create and delete the vertices of I_i ($i = 1, \ldots, n$). Then define a random variable X_i for the number of witnesses that change in step i. Use backwards analysis to determine the total running time $\sum_i X_i$ to manage the witness structure. **[14 points]**

This assignment is due on December 16th. The solutions will be made available on December 18th.