Lehrstuhl für Informatik I
Effiziente Algorithmen und wissensbasierte Systeme
Universität Würzburg

Würzburg, October 29, 2014

Prof. Dr. Alexander Wolff
Dipl.-Inform. Philipp Kindermann

Homework Assignment #4
Computational Geometry (Winter Term 2014/15)

Exercise 1

We consider the following problem in the plane: Given a polygon P and a convex polygon Q, we want to scale P such that, after displacement, P still completely fits into Q and is as large as possible. Thus, we are looking for a vector $\vec{v} \in \mathbb{R}^2$ that allows for the largest possible scaling factor $\lambda > 0$ such that $\lambda \cdot P + \vec{v} \subseteq Q$.

Give a linear program that solves this problem. **[5 points]**

Exercise 2

Algorithm 1: RandMax

```
Input: Finite set A ⊂ ℝ
Output: Maximum max_{a∈A} a of A
if |A| = 1 then
    return the only element a ∈ A
else
    a = randomly chosen element from A
    b = RandMax(A \ {a})
    if b ≥ a then
        return b
    else
        unnecessarily test every element of A \ {a} to ensure that a is maximum
        return a
```

Consider Algorithm 1, which calculates the maximum of a set of numbers. First, give a tight bound on the worst-case running time of the algorithm. Then, consider the random choice of the element a and show that the expected running time is strictly less than the worst-case running time of the algorithm. **[6 points]**

Exercise 3

Consider n trains that are moving on parallel tracks. Train z_i ($i = 1, \dots, n$) is moving with constant speed v_i and is at position p_i at time 0. At time 0, only time starts; all trains already run at their respective speeds. Design an algorithm that lists, in $O(n \log n)$ time, for a given number $t_{stop} > 0$, any train that has been ahead of the others at least once in the time interval $[0, t_{stop}]$.

Hint: Convert the problem into a geometric problem; then solve it with algorithms known from the lecture. **[9 points]**

This assignment is due at the beginning of the next lecture, that is, on November 05 at 10:15. Solutions will be discussed in the tutorial on Friday, November 07, 14:00–15:30 in room SE I.