TECHNISCHE UNIVERSITÄT KAISERSLAUTERN

Department of Computer Science
Sebastian Wild

Issue Date: 31.05.2017
Version: 2017-05-31 12:45

6th Exercise sheet for Advanced Algorithmics, Summer 17

Hand In: *Until Wednesday, 07.06.2017, 12:00 am, hand-in box in 48-4 or via email.*

Problem 15

20 points

A one-sided-error Monte Carlo algorithm A might give a wrong answer ever other time. We could use majority voting to amplify the probability, but that would not exploit the one-sidedness.

Describe how we can reduce the error probability to an arbitrary given constant $\delta > 0$, and compute the running time of the resulting method.

What is the running time to obtain a correct result *with high probability*? Compare your result to the majority-voting result from class for two-sided error Monte Carlo methods.

Problem 16

20 + 30 points

Recall the randomized complexity classes from class.

Prove the following relations.

a) $\mathcal{P} \subseteq \text{ZPP} \subseteq \text{RP} \subseteq \text{BPP} \subseteq \text{PP}$

b) $\text{RP} \subseteq \mathcal{NP}$

Hint: Recall the *probabilistic method*.

Problem 17

10 + 30 points

a) Prove that every algorithm that randomly shuffles a given list of n items so that afterwards all possible orderings are equally likely must use $\Theta(n \log n)$ random bits.

b) Design a (randomized) algorithm A that generates a random permutation of the numbers $1, \ldots, n$. Each permutation is to have the same probability.

Argue that your algorithm has the desired property and determine $\mathbb{E}\text{-Time}_A(n)$ as well as the *expected* number of random bits to generate a permutation of length n.

Can you find a method with optimal number of random bits (asymptotically and in expectation)?