Effiziente Algorithmen

Summer term 2012

Prof. Dr. Georg Schnitger, Bert Besser

Arbeitsgruppe Theoretische Informatik, Institut für Informatik

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

Assignment 2

Take home: 04/23/2012
Submit: 04/30/2012

Exercise 2.1. (4+4) *From Monte Carlo to Las Vegas*

- Let $0 < \epsilon_2 < \epsilon_1 < 1$ be given constants. Let an algorithm A output a correct solution with probability at least $1 - \epsilon_1$.

  State a sufficient and constant number k of repetitions of A such that it outputs at least one correct solution with probability at least $1 - \epsilon_2$.

- Let algorithm A run in expected time $T(n)$ and output a correct solution with probability at least p. Let furthermore A' be an algorithm that checks the correctness of A's output in worst-case time $T'(n)$.

  State an algorithm that always outputs a correct solution and that runs in expected time $\frac{T(n)+T'(n)}{p}$.

Exercise 2.2. (8) *Bounding the worst case runtime of a Monte Carlo algorithm*

Let A be a Monte-Carlo algorithm, i.e. a two-sided error algorithm, that either accepts or rejects an input and that decides incorrectly with probability at most $\frac{1}{4}$. A runs in *expected* time $T(n)$.

State an algorithm B that uses A as a subroutine, that runs in *worst-case* time at most $c \cdot T(n)$ where c is a constant and that decides incorrectly with probability at most $\frac{1}{3}$.

Hint: Markov's inequality.

Exercise 2.3. (6+2) *Randomized, hierarchical majority decisions*

Consider a complete ternary tree T. Each leaf has depth d and each internal node has exactly three children. T has $n = 3^d$ leaves, each labelled with either 0 or 1. An internal node is supposed to get the label of the majority of its children. We seek the label of T's root.

- Describe a randomized algorithm that computes the root's label such that the number of inspected leaf labels is as small as possible. Analyze the expected number of inspected leaf labels.

- What is the minimum number of inspected leaf labels for any nondeterministic algorithm?