Prof. Dr. Christian Scheideler
Universität Paderborn

Paderborn, April 27, 2018

Randomized Algorithms

SS 2018

Homework Assignment 3

Problem 9:
Prove Theorem 3.10.

Problem 10:
Consider the situation that we have n processes, where process i initially stores some number $x_i \in \mathbb{N}$. The goal for the n processes is to compute the minimum of these numbers. In order to do so, they execute the following algorithm in synchronized rounds:
In each round, each process i contacts a process j uniformly and independently at random and requests the number x_j currently stored in j. It then sets $x_i := \min\{x_i, x_j\}$.
Our goal is to prove that at most $O(\log n)$ rounds are needed till all processes know the minimum. For that we separate the analysis into three stages, where the following needs to be shown:

(a) With probability at least $1 - 1/n$, after $O(\log n)$ rounds at least $c \log n$ many processes will know the minimum (where c is a sufficiently large constant so that (b) works).

(b) If at the beginning of a round, k many processes know the minimum, where $c \log n \leq k \leq n/2$, then with probability at least $1 - 1/n$ at least $(5/4)k$ processes will know the minimum at the end of the round.

(c) If at the beginning of a round, k many processes do *not* know the minimum yet, where $k \leq n/2$, then with probability at least $1 - 1/n$ at most $(3/4)k$ processes will not know the minimum yet at the end of the round.

Hint: properly define binary random variables and use the Chernoff bounds in each stage.

Problem 11:
Consider the following online job scheduling problem: The input sequence σ consists of a sequence of jobs $J_1, \dots, J_n$, where each J_i requires some time $t_i \in \mathbb{N}$ to be processed. The task of the online algorithm is to assign each job to one of m machines, $M_1, \dots, M_m$, so that the makespan is minimized, where the makespan is the maximum over all machines of the total time needed by a machine to process the jobs assigned to it.
Suppose we use the simple online algorithm RANDOM: for each job J_i, choose a machine uniformly and independently at random.

(a) Determine the expected total time needed by a specific machine M_i to process the jobs assigned to it.

(b) Determine the variance of the total time needed by a specific machine M_i to process the jobs assigned to it.

(c) Find an upper bound on the expected makespan of algorithm RANDOM (recall that the makespan is the *maximum* total runtime assigned to a machine). For that, try to apply either Chebychev's inequality or the Chernoff bounds to bound the deviation from the expected total runtime for a single machine, and conclude from that on the makespan. (For the Chernoff bounds, you may assume that it also holds if the variables X_i satisfy $X_i \in [0, 1]$ instead of $X_i \in \{0, 1\}$. For that to hold, one may have to renormalize the X_i's if, for example, X_i is defined as $X_i \in \{0, t_i\}$.)