Dr. Ulf-Peter Schroeder

Paderborn, 19. Juli 2017
Abgabe: Keine!

Übungen zur Vorlesung
Methoden des Algorithmenentwurfs
SS 2017
Blatt 13

Aufgabe 34:
Consider a BALLS-AND-BINS experiment with $2n$ balls but only two bins. As usual, each ball independently selects one of the two bins, both bins equally likely. The expected number of balls in each bin is n. In this problem, we explore the question of how big their difference is likely to be. Let X_1 and X_2 denote the number of balls in the two bins, respectively. (X_1 and X_2 are random variables.) Prove that for any $\epsilon > 0$ there is a constant $c > 0$ such that the probability $Pr[X_1 - X_2 > c\sqrt{n}] \leq \epsilon$.

Aufgabe 35:
Some people designing parallel physical simulations come to you with the following problem. They have a set P of k basic processes and want to assign each process to run on one of two machines, M_1 and M_2. They are then going to run a sequence of n jobs, $J_1, \ldots, J_n$. Each job J_i is represented by a set $P_i \subseteq P$ of exactly $2n$ basic processes which must be running (each on its assigned machine) while the job is processed. An assignment of basic processes to machines will be called perfectly balanced if, for each job J_i, exactly n of the basic processes associated with J_i have been assigned to each of the two machines. An assignment of basic processes to machines will be called NEARLY BALANCED if, for each job J_i, no more than $\frac{4}{3}n$ of the basic processes associated with J_i have been assigned to the same machine.

(a) Show that for arbitrarily large values of n, there exist sequences of jobs $J_1, \ldots, J_n$ for which no perfectly balanced assignment exists.

(b) Suppose that $n \geq 200$. Give an algorithm that takes an arbitrary sequence of jobs $J_1, \ldots, J_n$ and produces a nearly balanced assignment of basic processes to machines. Your algorithm may be randomized, in which case its expected running time should be polynomial, and it should always produce the correct answer.

Aufgabe 36:
For the problem BIN PACKING you are given n objects with weights $w_1, \ldots, w_n \in [0, 1]$ and an unlimited number of bins with capacity 1 each. The objective is to use as few bins as possible. More formally, we are looking for the smallest number k for which the set $\{1, \ldots, n\}$ can be partitioned into k sets $S_1, \ldots, S_k$ such that for each $i \in \{1, \ldots, k\}$ the equation $\sum_{j \in S_i} w_j \leq 1$ is satisfied. In the online version of BIN PACKING every object i has to be assigned to a bin without knowing the number n and without knowing the weights of the object $i+1, \ldots, n$. Describe a strict 2-competitive online algorithm for BIN PACKING and prove its competitive ratio.