Research Institute for Discrete Mathematics
Approximation Algorithms
Summer term 2017

Prof. Dr. S. Hougardy
D. Rotter

Exercise Set 6

Exercise 6.1:
Let $A = (a_i)_{1 \leq i \leq p}$ and $B = (b_j)_{1 \leq j \leq q}$ be two inputs of the BIN PACKING problem. We write $A \subseteq B$ if there are indices $1 \leq k_1 < k_2 < \cdots < k_p \leq q$ with $a_i = b_{k_i}$ for $1 \leq i \leq p$. An algorithm for the BIN PACKING problem is called monotone if for inputs A and B with $A \subseteq B$ the algorithm needs at least as many bins for B as for A. Show:

(i) NEXT FIT is monotone. (2 points)

(ii) FIRST FIT is not monotone. (2 points)

Exercise 6.2:
Show that if all item sizes $a_1, \ldots, a_n$ satisfy $a_i > \frac{1}{3}$ then BIN PACKING can be solved optimally in polynomial time.

(2 points)

Exercise 6.3:
Show that if all item sizes are of the form $a_i = k \cdot 2^{-b_i}$ for some $b_i \in \mathbb{N}, i = 1, \ldots, n$ and some fixed $k \in \mathbb{N}$ then the FIRST FIT DECREASING algorithm always finds an optimum solution.

(4 points)

Exercise 6.4:
Consider the following MULTIPROCESSOR SCHEDULING PROBLEM: Given a finite set A of tasks, a number $t(a) \in \mathbb{N}$ for each $a \in A$ (the *processing time*) and a number m of processors, find a partition $A = \dot{\bigcup}_{i=1}^{m} A_i$ of A into m pairwise disjoint sets A_i such that $\max_{i=1}^{m} \left\{ \sum_{a \in A_i} t(a) \right\}$ is minimum.

(i) Consider a greedy algorithm that successively assigns jobs (in an arbitrary order) to the currently least used machine. Show that this is a 2-approximation algorithm.

(ii) Show that the modification of the greedy algorithm in which jobs are first sorted by $t(a)$ in non-increasing order and are then processed in that order is a $\frac{3}{2}$-approximation.

(3+3 points)

Deadline: Thursday, June 1st, before the lecture.

The student council of mathematics will organize the math party on 01/06 in N8schicht. The presale will be held on Mon 29/05, Tue 30/05 and Wed 31/05 in the mensa Poppelsdorf. Further information can be found at fsmath.uni-bonn.de