Homework 5

Released on November 07, 2016

1. Given n jobs with processing times $p_1, \ldots, p_n$, to be processed by a machine, assign the processing order of jobs such that the total completion time is minimized, *i.e.*, let C_i be the completion time of job i, the goal is to minimize $\sum_{i=1}^{n} C_i$. Give a greedy algorithm that solves the problem.

Hint: the proof idea is similar to the proof of the claim in s.12, week 8-2.

2. A minimal (inclusionwise) dependent set in a matroid is called a *circuit*. Let $(S, \mathcal{I})$ be a matroid and $I \in \mathcal{I}$, $x \in S$ such that $I + x \notin \mathcal{I}$.

(1) Prove that there exists a unique circuit $C \subseteq I + x$.

(2) Let $y \in C$ and $y \neq x$. Show that $I + x - y \in \mathcal{I}$ (using (1)).

3. What is the average-case running time of Increment(A)? (Use standard probabilistic analysis, you may use equation A.8, p.1148 from textbook)

4. Consider an ordinary min-heap data structure and a sequence n of Insert and Extract-Min. Give a potential function Φ such that the amortized cost of Insert is $O(\log n)$ and the amortized cost of Extract-Min is $O(1)$, and show that it works.

5. Design a data structure to support the following two operations for a dynamic multiset S of integers, which allows duplicate values:

- Insert(S, x) inserts x into S.
- Delete-Larger-Half(S) deletes the largest $\lceil |S|/2 \rceil$ elements from S.

Explain how to implement this data structure so that any sequence of m Delete-Larger-Half and Insert operations runs in $O(m)$ time.