Tutorial 2: CS21003 Algorithms 1

Prof. Partha Pratim Chakrabarti and Palash Dey
Indian Institute of Technology, Kharagpur

February 3, 2022

1. In the activity selection problem, the input is a set of n jobs each having a start time and end time. We have only one machine and it can execute one job at any point of time. Jobs once started can not be interrupted before it finishes. We need to compute a maximum cardinality subset of non-overlapping jobs. Find which of the following greedy algorithms are correct.

   (a) Choose the job x that starts first, discard jobs that overlap with x, and recurse.

   (b) Choose the smallest job x, discard jobs that overlap with x, and recurse.

   (c) Choose the job x that overlaps with the least number of other jobs, discard jobs that overlap with x, and recurse.

   (d) Choose the job x that ends last, discard jobs that overlap with x, and recurse.

   (e) Choose the job x that starts last, discard all jobs that overlap with x, and recurse.

   (f) If no two jobs overlap, choose them all. Otherwise, discard the job with longest duration and recurse.

   (g) Choose the job that finishes earliest, discard jobs that overlap with it, and recurse.

   (h) If no two jobs overlap, choose them all. Otherwise, discard a job that overlaps with the most other jobs and recurse.

   (i) If any job x completely contains another job, discard x and recurse. Otherwise, choose the job y that ends last, discard all classes that overlap with y, and recurse.

   (j) Let x be the class with the earliest start time, and let y be the class with the second earliest start time.

      ▷ If x and y are disjoint, choose x and recurse on everything but x.

      ▷ If x completely contains y, discard x and recurse.

      ▷ Otherwise, discard y and recurse.

2. Suppose there are n lectures with start and end times $S[1, \ldots, n]$ and $F[1, \ldots, n]$. Obviously, if two lectures overlap, then both of them cannot be conducted in a single hall. If two lectures do not overlap, then we are allowed to conduct them in the same hall. Design a greedy algorithm to compute the minimum number of lecture halls needed to conduct these n lectures.

3. Suppose we have n jobs with positive lengths $\ell_1, \ldots, \ell_n$ and positive deadlines $d_1, \ldots, d_n$. In a schedule σ (which is a permutation of $\{1, \ldots, n\}$), let $C_j(\sigma)$ be the finish time of job j in the schedule σ. The lateness $\lambda_j(\sigma)$ of a job j in the schedule σ is defined as $\max\{0, C_j(\sigma) - d_j\}$. Which of the following greedy algorithms outputs a schedule which minimizes total lateness $\sum_{i=1}^{n} \lambda_j(\sigma)$.

   (a) Schedule the jobs in non-decreasing order of deadline d_j.

   (b) Schedule the jobs in non-decreasing order of lengths ℓ_j.

   (c) Schedule the jobs in non-decreasing order of $d_j \ell_j$.

   (d) None of the above.

4. We have seen in the class that prefix-freeness of a binary code is a sufficient condition for unique decoding. Is prefix-freeenss a necessary condition for unique decoding?

5. What is the maximum number of bits that Huffman coding algorithm might use to encode a symbol. Show a tight example.
6. Suppose the frequencies of the symbols in an Huffman coding instance sums to 1. What is the maximum and minimum number of bits that will be used to encode a symbol having frequency 0.4 by the Huffman coding algorithm?