

CS180 Homework 5

Due: 4:00pm, 05/04/2016

1. Consider the problem of printing a paragraph with a mono-spaced font (all characters having the same width) on a printer. The input text is a sequence of n words of lengths l_1, l_2, \dots, l_n , measured in characters. We want to print this paragraph neatly on a number of lines that hold a maximum of M characters each. If a given line contains words i through j , where $i \leq j$, and we leave exactly one space between words, the number of extra space characters at the end of the line is $M - j + i - \sum_{k=i}^j l_k$, which must be non-negative so that the words fit on the line. We wish to minimize the sum, over all lines except the last, of the *cube* of sum of the numbers of extra space characters at the ends of lines. Give an algorithm to print a paragraph of n words on a printer in a way that minimizes the above sum. Analyze the running time and space requirements of your algorithm.
2. A palindrome is any string that is exactly the same as its reversal, like I, or DEED, or RACECAR. Describe and analyze an algorithm to find the length of the longest subsequence of a given string that is also a palindrome. For example, the longest palindrome subsequence of MAHDYNAMICPROGRAMZLETMESHOWYOUTHEM is MHYMRORMYHM, so given that string as input, your algorithm should output the number 11.
3. Let $S[1 \dots n]$ be an array of characters representing a string. We wish to break S into substrings by cutting it at specified positions. The positions are given in a Boolean array $P[1 \dots n-1]$, where $P[i] = 1$ means that S must be cut at position i and $P[i] = 0$ means that S must not be cut at position i . (Thus P functions as a mask.) For instance, if $S = \text{STRING}$ and $P = 10101$, then S must be cut at position 1, 3, and 5, giving the substrings S, TR, IN, G. The *cost* of a cut equals the length of the substring that is cut and the *total cost* of all cuts equals the sum of the costs of the individual cuts. The order in which cuts are made affects the total cost. For instance, if the order is 1, 3, 5, then the first cut costs 6 (the length of STRING), the second costs 5 (length of TRING), and the third costs 3 (length of ING), for a total cost is 14. If the order is 3, 1, 5, then the costs are 6, 3, and 3, for a total of 12. And so on. Given S and P , give a dynamic programming algorithm to compute the minimum total cost for cutting S according to P . The target time bound is $\Theta(n^3)$. Also give an algorithm to print out the locations of the corresponding cuts.
4. Recall the scheduling problem in the lecture in which we sought to minimize the maximum lateness. There are n jobs, each with a deadline d_i and a required processing time t_i , and all jobs are available to be scheduled starting at time s . For a job i to be done, it needs to be assigned a period from $s_i \geq s$ to $f_i = s_i + t_i$, and different jobs should be assigned nonoverlapping intervals. As usual, such an assignment of times will be called a *schedule*.

In this problem, we consider the same setup, but want to optimize a different objective. In particular, we consider the case in which each job must either be done by its deadline or not at all. We'll say that a subset J of the jobs is schedulable if there is a schedule for the jobs in J so that each of them finishes by its deadline. Your problem is to select a schedulable subset of maximum possible size and give a schedule for this subset that allows each job to finish by its deadline.

- (a) Prove that there is an optimal solution J (i.e., a schedulable set of maximum size) in which the jobs in J are scheduled in increasing order of their deadlines.
- (b) Assume that all deadlines d_i and required times t_i are integers. Give an algorithm to find an optimal solution. Your algorithm should run in time polynomial in the number of jobs n , and the maximum deadline $D = \max_i d_i$.

★ Homework assignments are **STRICTLY** due on the exact time indicated. Please submit a hard copy of your homework solution with your **Name, Bruin ID, Discussion Number**, clearly indicated on the first page. If your homework consists of multiple pages, please **staple** them together. Email attachments or other electronic delivery methods are not acceptable.

- ★ We recommend to use \LaTeX , \LyX or other word processing software for submitting the homework. This is not a requirement but it helps us to grade the homework and give feedback. For grading, we will take into account both the correctness and the clarity. Your answer are supposed to be in a simple and understandable manner. Sloppy answers are expected to receiver fewer points.
- ★ Unless specified, you should justify your algorithm with proof of correctness and time complexity.