

CSCI 570 Homework 3

Due Sunday Mar. 08 (by 23:30)

For all divide-and-conquer algorithms follow these steps:

1. Describe the steps of your algorithm in plain English.
2. Write a recurrence equation for the runtime complexity.
3. Solve the equation by the master theorem.

For all dynamic programming algorithms follow these steps:

1. Define (in plain English) subproblems to be solved.
2. Write the recurrence relation for subproblems.
3. Write pseudo-code to compute the optimal value
4. Compute its runtime complexity in terms of the input size.

1. Suppose we define a new kind of directed graph in which positive weights are assigned to the vertices but not to the edges. If the length of a path is defined by the total weight of all nodes on the path, describe an algorithm that finds the shortest path between two given points A and B within this graph.
2. For each of the following recurrences, give an expression for the runtime $T(n)$ if the recurrence can be solved with the Master Theorem. Otherwise, indicate that the Master Theorem does not apply.
 - $T(n) = 3T(n/2) + n^2$
 - $T(n) = 4T(n/2) + n^2$
 - $T(n) = T(n/2) + 2^n - n$
 - $T(n) = 2^n T(n/2) + n^n$
 - $T(n) = 16T(n/4) + n + 10$
 - $T(n) = 2T(n/2) + n \log n$
 - $T(n) = 2T(n/4) + n^{0.51}$
 - $T(n) = 0.5T(n/2) + 1/n$
 - $T(n) = 16T(n/4) + n!$
 - $T(n) = 10T(n/3) + n^2$
3. Suppose that we are given a sorted array of distinct integers $A[1, \dots, n]$ and we want to decide whether there is an index i for which $A[i] = i$. Describe an efficient divide-and-conquer algorithm that solves this problem and explain the time complexity.
4. We know that binary search on a sorted array of size n takes $O(\log n)$ time. Design a similar divide-and-conquer algorithm for searching in a sorted singly linked list of size n . Describe the steps of your algorithm in plain English. Write a recurrence equation for the runtime complexity. Solve the equation by the master theorem.
5. Given n balloons, indexed from 0 to $n - 1$. Each balloon is painted with a number on it represented by array *nums*. You are asked to burst all the balloons. If the you burst balloon i you will get $nums[i - 1] \times nums[i] \times nums[i + 1]$ coins. Here left and right are adjacent indices of i . After the burst, the left and right then becomes adjacent. You may assume $nums[-1] = nums[n] = 1$ and they are not real therefore you cannot burst them. Design a dynamic programming algorithm to find the maximum coins you can collect by bursting the balloons wisely. Analyze the running time of your algorithm.

Example. If you have the $nums = [3, 1, 5, 8]$. The optimal solution would be 167, where you burst balloons in the order of 1, 5, 3 and 8. The array $nums$ after each step is:

$$[3, 1, 5, 8] \rightarrow [3, 5, 8] \rightarrow [3, 8] \rightarrow [8] \rightarrow []$$

And the number of coins you get is $3 \times 1 \times 5 + 3 \times 5 \times 8 + 1 \times 3 \times 8 + 1 \times 8 \times 1 = 167$.

6. Given a non-empty string str and a dictionary containing a list of unique words, design a dynamic programming algorithm to determine if str can be segmented into a sequence of dictionary words. If $str = \text{"algorithmdesign"}$ and your dictionary contains "algorithm" and "design". Your algorithm should answer Yes as str can be segmented as "algorithmdesign". You may assume that a dictionary lookup can be done in $O(1)$ time.
7. A tourism company is providing boat tours on a river with n consecutive segments. According to previous experience, the profit they can make by providing boat tours on segment i is known as ai . Here, ai could be positive (they earn money), negative (they lose money), or zero. Because of the administration convenience, the local community requires that the tourism company do their boat tour business on a contiguous sequence of the river segments (i.e., if the company chooses segment i as the starting segment and segment j as the ending segment, all the segments in between should also be covered by the tour service, no matter whether the company will earn or lose money). The company's goal is to determine the starting segment and ending segment of boat tours along the river, such that their total profit can be maximized. Design a dynamic programming algorithm to achieve this goal and analyze its runtime.