

Algorithms Tutorial 4

Problems

1. You are traveling by a canoe down a river and there are n trading posts along the way. Before starting your journey, you are given for each $1 \leq i < j \leq n$ the fee $F(i, j)$ for renting a canoe from post i to post j . These fees are arbitrary. For example it is possible that $F(1, 3) = 10$ and $F(1, 4) = 5$. You begin at trading post 1 and must end at trading post n (using rented canoes). Your goal is to design an efficient algorithm which produces the sequence of trading posts where you change your canoe which minimises the total rental cost.
2. You are given a set of n types of rectangular boxes, where the i^{th} box has height h_i , width w_i and depth d_i . You want to create a stack of boxes which is as tall as possible, but you can only stack a box on top of another box if the dimensions of the 2-D base of the lower box are each strictly larger than those of the 2-D base of the higher box. Of course, you can rotate a box so that any side functions as its base. It is also allowable to use multiple instances of the same type of box.
3. You have an amount of money M and you are in a candy store. There are n kinds of candies and for each candy you know how much pleasure you get by eating it, which is a number between 1 and 100, as well as the price of each candy. Your task is to choose which candies you are going to buy to maximise the total pleasure you will get by gobbling them all.
4. Consider a 2-D map with a horizontal river passing through its centre. There are n cities on the southern bank with x -coordinates $a_1 \dots a_n$ and n cities on the northern bank with x -coordinates $b_1 \dots b_n$. You want to connect as many north-south pairs of cities as possible, with bridges such that no two bridges cross. When connecting cities, you are only allowed to connect the i^{th} city on the northern bank to the i^{th} city on the southern bank.

5. You are given a boolean expression consisting of a string of the symbols *true* and *false* and with exactly one operation *and*, *or*, *xor* between any two consecutive truth values. Count the number of ways to place brackets in the expression such that it will evaluate to true. For example, there is only 1 way to place parentheses in the expression *true* and *false* xor *true* such that it evaluates to true.
6. A company is organising a party for its employees. The organisers of the party want it to be a fun party, and so have assigned a fun rating to every employee. The employees are organised into a strict hierarchy, i.e. a tree rooted at the president. There is one restriction, though, on the guest list to the party: an employee and their immediate supervisor (parent in the tree) cannot both attend the party (because that would be no fun at all). Give an algorithm that makes a guest list for the party that maximises the sum of the fun ratings of the guests.
7. You have n_1 items of size s_1 and n_2 items of size s_2 . You would like to pack all of these items into bins, each of capacity C , using as few bins as possible.
8. You are given n activities and for each activity i you are given its starting time s_i , its finishing time f_i and the profit p_i which you get if you schedule this activity. Only one activity can take place at any time. Your task is to design an algorithm which produces a subset S of those n activities so that no two activities in S overlap and such that the sum of profits of all activities in S is maximised.
9. Your shipping company has just received N shipping requests (jobs). For each request i , you know it will require t_i trucks to complete, paying you d_i dollars. You have T trucks in total. Out of these N jobs you can take as many as you would like, as long as no more than T trucks are used in total. Devise an efficient algorithm to select jobs which will bring you the largest possible amount of money.
10. Again your shipping company has just received N shipping requests (jobs). This time, for each request i , you know it will require e_i employees and t_i trucks to complete, paying you d_i dollars. You have E employees and T trucks in total. Out of these N jobs you can take as many of them as you would like, as long as no more than E employees and T trucks are used in total. Devise an efficient algorithm to select jobs which will bring you the largest possible amount of money.

11. Because of the recent droughts, n proposals have been made to dam the Murray river. The i^{th} proposal asks to place a dam x_i meters from the head of the river (i.e., from the source of the river) and requires that there is not another dam within r_i metres (upstream or downstream). What is the largest number of dams that can be built? You may assume that $x_i < x_{i+1}$.
12. You are given an $n \times n$ chessboard with an integer in each of its n^2 squares. You start from the top left corner of the board; at each move you can go either to the square immediately below or to the square immediately to the right of the square you are at the moment; you can never move diagonally. The goal is to reach the right bottom corner so that the sum of integers at all squares visited is minimal.
 - a) Describe a greedy algorithm which attempts to find such a minimal sum path and show by an example that such a greedy algorithm might fail to find such a minimal sum path.
 - b) Describe an algorithm which always correctly finds a minimal sum path and runs in time n^2 .
 - c) Describe an algorithm which computes the number of such minimal paths.
 - d) Assume now that such a chessboard is stored in a read only memory. Describe an algorithm which always correctly finds a minimal sum path and runs in linear space (i.e., amount of read/write memory used is $O(n)$) and in time $O(n^2)$.
13. A palindrome is a sequence of at least two letters which reads equally from left to right and from right to left. Given a sequence of letters, find efficiently its longest subsequence (not necessarily contiguous) which is a palindrome. Thus, we are looking for a longest palindrome which can be obtained by crossing out some of the letters of the initial sequence without permuting the remaining letters.
14. A partition of a number n is a sequence $\langle p_1, p_2, \dots, p_t \rangle$ (we call the p_k *parts*) such that $1 \leq p_1 \leq p_2 \leq \dots \leq p_t \leq n$ and such that $p_1 + \dots + p_t = n$.
 - (a) Find the number of partitions of n in which every part is smaller or equal than k , where n, k are two given numbers such that $1 \leq k \leq n$.
 - (b) Find the total number of partitions of n .

first k columns $1 \leq k \leq n$. Each sub-problem can be assigned a type, which is the pattern occurring in the last column.

- (b) Using the notions of compatibility and type, give an $O(n)$ -time algorithm for computing an optimal placement.

17. Skiers go fastest with skis whose length is about their height. Your team consists of n members, with heights h_1, h_2, \dots, h_n . Your team gets a delivery of $m \geq n$ pairs of skis, with lengths l_1, l_2, \dots, l_m . Your goal is to design an algorithm to assign to each skier one pair of skis to minimise the sum of the absolute differences between the height h_i of the skier and the length of the corresponding ski he got, i.e., to minimise

$$\sum_{1 \leq i \leq n} |h_i - l_{s(i)}|$$

where $s(i)$ is the index of the skis assigned to the skier of height h_i .

18. You have to cut a wood stick into several pieces. The most affordable company, Analog Cutting Machinery (ACM), charges money according to the length of the stick being cut. Their cutting saw allows them to make only one cut at a time. It is easy to see that different cutting orders can lead to different prices. For example, consider a stick of length 10 m that has to be cut at 2, 4, and 7 m from one end. There are several choices. One can cut first at 2, then at 4, then at 7. This leads to a price of $10 + 8 + 6 = 24$ because the first stick was of 10 m, the resulting stick of 8 m, and the last one of 6 m. Another choice could cut at 4, then at 2, then at 7. This would lead to a price of $10 + 4 + 6 = 20$, which is better for us. Your boss demands that you design an algorithm to find the minimum possible cutting cost for any given stick.
19. For bit strings $X = x_1 \dots x_m$, $Y = y_1 \dots y_n$ and $Z = z_1 \dots z_{m+n}$ we say that Z is an interleaving of X and Y if it can be obtained by interleaving the bits in X and Y in a way that maintains the left-to-right order of the bits in X and Y . For example if $X = 101$ and $Y = 01$ then $x_1x_2y_1x_3y_2 = 10011$ is an interleaving of X and Y , whereas 11010 is not. Give an efficient algorithm to determine if Z is an interleaving of X and Y .
20. Some people think that the bigger an elephant is, the smarter it is. To disprove this you want to analyse a collection of elephants and place as large a subset of elephants as possible into a sequence whose weights are increasing but their IQs are decreasing. Design an algorithm which given the weights and IQs of n

elephants, will find a longest sequence of elephants such that their weights are increasing but IQs are decreasing.

21. You have been handed responsibility for a business in Texas for the next N days. Initially, you have K illegal workers. At the beginning of each day, you may hire an illegal worker, keep the number of illegal workers the same or fire an illegal worker. At the end of each day, there will be an inspection. The inspector on the i^{th} day will check that you have between l_i and r_i illegal workers (inclusive) (too few illegal workers and American economy suffers; too many illegal workers and the conservative Americans start voting in a way you do not like). If you do not, you will fail the inspection and pay a fine of \$1000. Design an algorithm that determines the fewest number of inspections you will fail if you hire and fire illegal employees optimally. (Note: the proceeds from all fines go towards building The Wall.)
22. Given an array of N positive integers, find the number of ways of splitting the array up into contiguous blocks of sum at most K .

Solution: Solve the subproblem: What is the number of ways I can split the first i elements into contiguous blocks size K . The base case is $\text{opt}(0) = 1$. The recursion is:

$$\text{opt}(i) = \sum \{\text{opt}(j-1) : 1 \leq j \leq i, \text{sum}(j, i) \leq K\},$$

where $\text{sum}(j, i)$ is the sum of all numbers from j to i inclusive. The complexity is $O(n^2)$, as each subproblem requires a $O(n)$ search.

23. There are N levels to complete in a video game. Completing a level takes you to the next level, however each level has a secret exit that lets you skip to another level later in the game. Determine if there is a path through the game that plays exactly K levels.
24. Given a sequence of n positive or negative integers A_1, A_2, \dots, A_n , determine a contiguous subsequence A_i to A_j for which the sum of elements in the subsequence is maximised.
25. Consider a row of n coins of values v_1, v_2, \dots, v_n , where n is even. We play a game against an opponent by alternating turns. In each turn, a player selects either the first or last coin from the row, removes it from the row permanently, and receives the value of the coin. Determine the maximum possible amount of money we can definitely win if we move first.