CS515: Algorithms and Data Structures, Fall 2017

Homework 2*

Due: Tue, 10/24/17

Homework Policy:

1. Students should work on homework assignments in groups of preferably three people. Each group submits to TEACH one set of typeset solutions, and hands in a printed hard copy in class or slides the hard copy under my door before the midnight of the due day. The hard copy will be graded.
2. The goal of the homework assignments is for you to learn solving algorithmic problems. So, I recommend spending sufficient time thinking about problems individually before discussing them with your friends.
3. You are allowed to discuss the problems with other groups, and you are allowed to use other resources, but you *must* cite them. Also, you must write everything in your own words, copying verbatim is plagiarism.
4. *I don't know policy:* you may write "I don't know" *and nothing else* to answer a question and receive 25 percent of the total points for that problem whereas a completely wrong answer will receive zero.
5. Algorithms should be explained in plain english. Of course, you can use pseudocodes if it helps your explanation, but the grader will not try to understand a complicated pseudocode.
6. More items might be added to this list. ☺

Problem 1. **[30 pts]** Design and analyze an algorithm that for a given positive integer n counts the number of different ways to write n as a sum of 1, 2, 3, and 4. For examples, if $n = 4$, the output should be 8, as

$$\begin{aligned}
4 &= 4,\\
4 &= 1+3,\\
4 &= 1+1+2,\\
4 &= 2+2,\\
4 &= 1+1+1+1,\\
4 &= 1+2+1,\\
4 &= 2+1+1,\\
4 &= 3+1.
\end{aligned}$$

Note that $3+1$ and $1+3$ are counted as different ways of writing 4.

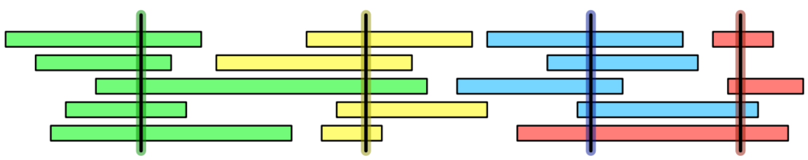
*Some of the problems are from Jeff Erickson's lecture notes. Looking into similar problems from his lecture notes on recursion and dynamic programming is recommended.

Problem 2. **[35 pts]** You are driving a bus along a highway, full of rowdy, hyper, thirsty students and a soda fountain machine. Each minute that a student is on your bus, that student drinks one ounce of soda. Your goal is to drop the students off quickly, so that the total amount of soda consumed by all students is as small as possible.

You know how many students will get off of the bus at each exit. Your bus begins somewhere along the highway (probably not at either end) and move s at a constant speed of 3.14 miles per hour. You must drive the bus along the highway; however, you may drive forward to one exit then backward to an exit in the opposite direction, switching as often as you like. (You can stop the bus, drop off students, and turn around instantaneously.)

Describe an efficient algorithm to drop the students off so that they drink as little soda as possible. Your input consists of the bus route (a list of the exits, together with the travel time between successive exits), the number of students you will drop off at each exit, and the current location of your bus (which you may assume is an exit). Prove your algorithm is correct and analyze its running time.

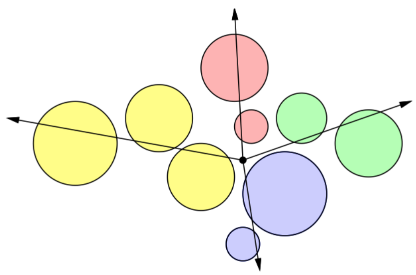
Problem 3. **[35 pts]** Let X be a set of n intervals on the real line. We say that a set P of points stabs X if every interval in X contains at least one point in P. Describe and analyze an efficient algorithm to compute the smallest set of points that stabs X. Assume that your input consists of two arrays $X_L = [1..n]$ and $X_R = [1..n]$, representing the left and right endpoints of the intervals in X. Prove that your algorithm is correct.



A set of intervals stabbed by four points (shown here as vertical segments)

Here is a set of practice problems on asymptotic running time analysis, and recursion. Do *not* submit solutions for the following problems, they are just for practice.

Practice Problem A. Given a set C of n circles in the plane, each specified by its radius and the (x, y) coordinates of its center, compute the minimum number of rays from the origin that intersect every circle in C. Your goal is to design and analyze an efficient algorithm for this problem. For full credit your algorithms must accomplish the task in polynomial time.



(a) **[20 pts]** Suppose it is possible to shoot a ray that does not intersect any balloons. Describe and analyze a greedy algorithm that solves the problem in this special case.

(b) **[10 pts]** Describe and analyze a greedy algorithm whose output is within 1 of optimal. That is, if m is the minimum number of rays required to hit every balloon, then your greedy algorithm must output either m or $m + 1$.

Practice Problem B. Suppose we need to distribute a message to all the nodes in a rooted tree. Initially, only the root node knows the message. In a single round, any node that knows the message can forward it to at most one of its children. Design and analyze an algorithm to compute the minimum number of rounds required for the message to be delivered to all nodes. For full credit your algorithms must accomplish the task in polynomial time.

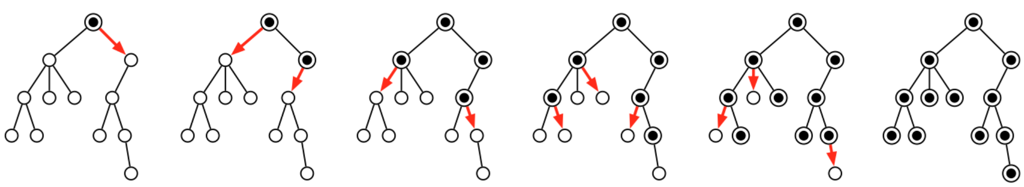


Figure 1: A message being distributed through a tree in five rounds.