Algorithms Midterm review
Chapters 1-4 problem session
In class, March 3, 2025

Problems

1. You've been hired to store a sequence of n books on shelves in a library. The order of the books is fixed by the cataloging system and cannot be changed; each shelf must store a contiguous interval of the given sequence of books. You are given two arrays $H[1..n]$ and $T[1...n]$, where $H[i]$ and $T[i]$ are respectively the height and thickness of the i^{th} book in the sequence. All shelves in this library have the same length L; the total thickness of all books on any single shelf cannot exceed L. You can adjust the height of each shelf to match the tallest book on that shelf. Describe and analyze an efficient algorithm to assign books to shelves to minimize the total sum of heights of the shelves.

2. In class, we discussed an algorithm for scheduling the maximum number of classes possible. At a high level, this algorithm greedily selected the class which ended first, eliminated any class that conflicted, and then recursed. This was not the only greedy strategy we could use, however! For each of the following alternative greedy strategies, either prove that the resulting algorithm is always optimal, or describe a (small) input example for which the algorithm does not produce an optimal schedule.

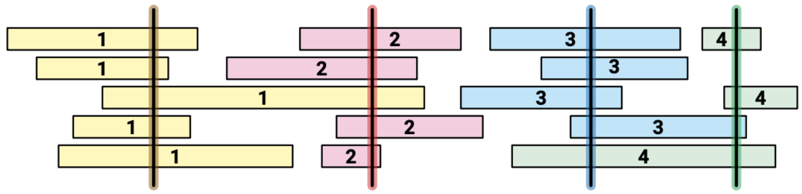
   (Hint: exactly one of these actually works, and the other two do not.)

   (a) Choose the course x that *ends last*, discard classes that conflict with x, and recurse.

   (b) Choose the course x that *starts last*, discard all classes that conflict with x, and recurse.

   (c) If no classes conflict, choose them all. Otherwise, choose the course with *longest* duration, remove it, and recurse.

3. Let X be a set of n intervals on the real line. We say that a set P of points *stabs* X (or is the stabbing set) if every interval in X contains at least one point in P. Our goal is develop an efficient algorithm to compute minimum set of points that stabs X. Assume that your input consists two arrays $L[1..n]$ and $R[1..n]$, where the i^{th} interval in X has left endpoint $L[i]$ and right endpoint $R[i]$.



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

(a) Given a set of intervals and any point x, define depth(x) as the number of intervals stabbed by x (or number of intervals that contains the point x). Consider the following max-depth greedy strategy: sort the right end points of the intervals based on their depth. Add the right end point with maximum depth to P. Remove from all the intervals stabbed by this point. Recompute the depths of the remaining intervals. Repeat until no intervals remain. Give a small counterexample to show that this greedy strategy is *not* optimal.

(b) Give an efficient algorithm to compute a smallest stabbing set of minimum size.

4. The robber from our last homework is back! The thief has found himself a new place for his thievery however, and is again trying to maximize the amount he can steal. However, this time there is only one entrance to this area, called root. Besides the root, each house has one and only one parent house. After a tour, the smart thief realized that all houses in this place form a binary tree. (Clearly, our thief has recently taken data structures.) The thief also quickly realizes after inspecting the security system that it will automatically contact the police if any two directly-linked houses are broken into on the same night.

Given as input a tree of houses, return the maximum amount of money the thief can rob in one night without alerting the police.

5. Your grandmother dies and leaves you her treasured collection of n radioactive Beanie Babies. Her will reveals that one of the Beanie Babies is a rare specimen worth 374 million dollars, but all the others are worthless. The valuable Beanie Baby is either slightly more or slightly less radioactive than the others, but you don't know which. Otherwise, as far as you can tell, they are all identical.

   You have access to a state-of-the-art Radiation Comparator at your job. The Comparator has two chambers. You can place any two disjoint sets of Beanie Babies in Comparator's two chambers; the Detector will then indicate which of the two subsets emits more radiation, or that the two subsets are equally radioactive. (The two subsets are equally radioactive if and only if they contain the same number of Beanie Babies, and they are all worthless.) The Comparator is slow and consumes a lot of power, and you really aren't supposed to use it for personal projects, so you really want to use it as few times as possible.

   Describe an efficient algorithm to identify the valuable Beanie Baby. How many times does your algorithm use the Comparator in the worst case, as a function of n?

6. Recall that a palindrome is any string that is equal to its reversal, like REDIVIDER or POOP. Describe an algorithm to find the length of the longest subsequence of a given string that is a palindrome.v