CS3100: Algorithms
Chapters 1-4 problem session
In class, October 4, 2019

Problems

1. Professor Chambers is driving her car from St. Louis to Denver along I-70 to get to an important conference. Her car's gas tank, when full, holds enough gas to travel 323 miles, and she has a map that gives distances between all the gas stations on her route; since she's essentially driving in a straight line, we can model this as an array $A[1..n]$, where $A[i]$ holds the number of miles from the $(i-1)^{st}$ gas stop to the i^{th} stop. She obviously wishes to make as few stops as possible along the way. Give an efficient algorithm by which Professor Chambers can determine at which gas stations she should stop.

2. Design and analyze an efficient algorithm to solve the following problem: Given a set of n integers, does the set contain a pair of elements a, b such that $a + b = 0$?

   (Hint: You can do better than $O(n^2)$.)

3. Suppose you are managing the construction of billboards on the Information Highway, a heavily traveled road that runs west-east for M miles. The possible sites for billboards are given by numbers $x_1, x_2, \ldots, x_n$, each in the interval $[0, M]$ (specifying their position along the highway, measured in miles from its western end). If you place a billboard at location x_i, you receive a revenue of $r_i > 0$.

   Regulations imposed by the county Highway Department require that no two of the billboards be within less that or equal to 5 miles of each other. You'd like to place billboards at a subset of the sites so as to maximize your total revenue, subject to this restriction.

   Example: Suppose $M = 20, n = 4$, $\{x_1, x_2, x_3, x_4\} = \{6, 7, 12, 14\}$, and $\{r_1, r_2, r_3, r_4\} = \{5, 6, 5, 1\}$. Then the optimal solution would place billboards at x_1 and x_3 for a total revenue of 10.

   Give an algorithm that takes an instance of this problem as input and returns the maximum total revenue that can be obtained from any valid subset of sites. The running time should be polynomial in terms of n, and as small as possible.

4. Consider a weighted version of the class scheduling problem from class, where different classes offer different number of credit hours (totally unrelated to the duration of the class, since we love those 0 or 1 credit hour labs that last for 4 hours). Your goal is now to choose a set of non-conflicting classes that give you the largest possible number of credit hours, given an array of start times, end times, and credit hours as input.

   (a) Prove that the greedy algorithm from lecture - choosing the class that ends first and recursing - does NOT always return an optimal schedule.

   (b) Describe an algorithm to compute the optimal schedule.