AD – Assignment 3

Greedy Algorithms

Søren Dahlgaard

Hand-in date: March 7th

Should be done in groups of 2–3 students

1 Motivation

The goal of this assignment is to familiarize the student with the proof methods of greedy algorithms: the greedy-choice property and optimal substructure. Furthermore you should see how we can attack some optimization problems by turning them into easier decision problems, which can be used to solve the original problem.

2 Problem statement

It is summer in Sunny Beach. A famous Danish chain of bars (not to be named) owns n bars along the shore, which are connected by the shore highway. Each bar has a limited supply b_i of beers. Since no one knows which bar the customers will prefer, we would like each bar to have the same amount of beers. In order to achieve this, that company has hired two Danish students to haul beers between the bars in their truck. However, being Danish, the students each drink a beer per kilometer they drive[1].

Your task will be to help the chain calculate the largest amount of beers $\hat{b}$ they can have at *all* bars. Specifically, you must describe an algorithm with the following input and output specifications:

Input: The position p_i and beer supply b_i of each bar. Here b_i is the supply for the bar at position p_i. You may assume that the positions are in sorted order – i.e. $p_1 < p_2 < \ldots < p_n$.

Output: The largest amount $\hat{b}$, such that each bar can have a beer supply of at least $\hat{b}$ after the two students have transferred beer between the bars.

3 Example

Consider the example in Figure 1. Here we have $b = (20, 40, 80, 10, 20)$ and $p = (0, 5, 13, 33, 36)$ (in kilometers). In order to send one beer from bar 3 to bar 4 we need to put 41 beers in the truck, as the students will drink 40 before reaching their destination (to send two beers we need to put 42 in the truck). The optimal $\hat{b}$ for the example is 21 and can be achieved as follows:

[1] The author of this assignment does not condone drunk driving in any way!

1. Bar 2 sends 11 beers towards bar 1. One beer arrives while ten are “lost” on the way.
2. Bar 3 sends 59 beers towards bar 4. 19 arrive while 40 are lost on the way.
3. Bar 4 now has 29 beers and send eight towards bar 5. Two of these arrive and six are lost on the way.
4. The final distribution of beer is: $(21, 29, 21, 21, 22)$.

Note that we can employ as many trucks as possible starting at arbitrary places. You can check for yourself that it is not possible to get 22 beers at each bar.

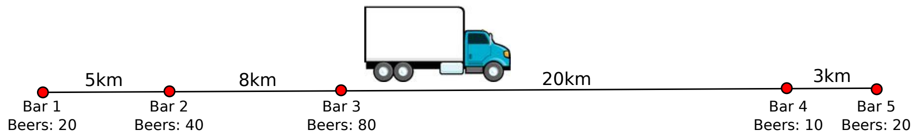

Figure 1: An example of a bar layout and b_i values.

4 Assignment

Task 1: Consider first the related decision problem: Given an integer $\bar{b}$. Determine whether it is possible to get at least $\bar{b}$ beers in every bar.
Give a greedy algorithm that solves this problem in $O(n)$ time, where n is the number of bars.
In order for the greedy choice property and optimal substructure to make sense for a decision problem, you can define an optimal solution to be a solution with at least $\bar{b}$ beers in every bar if such a solution exists; otherwise, any solution is an optimal solution.

Hint: Consider the bars in increasing order (left to right in Figure 1); for bar $i < n$, either it needs beers from bar $i + 1$ or it may have a surplus of beers for bar $i + 1$.

Task 2: Give a formal proof of correctness of your algorithm from task 1. If you are unsure how to do this, you can follow the structure from the book by showing that the problem exhibits the greedy-choice and optimal substructure properties.

Task 3: Using the procedure from task 1 give an $O(n \log B)$ algorithm to find the maximum value $\hat{b}$. Here B is the maximum amount of beers in any bar (i.e. $\max(b_1, \ldots, b_n)$). You must argue for the correctness and running time of your algorithm.