Exercises - Greedy - Algoritmiek

Tutorial 22 February 2022

1. **Pancakes (part 1):** You have bought a Pancakebot, a 'printer' for pancakes. Your idea is to use it to put on children's parties. There are n potential children's parties you can bring your Pancakebot to, where party i has a fixed start time $s(i)$ and end time $e(i)$. Because you have only one Pancakebot, you have to select a subset of the parties to go to. Each party has the same profit, 100 euros. Which parties do you select to earn as much money as possible?
   We consider several strategies by which to select the parties. Each strategy proposes a rule to select a party; after a party is selected, we remove all parties that overlap in time with the selected party (i.e. that are not compatible) and re-apply the rule until no longer possible. This always yields a feasible solution.
   (a) Consider a strategy where you always select a party that starts earliest, that is, with the minimal start time $s(i)$. Give an example that shows that following this strategy might yield a sub-optimal solution.
   (b) Consider instead a strategy where you always select a party of shortest duration, that is, with minimal $e(i) - s(i)$. Give an example that shows that following this strategy might yield a sub-optimal solution.
   (c) Consider now a strategy where you always select a party that ends earliest, that is, with the earliest ending time $e(i)$. Use an exchange argument to argue that this strategy yields an optimal solution.
   Hint: consider an optimal solution that has the most parties in common with the greedy solution.
   (d) As an alternative argument, use induction to prove that greedy always "stays ahead". That is, $e(G_i) \leq e(O_i)$ for all $1 \leq i \leq k$, where $G_1, \ldots, G_k$ are the greedily selected parties and $O_1, \ldots, O_\ell$ is an optimal solution. Then show that $k = \ell$.

2. **Pancakes (part 2):** Your idea to make pancakes using the Pancakebot is a great success! You decide to change your earnings model. For exactly 1 hour at party i, you ask an amount of k_i euros. Party i still has starting time s_i, which is guaranteed to be an integer. The duration is 1 hour, so the ending time of party i is $s_i + 1$. Which parties do you go to to make as much money as possible.
   (a) Give an optimal greedy strategy and prove the greedy choice property using an exchange argument (uitwisselargument).
   (b) Suppose there are n parties and $1 \leq s_i \leq n$. Give an algorithm to compute an optimal solution with running time $O(n)$.

3. **Lampposts:** A long road has n houses, where house i has distance s_i from the start (you may assume the houses are sorted by distance, $i < j \Rightarrow s_i \leq s_j$). The city has decided to place lampposts along the road such that each house has a lamppost within distance M.
   Example: If s is: 340, 670, 1200, 1600, 2400, 2710 and $M = 500$, then three lampposts suffice, for example on positions 500, 1500, 2400.
   Give a greedy algorithm that computes in linear time the minimum number of lampposts needed. Prove the greedy choice property.

4. **Bicycle rental:** A rental shop has n bicycles. The height of bicycle j is f_j. De rental shop can rent all bikes to a group of n people. The height of person i is p_i. Because everyone likes to have a fitting bike, we calculate a *mismatch* of $|f_j - p_i|$ if person i rides bike j. Give a greedy algorithm that gives a bike to each person, while minimizing the sum of mismatches. Prove the greedy choice property! The running time of your algorithm should be $O(n \log n)$.

5. **Fibonacci-Huffman:** Give an optimal Huffman-code for $\{a, b, c, d, e, f, g, h\}$, where the relative frequencies are as the Fibonacci-numbers:

σ	a	b	c	d	e	f	g	h
$f(\sigma)$	1	1	2	3	5	8	13	21