Eidgenössische Technische Hochschule Zürich

Ecole polytechnique fédérale de Zurich
Politecnico federale di Zurigo
Federal Institute of Technology at Zurich

Department Informatik
Markus Püschel
Peter Widmayer
Thomas Tschager
Tobias Pröger

10th November 2016

Datenstrukturen & Algorithmen — Exercise Sheet 8 — AS 16

Hand-in: Thursday, 17th November 2016 before the start of the lecture at 10:00 in the entrance area of ML D28. Please staple all sheets together and use this sheet as the cover page. Fill out the first two fields of the form below.

Exercise class (Room & Day): ______________________

Submitted by: ______________________

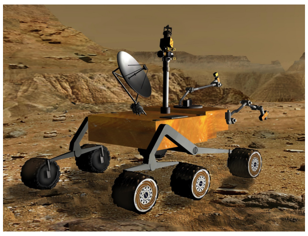
Corrected by: ______________________

Bonus points: _____

Exercise 8.1 *Mars mission.*

The rover *Curiosity* landed on Mars and is located at a starting position S. The goal is to move to a target position Z, and to collect rock samples that are as valuable as possible. To not use too much energy, the rover is only allowed to take a step to the east (right) and to the south (down). The rover can only collect a rock sample, if it is on the corresponding field. The value of each rock sample is stored in an $(m \times n)$ matrix, e.g.

S	9	2	5	11	8
17	21	32	5	15	3
2	2	3	8	1	5
8	2	8	11	15	9
0	5	3	10	4	**Z**



In the matrix above, an example of a south-east path from S to Z is shown where the value of the collected rock samples is maximum. This path can be described by enumerating the corresponding matrix positions: $(1,1) \to (2,1) \to (2,2) \to (2,3) \to (2,4) \to \cdots \to (5,6)$.

Provide a *dynamic programming* algorithm that takes an $(m \times n)$ matrix A with $A[1,1] = A[m,n] = 0$, and that computes a south-east path from $S = (1,1)$ to $Z = (m,n)$ where the value of the collected rock samples is maximal. Notice that we search for the path itself, and not just for the maximum value. Provide also the running time of your solution in dependency of m and n.

Please turn over.

Exercise 8.2 *Hike.*

Alice and Bob would like to do a hiking trip with a picnic and have selected n items that they want to take with them. The i-th item weighs $g_i \in \mathbb{N}$ gram. Both Alice and Bob have a backpack of exactly the same size, that could take all items at once. For the sake of fairness, they want to distribute the items between the backpacks, such that they are almost equally heavy. Every item has to be placed in either one or the other backpack. Especially, an item cannot be left at home or split into two parts. Specifically, we are looking for two sets of A and B of items with $A \cap B = \emptyset$, $A \cup B = \{1, \dots, n\}$, whose weight difference is as small as possible among all such two sets.

Provide a dynamic programming algorithm that computes such two sets A and B with a minimum weight difference among all such sets as described above. Provide also the running time of your solution. Is the running time polynomial?

Exercise 8.3 *Numerical Puzzle.*

You are given a sequence of n digits from the set $\{0, \dots, 9\}$ and a positive integer σ. If plus signs are inserted between some digits, and if the digits between two plus signs are interpreted as a decimal number, then the sum of the corresponding numbers can be computed. In general, different insertions of plus signs yield different sums.

Example: For the sequence [6 9 2 5 0 2 1 3] we can obtain the sums, for example, $69 + 2 + 5 + 0 + 21 + 3 = 100$ and $6 + 9 + 250 + 21 + 3 = 289$.

The exercise is to decide whether plus signs can be inserted into a given sequence such that the sum equals exactly σ.

a) Design an efficient algorithm that uses dynamic programming to solve this problem. You may assume that σ is relatively small in relation to n. Provide also the running time of your solution. Is it polynomial in the size of the input?

   Hint: Notice that you only need to *decide* whether σ can be achieved or not.

b) How can we efficiently find all arrangements of plus signs that yield the desired sum?