Exercise 12.1  

Algorithm design I (part of an exam in February 2011).

You are asked to plan a theater festival. Given a set $V = \{v_1, \ldots, v_n\}$ of possible performances, you need to select a subset of performances to be scheduled. There is exactly one stage available. Each performance consists of a starting time $s_i$ and end time $e_i$, where $0 \leq s_i < e_i < \infty$. If a performance $v_i$ is scheduled, it occupies the stage in the time interval $(s_i, e_i)$ (open on the right). There are no two performances in $V$ that share the same time interval. Two performances $v_i, v_j$ are said to be compatible, if their intervals $(s_i, e_i)$ and $(s_j, e_j)$ do not overlap, i.e., if $e_i \leq s_j$ or $e_j \leq s_i$. Your task is to create a schedule that contains the maximum number of mutually compatible performances.

Example: for a set $V = \{v_1, v_2, v_3\}$ with $s_1 = 10, e_1 = 13; s_2 = 11, e_2 = 14; s_3 = 13, e_3 = 16$, the schedule $\{v_1, v_3\}$ is a set with a maximum number of compatible performances.

a) Briefly describe your idea for an algorithm that, from a given set $V$ of possible performances, efficiently computes a set with a maximum number of mutually compatible performances.

b) Provide compact pseudocode for your algorithm in a).

c) Prove that your algorithm is correct, and provide its running time.

Exercise 12.2  

Algorithm design II (part of an exam in February 2011).

The marketing department of a bank wants to evaluate the data of its „customers recruit customers“ program. Every customer of the bank has either been recruited directly by the bank, or by an existing customer. The bank is interested in finding out whether a customer $a$ has been recruited directly, or by another customer $b$.

Formally, the following problem should be solved: you are given a set $K = \{1, \ldots, n\}$ of customer numbers, and a set $S$ of ordered pairs $(u, v)$, $u \in (K \cup \{0\}), v \in K, u \neq v$, meaning that $u$ (where 0 represents the bank) has recruited customer $v$. For each customer $v \in K$ there exists exactly one pair $(u, v) \in S$, since the recruiter is unique for each customer. We call $u$ a successor of $v$, if there is a $k$ and a sequence of customers $v = v_0, v_1, v_2, \ldots, v_k = u$ such that $(v_i, v_{i+1}) \in S$ for every $0 \leq i < k$. The operation SUCCESSOR?(a, b) returns TRUE if $a$ is a successor of $b$, otherwise it returns FALSE.

For each of the following cases, design a data structure that supports the required operations as efficiently as possible.

Hint: Of course, you can use algorithms known from the lecture without repeating their pseudocode.

a) In addition to the initialization of the data structure by given sets $K$ and $S$, first only the operation SUCCESSOR?(a, b) has to be supported. In particular, $K$ and $S$ are static, i.e., neither insertions nor deletions are allowed.

Describe your data structure in words. Give the running time that is necessary for the initialization of the data structure. Provide pseudocode for the operation SUCCESSOR?(a, b), and provide its running time.
b) Now assume that, in addition to a), you are required to also support the operation $\text{DELETE}(a)$ to remove a customer $a \in K$ if it has no successors, and otherwise generate an error message.

Provide pseudocode for the operation $\text{DELETE}(a)$, and specify its running time. Describe the changes from the data structure used in a) in words (or specify a new data structure). Describe also the changes from the running time in a).

c) Finally, also the operation $\text{RECRUIT}(a, b)$ has to be supported, that allows to insert a new customer $a \notin K$ recruited by $b \in K$.

Provide pseudocode for the operation $\text{RECRUIT}(a, b)$, and specify its running time. Describe the changes from the data structure used in b) (or specify a new data structure). Describe also the changes from the running times in a) and b).

$\textbf{Hint}$: If it helps, you may assume that only a constant amount of memory is required for working with arbitrary precision on rational numbers.