Exercise 8.1  Splay Trees & Optimal Search Trees.

Consider the following tree:

![Diagram of a tree](image)

a) Give a sequence of insertions (with key values) on an initially empty splay tree that leads to the above tree.

b) Give a sequence of 7 keys with corresponding access frequencies and access frequencies for each interval between two keys, such that the optimal search tree for this sequence has the above structure.

Exercise 8.2  Branch and Bound.

In this exercise, we wish to apply the branch and bound method to a classical optimization problem. Given an (undirected) graph $G = (V,E)$ with $n = |V|$ vertices and $m = |E|$ edges, the **Minimum Dominating Set Problem** asks to compute a minimum dominating set $D \subseteq V$ of vertices, i.e. a smallest set $D \subseteq V$ such that every vertex in $V \setminus D$ has a neighbor in $D$.

We consider the following example:

![Diagram of a graph](image)

In this graph, $D = \{a, b, c, h, i\}$ is a dominating set from which we cannot eliminate any vertex without losing dominance. However, $D$ is not as small as possible, so it is not a minimum dominating set. The exercise is to find a minimum dominating set for the above graph by applying the branch and bound method by hand. Proceed as follows:

a) First develop a lower bound on the number of required vertices for a given partial solution.

A partial solution is described by two sets $I$ and $O$. The set $I$ contains the vertices that are chosen to be part of the dominating set, while $O$ contains the vertices that are chosen **not** be part of it.
b) Provide a branching rule to specify which vertex is the next one to be considered.

c) Perform branch and bound using the answers you gave in a) and b). Draw a complete
decision tree and indicate the order of the decisions and the corresponding lower bounds.
Identify the final solution.

Exercise 8.3 Numerical Puzzle.

You are given a sequence of $n$ digits $0, \ldots, 9$, and a positive integer $\sigma$. Various sums can be obtained by inserting plus signs in different positions between the digits. If digits are not separated by plus signs, they are treated as a single decimal number.

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

a) Design an efficient algorithm for this problem using dynamic programming. You may assume that $\sigma$ is relatively small in relation to $n$.

*Hint:* Note that you only need to decide whether $\sigma$ can be achieved or not.

b) Provide the asymptotic running time of the algorithm in b). Is it polynomial in the size of the input?

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

**Hand-in:** until Wednesday, 24th April 2013.