CIS 435 Homework 4 (Due: November 26, 2003)

Solve ALL the problems. Collaboration is prohibited

8 points total

Problem 1. (Huffman codes)  (2 points)

(a) What is the minimum cost of a Huffman code for 13 items with weights:

\[2, 4, 3, 5, 7, 11, 1, 9, 3, 6, 2, 8\]

Draw a corresponding optimal Huffman tree.

(b) Write the closed formula for the minimum cost of a Huffman code for \(n\) items with weights:

\[1, 2, 4, 8, \ldots 2^{n-1}\].

Problem 2  (2 points)

A (potentially unbalanced) binary search tree \(T\) is a binary tree in which each left son contains a key which is smaller than its father and each right son contains a key which is greater than its father and there are no special requirements on the height and the structure of \(T\). We say that two permutations of numbers \(1, 2, \ldots, n\) are \(bst\)-equivalent if the same (not necessarily balanced) binary search tree results after inserting (using the method from the class) the numbers in orders corresponding to any of these permutations. Compute the number of all permutations which are \(bst\)-equivalent to the permutation

\[6 2 1 3 10 8 7 9\]

Problem 3. (Towers of Hanoi)  (2 points)

Write the configuration of the Tower of Hanoi problem for 5 elements after 20 moves. All elements 1, 2, 3, 4, 5 are to be moved from the tower (stack) 1 to the tower 3 using minimal number of moves. Additionally the tower 2 is used. It is not allowed to place larger element above a smaller one. One move consists in shifting top element from one tower to another tower, as its top element. The configuration specifies which elements are on tower 1, tower 2 and tower 3. Elements on one tower are to be written horizontally (in increasing order). For example the initial configuration is (\(< 12345 >\), \(< \emptyset >\), \(< \emptyset >\)), the final one is (\(< \emptyset >\), \(< \emptyset >\), \(< 12345 >\)). The empty set is denoted here by \(\emptyset\). The problem is also described in "Additional informal lecture notes", see the web page of the course.

Problem 4. (Red-black trees)  (2 points)

Draw red-black tree which results after inserting elements

\[12, 10, 9, 13, 11, 8, 7, 6, 5, 4, 3, 2, 1, 14\]

(in this order) into initially empty red-black tree using algorithm from the class. Indicate which vertices are black (as square nodes).