CS140: Algorithm
Assignment 6
Assigned 10/29/99, Due 11/5/99

1. In the skyline problem, one is given the positions and shapes of n rectangular buildings along a city skyline. The objective is to draw the (two-dimensional) skyline of the buildings, eliminating hidden lines. For example, consider the buildings:

   $(1, 11, 5), (2, 6, 7), (3, 13, 9), (12, 7, 16), (14, 3, 25), (19, 18, 22), (23, 13, 29),$ and $(24, 4, 28)$

   shown in Figure (a). Note a building is described as (left-side,height,right-side). The solution for this input, which is illustrated in Figure (b), can be represented as

   $$(1, 11, 3, 13, 9, 0, 12, 7, 16, 3, 19, 18, 22, 3, 23, 13, 29, 0).$$

   (a) Reduction 1:
      i. Show how to compute SKYLINE$(b_1, b_2, ..., b_n)$ given SKYLINE$(b_1, b_2, ..., b_{n-1})$ and b_n where $n > 1$.
      ii. Show how to solve the base case $n = 1$.
      iii. The previous steps define a recursive algorithm for computing the skyline of a set of buildings. Write a recurrence relation for the running time of this algorithm and solve it.

   (b) Reduction 2:
      i. Show how to compute SKYLINE$(b_1, b_2, \ldots, b_n)$ given SKYLINE$(b_1, b_2, \ldots, b_{n/2})$ and SKYLINE$(b_{n/2+1}, \ldots, b_n)$, where $n > 1$ is a power of 2.
      ii. Write a recurrence relation for the running time of your algorithm and solve it.

2. In the maximum consecutive subsequence (MCS) problem, one is given a sequence of integers $x_1, x_2, \ldots, x_n$ and is asked to find an i and j such that $\sum_{k=i}^{j} x_i$ is as large as possible.

   (a) Find a maximum consecutive subsequence for the input: $2, -1, 3, 4, -20, 16, 3, -2, 5$.

   (b) Repeat for the sequence 2, -5, 18, 2, -4, 2, -2, 3

   (c) Can you produce a solution to a sequence of n integers from the solution to the first $n-1$? Can you produce a solution to a sequence of n integers given the solution of to every subproblem of size at most $n-1$? (A subproblem is a sequence $x_i, x_{i+1}, \ldots, x_j$, where $j \geq i$.) Explain your answer.

(d) Define a new problem (Modified MCS) such that MCS reduces to MMCS and MMCS is (easily) self-reducible. Give the two reduction and analyze their running times.

3. Records consisting of a word and its definition are stored in a balanced binary search tree. The records are keyed on the words $w_1 < w_2 < \ldots < w_n$, where the order is alphabetical. Users query the data structure for word definitions. After some time you find that some words are queried more often than others. Specifically, you develop statistics on usage and compute the probability p_i that word w_i is queried. You realize that you may be able to reduce average search time if you build a tree that, rather than being balanced, minimizes the cost function $\sum_{i=1}^{n} c_i p_i$, where c_i is the number of queries needed to reach the node containing w_i.

   (a) Give an example of a set of words and probabilities such that cost $\sum_{i=1}^{n} c_i p_i$ of a balanced binary search tree for the data is not minimal. Give an unbalanced search tree that has smallest possible cost.

   (b) Can you reduce the problem of constructing an optimal binary search tree on n items to the problem of computing such a tree for the first $n-1$ items? Explain the difficulty.

   (c) Devise a recursive algorithm for this problem. Specifically, let OPT-BST$(w_1, p_1, \ldots, w_n, p_n)$ denote the minimal cost of any binary search tree for the n words. Give an inductive definition of OPT-BST on n words in terms of OPT-BST on subproblems of size at most $n-1$. (A subproblem of size k is $w_i, p_i, w_{i+1}, p_{i+1}, \ldots, w_{i+k-1}, p_{i+k-1}$.) The number of subproblems you need in the definition is less than n. What are the appropriate base cases for your recursive algorithm? What is the running time of your recursive algorithm?

   (d) Give a dynamic programming algorithm for the problem. What is the running time of your algorithm?

4. Problem 17-1 on page 353 of the text.