Due: Thu Apr 16th at the start of class

Homework #4

CMSC351 - Spring 2013

PRINT Name: ________________________________:

- Grades depend on neatness and clarity.
- Write your answers with enough detail about your approach and concepts used, so that the grader will be able to understand it easily. You should ALWAYS prove the correctness of your algorithms either directly or by referring to a proof in the book.
- Write your answers in the spaces provided. If needed, attach other pages.
- The grades would be out of 16. Four problems would be selected and everyone’s grade would be based only on those problems. You will also get 4 bonus points for trying to solve all problems.

1. Given two sets $S_1$ and $S_2$, and a real number $x$, find whether there exists an element from $S_1$ and an element from $S_2$ whose sum is exactly $x$. The algorithm should run in average time $O(n)$, where $n$ is the total number of elements in both sets. (Try to use the data structures you have learned in the lectures.)
2. [Prob 4.22, Pg 88] Determine the general structure of a binary search tree formed by inserting the numbers 1 to \( \pi \) in order (you may draw a general structure, but explain it properly). What is the height of this tree?
3. [Prob 4.16, Pg 88] Design a data structure that supports the following operations. Each operation should take $O(\lg n)$ time, where $n$ is the number of elements in the data structure. Explain the required process for doing each operation and prove that it takes $O(\lg n)$ time.
   a) $\text{Insert}(x)$: Insert the key $x$ into data structure only if it is not already there.
   b) $\text{Delete}(x)$: Delete the key $x$, if it is there!
   c) $\text{Find Smallest}(k)$: Find the $k$th smallest key in the data structure.
4. For each $1 \leq i \leq n$ job $i$ is given by two numbers $d_i$ and $p_i$, where $d_i$ is the deadline and $p_i$ is the penalty. The length of each job is equal to 1. We want to schedule all jobs. If we schedule job $i$ after its deadline $d_i$, we should pay its penalty $p_i$. Design a greedy algorithm to find a schedule which minimized the sum of penalties.
5. Suppose we want to make change for \( n \) cents and the only denominations allowed are 1, 10 and 25 cents.
   
   a. Find an example such that the greedy algorithm does not find the minimum number of coins required to make change for \( n \) cents (give a concrete counterexample).
   
   b. Give a \( O(n) \) dynamic programming algorithm to find the minimum number of coins required to make change for \( n \) cents.
6. **Weighted Interval Scheduling:** Consider a set of \( n \) intervals where each interval is given by \((s_i, t_i)\). Where \( s_i \) is the start time and \( t_i \) is the finish time. In addition each interval also has a weight given by \( w_i \). Give a dynamic programming algorithm to find the maximum weight of a non-conflicting set of intervals.
7. A professor needs to choose a sequence of locations to conduct a class, one for each day. The days are numbered 1, 2, ..., n. The two possible locations are AVW (A.V. Williams) and CSIC. For each 1 ≤ i ≤ n, the cost of conducting the class in AVW on day i is A_i and the cost of conducting the class in CSIC on day i is C_i. The cost of moving from AVW to CSIC (and vice versa) is some constant M. Give an O(n) algorithm which computes the cost of an optimal schedule of the class.