CS60003 Algorithm Design and Analysis, Autumn 2009–10

Mid-Semester Test

Maximum marks: 45 | September 20, 2009 | Total time: 2 hours

Roll no: ____________ **Name:** ______________________________

[*Write your answers in the question paper itself. Be brief and precise. Answer all questions.*]

1. Consider the following function that accepts a positive integer n as input.

```
playWith (n)
{
    while (n > 1) {
        Increment n by 1.
        while (n is even), set n = n/2.
    }
}
```

(a) Prove that the function terminates for every input integer $n \geqslant 1$. **(5)**

(b) Determine a tight bound on the running time of the above function. You should supply an argument to corroborate that your bound is tight (that is, achievable). **(5)**

2. (a) Let S and T be strings each of length n. Your task is to determine whether T can be obtained by cyclically rotating S. For example, the string star can be obtained by cyclically rotating the string tars, whereas the string arts cannot be obtained by cyclically rotating the string tars. Supply an $\mathrm{O}(n)$-time algorithm to solve this problem. **(5)**

(b) Let S and T be strings of lengths m and n respectively. Your task is to determine whether T is a sub-sequence of S, that is, whether the symbols of T occur in S in the same order as they appear in T, but not necessarily contiguously. For example, the string grim is a sub-sequence of the string algorithm, whereas the string gram is not. Supply an $\mathrm{O}(m + n)$-time algorithm to solve this problem. **(5)**

3. Consider the line-sweep algorithm for computing line-segment intersections. Suppose that some of the input segments are allowed to be vertical (that is, parallel to the y-axis). You may, however, assume that no two given vertical segments are collinear.

(a) Define a new type of event "Vertical Segment" to deal with this situation. Describe how you can efficiently handle this event. You are required to maintain the running time of the original algorithm. **(5)**

(b) Prove that you achieve the original running time of $O((n+h)\log n)$ in presence of "Vertical Segment" events. You may assume that a height-balanced binary search tree on k nodes can be so implemented that the predecessor or successor of a node in the tree can be located in $O(\log k)$ time. **(5)**

4. You are given a set of n real numbers $x_1, x_2, \ldots, x_n$. Your task is to cover these points by intervals of unit length (that is, by intervals of the form $[a, a+1] = \{x \in \mathbb{R} \mid a \leqslant x \leqslant a+1\}$ for real numbers a). Your goal is to minimize the number of intervals in the cover. (Here is a practical application of this problem. Suppose that the points x_i represent houses on a straight road. You want to cover all the houses by a set of cellular-phone towers each with a maximum range of $1/2$ km in each direction. Naturally, you attempt to serve all the houses with as few towers as possible.)

(a) Consider the following greedy strategy. Choose an interval of unit length to cover the maximum number of points in the given collection. Output this interval, and remove the points covered by this interval from the collection. (Serve the maximum possible number of houses by a single tower.) Repeat until no points are left. Give an example to demonstrate that this greedy algorithm may fail to provide an optimal solution. **(5)**

(b) Although the greedy strategy of Part (a) fails, there exist other greedy strategies that efficiently compute optimal solutions. Describe such a strategy. The running time of your greedy algorithm must be bounded by a polynomial in n. **(5)**

(c) Prove that your greedy algorithm correctly computes an optimal solution. **(5)**

ROUGH WORK