Theory Breadth Exam

September 14, 2005

Write each problem solution on a separate piece of paper. (You may use more than one page per problem if necessary.) Make sure your code—and not your name—is on each piece of paper you submit, as well as the problem name.

Formal Languages:

A language L is definite if there is an integer k and two sets A and B of strings such that all strings in A have length less than k, all strings in B have length k and

$$L = A \cup \{wb\colon w \in \Sigma^* \text{ and } b \in B\}.$$

In such case, we say that L is generated by (k, A, B).

1. Show that every definite language is accepted by a finite automaton.
2. Show that the class of definite languages is closed under complementation.
3. Give an example of definite languages L_1 and L_2 such that $L_1 L_2$ is not definite.

Theory of Computing:

1. Give an example of an undecidable language, with clear definition. Is it recognizable? (Vocabulary: decidable is the same as computable is the same as Turing computable; recognizable is the same as recursively/Turing enumerable.)
2. How do you prove that a language is not decidable? How do you prove that a language is not recognizable? (There are several correct answers; you will get credit for at most one for each question.)

3. Let $L = \{\#M : \; M(17) \uparrow\}$, where $\#M$ is the number that codes the Turing machine program for M, and $M(x) \uparrow$ means that the computation $M(x)$ does not halt. Say whether L is decidable, recognizable, or neither. Sketch a proof of your answer.

Data Structures:

Consider a balanced binary search tree (BST) T for a finite set of distinct real numbers. Besides the usual BST fields: key[x], left[x] and right[x] in each node x, there is an additional field rsize[x]. This field is an integer equal to the number of nodes in the right subtree rooted at x (not including x). We will refer to T as an *augmented* BST.

1. Draw a picture of such a BST for the set { 6, 12, 4, 8, 14, 2, 10}. For each node x, as a part of your picture show the values of all the fields for x.
2. Design a function OS-rank(tree T, int i, int size) that given an augmented BST T (or a pointer to such a tree) returns the key whose ordinal number (rank) in the set represented by T is i (size is the number of elements in T).

   Provide a narrative description, pseudocode, justification of the correctness and the running time of your function. Note: Solutions with the running time $\Omega(n)$, n the size of T, will receive only partial credit.

Algorithms:

Prove that every comparison based sorting algorithm has time complexity in $\Omega(n \log(n))$, where n is the number of elements to be sorted.

Logic/Discrete Math:

Recall that an *exponential generating function* for a sequence $\mathbf{a} = \langle a_n \rangle_{n=0}^{\infty}$ is the function

$$f_{\mathbf{a}}(x) = \sum_{n=0}^{\infty} \frac{a_n}{n!} x^n.$$

For instance, if the sequence $\mathbf{a}$ is a constant sequence such that $a_n = 2$, then its closed form is $f_{\mathbf{a}} = 2 \cdot e^x$.

Find the closed form (i.e. no infinite sums in your solution) of the exponential generating function $f_{\mathbf{p}}$ for the sequence $\mathbf{p} = \langle p_n \rangle_{n=0}^{\infty}$ where p_n is the number of permutations of an n-element set if n is of the form $n = 3m + 1$, and $p_n = 0$ otherwise.

Choose one of the following problems:

Graph Algorithms:

Describe Dijkstra's algorithm for solving the single source shortest path problem:

Instance: An undirected graph $G = (V, E)$ with nonnegative integer edge weights $w(a, b)$ for $a, b \in V$, plus a vertex $u \in V$.

Problem: For every vertex $z \in V$, find the distance from u to z.

Give pseudocode, explain the algorithm, analyze its time complexity, and identify any data structures needed to achieve the best complexity.

Computational Complexity:

We define the class NL (nondeterministic logspace) to be NSPACE($\log n$). It is not known whether NL=P or even NL=NP. Suppose that NL=NP. From that, prove that NP is *properly* contained in exponential time ($\cup_k$DTIME(2^{kn}))