Examination "Theory Breadth" February 8, 2006

SOLVE EACH PROBLEM ON A SEPARATE PAGE. WRITE ONLY ON ONE SIDE OF EACH PAGE. WRITE YOUR CODE ON EACH PAGE. DO NOT WRITE YOUR NAME ON ANY PAGE.

1. For each of the following sets, determine whether it is

   R: Regular

   C: Context-free and not regular

   N: Not context-free

   For each language, prove your answer.

   (a) $T = \{x \in \{a,b\}^* |$ every block of as in x has an even number of as or every block of bs in x has an odd number of bs$\}$.

   (b) $U = \{0^i 1^j 2^i | \; i, j \in \mathbb{N}$ and $i \geq 1\}$.

2. For each of problems A, B, and C below, is the problem (1) decidable (= recursive), (2) Turing enumerable (= recursively enumerable) but not decidable, or (3) not Turing enumerable?

   A: Does a given deterministic finite automaton accept infinitely many strings?

   B: Does a given pushdown automaton accept infinitely many strings?

   C: Does a given Turing machine accept infinitely many strings?

   Sketch proofs of your answers.

3. (a) Define the Binary Heap representation for priority queues. Describe both tree and array representations. List the basic priority queue operations on Binary Heaps and their running times.

   (b) Show the binary heap (maximum element at the root) that is constructed with the Bottom-Up (linear time) build-heap procedure for the following set of keys: $S = \{F, E, B, R, U, A, R, Y\}$. Assume alphabetical order among the keys. That is, A is less than B, etc. Show both tree and array representation for your heap.

   (c) Consider a binary heap and its tree representation and assume that all the key values are distinct. To recall, the node level is defined as follows: the root is at the level 1, the children of the root are at the level 2, the children of the children of the root are at the level 3, etc.

   Prove (a precise argument is needed) that the $k-th$ largest element in the set of keys is located in one of the levels 1 through k. For example, the third largest element is not farther from the root than at the level 3.

4. Describe an algorithm for finding a minimum cost spanning tree in a connected edge weighted graph. Give pseudo-code, identify the critical data structures you are using, explain why your algorithm is correct, and analyze the complexity, including an explanation of how the choice of data structures affects the complexity.

5. Let n be a positive integer. Let p_1, p_2, and p_3 be distinct positive prime numbers. Give a closed form expression for the number of positive integers that are less than or equal to n, and neither divisible by p_1, nor divisible by p_2, nor divisible by p_3. That is, an expression for

$$|\{x : 1 \leq x \leq n \text{ and } p_1 \not| x \text{ and } p_2 \not| x \text{ and } p_3 \not| x\}|.$$

6. **Graph Algorithms:** Construct an algorithm to find a pair of most distant vertices in an undirected tree. Analyze the running time of the algorithm. There will be no credit for algorithms that do not run in polynomial time in the number of vertices in the tree. There will be a deduction of 20% of the score for algorithms that do not run in linear time.