CSE 431 Spring 2009
Assignment #1

Due: Friday, April 10, 2009

Reminder Sign up for the *cse431@cs* mailing list.

Reading assignment: Read Chapter 3 of Sipser's text.

Problems:

1. Give a Turing machine diagram for a Turing machine that on input a string $x \in \{0,1\}^*$ halts (accepts) with its head on the left end of the tape containing the string $x' \in \{0,1\}^*$ at the left end (and blank otherwise) where x' is the successor string of x in lexicographic order; i.e. the next string in the sequence $\epsilon, 0, 1, 00, 01, 10, 11, 000, \ldots$ in which the strings are listed in order of increasing length with ties broken by their corresponding integer value. (Briefly document your TM.)

2. Give implementation-level descriptions (not full formal descriptions) of Turing machines that decide the following languages over the alphabet $\{0,1\}$:

   (a) $\{0^a 1^b 0^c \mid b > 0 \text{ and } c = a \bmod b\}$.

   (b) $\{0^p \mid p \geq 2 \text{ is prime}\}$ using a Turing machine with 2 tapes. You may refer to the Turing machine in part (a) or other Turing machines for which implementation level descriptions have been given in the text.

3. Sipser's text (1st or 2nd edition) Problem 3.7.

4. Let a k-PDA be a pushdown automaton that has k stacks. Thus a 0-PDA is an NFA and a 1-PDA is a conventional PDA. Show that 2-PDAs are as powerful as Turing machines by simulating a Turing machine tape with two stacks.

5. A Turing machine with a 2-dimensional tape is like a 1-tape TM except that it marked with an infinite 2-dimensional grid of cells that are all blank, except for the input which is given in the cells under the read head and in the sequence of cells immediately to its right. Additional changes are that

   - the transition function δ, is $\delta : Q \times \Gamma \rightarrow Q \times \Gamma \times \{L, R, U, D\}$ where U and D indicate moves *up* and *down* one cell.
   - there is no end of the tape.

   Give an implementation level description of how an ordinary 1-dimensional Turing machine can simulate a 2-dimensional one; that is, the 1-dimensional TM should accept, reject, or run forever on exactly the same set of inputs as the 2-dimensional one does.