Problem Set 9

Submission instructions: Submit each problem on a **separate sheet of paper**, put your name on each sheet, and write your discussion section time and day (e.g. Tuesday 10am) in the upper righthand corner. These details may sound picky, but they make the huge pile of homeworks much easier to grade quickly and more importantly, since we return them in the discussion sections, easier for you to get them back.

Also, write on each exercise the name/netid of your group members.

Due: Friday, April 17, 2009 at 12:30 in Elaine Wilson office (SC 3229). If the door is locked, slide your solutions under the door.
Version: **1.01**

(Q1) High Level TM Design.
[**Category**: Construction of machines, **Points**: 10]

A perfect number is a positive integer that is the sum of its proper positive divisors, that is, the sum of the positive divisors excluding the number itself. For example, 6 is the first perfect number, because $1, 2, 3$ are its proper positive divisors, and $6 = 1 + 2 + 3$.

Design a Turing Machine that for a given input tape with n cells containing `0`'s, marks the positions which are perfect numbers.

Specifically, cells at the non-perfect-numbered positions are left containing `0`, cells at perfect-numbered locations are left containing `X`. For example, consider the input `0000000000`. This represents the first 10 numbers. So the Turing machine should halt with `00000X0000` on the tape.

(Q2) TM Encoding.
[**Category**: Comprehension, **Points**: 10]

In this problem we demonstrate a possible encoding of a TM using the alphabet $\{\mathtt{0}, \mathtt{1}, ;\}$ where ; is used as a seperator. We encode $M = (Q, \Sigma, \Gamma, \delta, q_0, q_{\mathrm{acc}}, q_{\mathrm{rej}})$ as a string representing $|Q|$, $|\Sigma|$, $|\Gamma|$, q_0, q_{acc}, q_{rej} and then the δ. Each of the first five quantities is represented using a numbering system where n is represented by a 1 followed by n $0's$.

Thus is $|Q| = n$, this means we have n states numbered 1 to n.

If $|\Sigma| = n$, this means we have n symbols in the alphabet. We adopt the convention that these symbols are 0 to $n - 1$ where each can be represented using the numbering system mentioned above.

We use a similar scheme for Γ with the restriction that the blank symbol is assigned the largest number.

q_0, q_{acc} and q_{rej} are the next 3 numbers represented.

D is either 10 to mean move left or 100 to mean move right.

The remaining string represents δ as follows. Each transition $(q, \mathtt{a}) \rightarrow (q', \mathtt{b}, D)$ is represented as the concatenation of the represntation of each of the 5 quantities and the reresentation of δ is simply the concatenation of the representation of each transitions. We use the convention that transitions not mentioned go to the reject state in the encoded machine.

Here is the representation of a mystery Turing machine M, using this encoding.

$$
\begin{array}{l}
1000000; \\
1000; \\
10000; \\
10; \\
100000; \\
1000000; \\
10; 100; 100; 100; 100; \\
100; 1; 100; 1; 100; \\
100; 10; 100; 10; 100; \\
100; 1000; 1000; 1000; 10; \\
1000; 1; 100000; 10; 100; \\
1000; 10; 10000; 1; 10; \\
10000; 10; 10000; 1; 10; \\
10000; 100; 100000; 10; 10; \\
10000; 1; 100000; 10; 10
\end{array}
$$

(a) Draw a state diagram for this TM (omitting the reject state).

(b) What is the language of this TM? Give a brief justification.

(Q3) TM with infinite number of tapes.
[**Category**: Construction, **Points**: 20]

An ITM is a special TM with one head and infinite number of tapes (one sided tapes). The tapes are numbered $0, 1, 2, 3, \cdots$. The cells on each tape are also numbered from left to right with $0, 1, 2, 3, \cdots$. When the machine starts, the head is on cell number 0 of tape number 0. If the head is on cell number i of tape number j, it can overwrite that cell and move either to cell $i+1$ or $i-1$ (if $i \geq 1$) of tape j or move to cell i of tape $j+1$ or tape $j-1$ (if $j \geq 1$).

Prove that ITM's and normal TM's are equivalent.
(you should prove that every normal TM can be simulated using an ITM -this should be the easy part- and every ITM can be simulated using a normal TM).

SEE NEXT PAGE FOR ANOTHER QUESTION!

(Q4) What does it do?
[**Category**: Comprehension, **Points**: 10]

What does the following TM do?