COM 2030 Exercise sheet 3:
Turing Machines

1. Using any of the building block Turing machines

   (a) L,R, and x;

   (b) the searching machines R_x, $R_{\neg x}$, L_x, and $L_{\neg x}$; or

   (c) the shifting machines S_R and S_L

   construct a composite Turing machine that converts a binary representation of a positive integer to a unary representation of that number (i.e. converts a binary representation of a positive integer n to a sequence of n 1's). Assume the machine starts in the configuration $\underline{\Delta} n \Delta\Delta \cdots$ where n is the binary representation of the number to be converted. It should halt in the configuration $\underline{\Delta} 1 \cdots 1 \Delta\Delta \cdots$ where n 1's follow the initial blank. In both the starting and ending configuration the blank on which the tape head is positioned is the leftmost tape cell on the tape.

   Explain the strategy your machine uses.

2. (a) Design a Turing machine that accepts exactly those strings of 0's and 1's that are valid representations of Turing machines according to the coding system described in Lecture 12 (you do not need to enforce the ordering convention between transitions adopted therein). Explain the strategy your machine uses.

   (b) Describe how you would go about extending the machine designed in (a) to:

      i. ensure that the transitions of the machine input to it were ordered according to the convention of Lecture 12;

      ii. ensure that it accepted only deterministic Turing machines.

   (c) Show how the shifting machine S_R, with input alphabet $\{x, y\}$, may be represented as a sequence of transitions according to the coding scheme, initially using alphabetic (i.e. non-binary) representations of states and tape symbols in each transition. Show how this representation may be converted to a single string of 0's and 1's (it is not necessary to convert all of it) and confirm informally that your machine of part (a) accepts this representation of S_R.

3. Suppose that $p(n)$ is a polynomial expression in n and M is a Turing machine that accepts each string w in $L(M)$ before executing more than $p(|w|)$ steps (recall that $|w|$ denotes the length of w). Show that the language $L(M)$ is decidable. You may assume the existence of a Turing machine N that computes p.

   (Hint: think about using a multiple tape machine.)

Due This exercise should be completed by Monday, November 17th, since the solutions will be disclosed in that morning's tutorial.