582206 Models of Computation (Autumn 2009)
Exercise 12 (8–11 December)

1. Give a state diagram for a nondeterministic Turing machine that recognizes the language

$$\{\, \#w_1\#w_2\#\ldots\#w_n\# \mid w_i \in \{\,0,1\,\}^* \text{ for all } i \text{ and } w_i = w_j \text{ for some } i \neq j \,\}$$

   over the alphabet $\{\,0, 1, \#\,\}$.

2. Define the language

$$ALL_{\mathrm{DFA}} = \{\, \langle\, A \,\rangle \mid A \text{ is a DFA and } L(A) = \Sigma^* \,\}\,.$$

   Show that ALL_{DFA} is decidable.

3. Consider enumerators, as in Problem 6 of Exercise 11. Show the following.

   (a) A language is Turing-recognisable if and only if some enumerator enumerates it.

   (b) A language is decidable if and only if some enumerator enumerates it in the order of length starting from the shortest strings.

   (c) A language L is decidable if and only if both L and its complement $\overline{L}$ can be enumerated by an enumerator.

4. We are given the task of implementing a tool for program checking. According to the specification, the user can load a Java program into the tool. After that, the user can choose one line of code in the Java program, and give an input. The tool should tell whether the Java program with the given input would eventually execute the chosen line of code.

   Can such a tool be implemented? Will your answer change, if instead of the user specifying the input to the Java probram, the task is changed to deciding whether the chosen line of code would be executed on *any* possible input? Justify your answer.