1. Show that a one-tape Turing machine that never moves its tape head left is equivalent to a deterministic finite automaton.

2. Formally define a Turing machine $M$ that given a binary string on its input tape adds one to it and leaves the result on the input tape. Show the sequence of moves your Turing machine $M$ makes on the input 1011 starting off from the configuration $q_01011$ where $q_0$ is the initial state of $M$.

3. Show that there are potentially nine distinct ways of classifying a language $L$ and its complement $\overline{L}$ as being (a) recursive, (b) recursively enumerable but not recursive, and (c) not recursively enumerable. Which of these ways are actually possible?

4. Are the recursive languages closed under the Kleene star operator? Briefly justify your answer.

5. Is the language $L = \{ w_i \mid M_i \text{ does not halt on } w_i \}$ recursively enumerable? Here $w_i$ is an encoding the $i^{th}$ input string and $M_i$ is an encoding of the $i^{th}$ Turing machine. Use reductions and the diagonalization language $L_d$ to prove your answer.

6. Is the language $L = \{ M \mid M \text{ does not halt on } \epsilon \}$ recursively enumerable? Here $M$ is an encoding of a Turing machine. Use reductions and the diagonalization language to prove your answer.

aho@cs.columbia.edu and verma@cs.columbia.edu