CS3231
Tutorial 9

1. We say that a TM enumerates a language L, if and only if the TM (irrespective of the input) works as follows: On the output tape, it prints elements x_1, x_2, ..., such that $L = \{x_1, x_2, \ldots\}$. Note that elements may be printed in arbitrary order. Different elements are separated by some delimiter such as comma or #. It is assumed that once TM has printed an element, it cannot overwrite it. Note also that TM above may never halt.

   (a) Show that L is RE if and only if some TM enumerates L as above.

   (b) Suppose the TM as above lists the elements in increasing order, that is in the above listing $x_1 < x_2 < x_3 < \ldots$. Then show that L must be recursive.

   (Here, the comparison of strings x and y is done as follows: $x < y$, if $|x| < |y|$ or $[|x| = |y|$ and x is lexicographically before $y])$.

2. Consider the language $\{wcw^R \mid w \in \{a,b\}^*\}$ over the alphabet $\{a,b,c\}$. Here w^R denotes w in reverse (that is, if $w = a_1 a_2 \ldots a_{n-1} a_n$, with each $a_i \in \{a,b\}$, then $w^R = a_n a_{n-1} \ldots a_2 a_1$).

   Give a description of a 1-tape Turing Machine which can accept the above language. What is the time complexity of your machine?

   Give a description of a 2-tape Turing Machine which can accept the above language in linear time.

   You would note that the complexity of 2-tape Turing Machine is better. It can be shown that no 1-tape Turing Machine can accept the above language in linear time.

   Thus in some cases one can improve the complexity of accepting languages using more tapes.

3. (a) Show that the recursively enumerable languages are closed under (i) union and (ii) intersection.

   (b) Show that the recursive languages are closed under (i) union and (ii) intersection.

4. Suppose L_1 is recursive and L_2 is recursively enumerable. Then show that $L_2 - L_1$ must be recursively enumerable. Could you say anything about $L_1 - L_2$?

5. (a) Show that every finite language is recursive.

   (b) Show that every co-finite language is recursive.
   (Note: A language L is said to be co-finite if the complement of L, $\Sigma^* - L$, is finite.)

   (c) Suppose that L is a recursive language and D is a finite language. Then show that $L\boldsymbol{\Delta}D$ must be recursive. (Here $L\boldsymbol{\Delta}D$ denotes the symmetric difference, $(L - D) \cup (D - L)$, of L and D).

   (d) Suppose L is a recusively enumerable language which is not recursive. Suppose $\mathbf{M}$ is a Turing Machine which accepts L. Then show that there must be infinitely many inputs on which $\mathbf{M}$ does not halt.

6. (a) Suppose L is a recursive language. Then show that $\{x \mid xx \in L\}$ is also recursive.

   (b) Suppose L is a recursive language. Then show that $\{x \mid \text{some prefix of } x \text{ is in } L\}$ is also recursive.

7. Use the construction of Universal Turing Machine to show that the following language is recursive:

   $\{ba^iba^jba^t : M_i \text{ accepts } w_j \text{ in} \leq t \text{ time steps } \}$

   Here M_i is the i-th Turing Machine, w_j is the j-th string and t is a natural number.

8. Halting problem is defined as follows:

   Input: i and j.

   Question: Does M_i halt on input w_j?

   Show that the halting problem is not decidable.