Theory of Computation

homework 2
Due: 10/21/2014

Problem 1 Show that if L_1 and L_2 are recursively enumerable languages, then so is $L_1 \cup L_2$.

Ans: Since L_1 and L_2 are RE languages, there must be a TM M_1 accepting L_1 and a TM M_2 accepting L_2. Now we construct another TM M' that simulates M_1 and M_2 in an interleaving style. Make M' accept if M_1 or M_2 accepts. Now,

- if $x \in L_1$, $M'(x) = M_1(x) =$ "yes",
- if $x \in L_2$, $M'(x) = M_2(x) =$ "yes",
- if $x \notin L_1 \cup L_2$, $M'(x) = \nearrow$.

So M' accepts $L_1 \cup L_2$ and $L_1 \cup L_2$ is recursively enumerable. ■

Problem 2 Show that the language

$$A = \{(M; x) \mid M(x) = \text{"Yes"}\}$$

is undecidable.

Ans: Suppose A is recursive. Then there exists a TM M_A that decides A such that

- $M_A(M;x) = \text{“Yes”},\ \text{if } (M;x) \in A,$
- $M_A(M;x) = \text{“No”},\ \text{if } (M;x) \notin A.$

Consider the program $D(M)$ that calls M_A:

1: **if** $M_A(M;M) = \text{“Yes”}$ **then**

2: $\quad D(M) = \text{“No”};$

3: **else**

4: $\quad D(M) = \text{“Yes”};$

5: **end if**

Now, consider $D(D)$:

- $D(D) = \text{“No”},\ \text{if } M_A(D;D) = \text{“Yes”}.$
- $D(D) = \text{“Yes”},\ \text{if } M_A(D;D) = \text{“No”}.$

Note that, however, $M_A(D;D) = \text{“Yes”}$ implies $D(D) = \text{“Yes”}$ because $(D;D) \in A$. Similarly, $M_A(D;D) = \text{“No”}$ implies $D(D) \neq \text{“Yes”}$ because $(D;D) \notin A$. As contradiction follows in both cases, A is undecidable. ■