6.045J/18.400J: Automata, Computability and Complexity Prof. Ron Rivest

Homework 4

Due: 19 March 2003

Problem 1: Let L be an undecidable language. (Note that L may, be recognizable, or co-recognizable (that is, its complement may be recognizable), but not both.)

Prove that the language

$$L' = 0L \cup 1\overline{L}$$

is neither recognizable nor co-recognizable.

Problem 2:(Sipser 4.19 and 5.12) In this problem, let w^R be the reverse of the string w.

Part A: Let A be the language:

$$A = \{< M > | \text{M is a DFA such that M accepts } w \text{ if and only if M accepts } w^R\}$$

Prove that A is decidable. (Hint: Theorem 4.5)

Part B: Let B be the language:

$$B = \{< T > | \text{T is a TM such that T accepts } w \text{ if and only if T accepts } w^R\}$$

Prove that B is undecidable. (Hint: One proof can be based on theorem 5.2 and its proof).

Problem 3: In class we defined a language, L, to be enumeratable if there exists an enumerator, E, such that L is the set of strings that E prints out at some point during its execution. (Note that any particular string $x \in L$ may be printed out finitely many times or may be printed out infinitely many times). In this problem, we consider a related notion which we call *Uber-Enumeratable.* We say a language is *Uber-Enumeratable* if there exists an enumerator, E', such that L is the set of strings which are printed out infinitely many times during the execution of E'.

Part A: Prove that if L is enumeratable than L is *Uber-Enumeratable.*

Part B: Prove that there exists a language, L', such that L' is *Uber-Enumeratable* but not enumeratable. (Hint: Consider the language $\overline{A_{TM}}$).