CS530: Introduction to Computational Complexity	Xiang-Yang Li

Homework 3

Due: Oct 28th, 2008 — *YanWei Wu*

Problem 1: Give pseudocode for the Turing machine that accepts the following lan- guage: $L = \{xx^r w \mid x \in \Sigma^+, w \in \Sigma^+\}$. Here $\Sigma = \{a, b, c\}$.

Then draw the diagram of the Turing machine.

Problem 2: Without directly using Rice Theorem, prove that the following language is not decidable. $L = \{\langle M \rangle \mid M$ is a Turing machine and the size of $L(M)$ can be divided by $3\}$.

Problem 3: Which of the following languages is undecidable? Give proofs of your answer. You can use Rice Theorem whenever you can to say a language is not decidable.

1. $L = \{\langle M \rangle \mid M$ is a Turing machine and the size of $L(M)$ can be divided by $3\}$.
2. $L = \{\langle M \rangle \mid M$ is a Turing machine and M has at most 75 states$\}$.
3. $L = \{\langle M_1, M_2 \rangle \mid M_1$ and M_2 are Turing machines and $L(M_1) \cap L(M_2) = \emptyset\ \}$.
4. $L = \{\langle M \rangle \mid M$ is a Turing machine and $L(M)$ is recursive$\}$.

Problem 4: Prove that Turing machines as a generator is equivalent to Turing machines as acceptor. Here a TM G as a generator produces a language $L(G)$, if it will print all strings of $L(G)$ on the tape (here the printer will not go backward when printing).

In other words, you have to prove

1. Given a generator G, there is always a TM M such that $L(M) = L(G)$, and
2. Given a TM M, there is always a generator G, such that $L(M) = L(G)$.