**Discussion Problems 6**

**Problem One: Designing Turing Machines**

Draw the state transition diagram for a Turing machine whose language is $L = \{ w \in \Sigma^* \mid w \text{ is a palindrome} \}$ over the alphabet $\Sigma = \{0, 1\}$.

**Problem Two: Nondeterministic Algorithms**

A *computable function* is a function $f : \Sigma^* \to \Sigma^*$ with the following property – there is a TM $M$ that, when given $w$ on its input tape, always halts with $f(w)$ on its tape.

Given any computable function $f$ and language $L$, we define $f(L) = \{ w \in \Sigma^* \mid \exists x \in L. f(x) = w \}$. In other words, $f(L)$ is the set of strings formed by applying $f$ to each string in $L$.

Prove that if $L \in \text{RE}$ and $f$ is a computable function, then $f(L) \in \text{RE}$. As the title of this problem suggests, you might want to build a nondeterministic Turing machine for $f(L)$.

**Problem Three: Unsolvable Problems**

Consider the language $L = \{ \langle M, w, q \rangle \mid \text{TM } M \text{ does not enter state } q \text{ when run on string } w \}$. Prove that $L \notin \text{RE}$ by showing if $L \in \text{RE}$, then $L_D \in \text{RE}$.