**Problems for Week Eight**

**Problem One: Designing Regular Expressions**

Below are a list of alphabets $\Sigma$ and languages over those alphabets. For each language, write a regular expression for that language.

i. Let $\Sigma = \{a, b, c\}$ and let $L = \{ w \in \Sigma^* \mid w$ ends in $cab \}$. Write a regular expression for $L$.

ii. Let $\Sigma = \{a, b\}$ and let $L = \{ w \in \Sigma^* \mid w \neq \varepsilon$ and the first and last character of $w$ are the same $\}$. Write a regular expression for $L$.

iii. Let $\Sigma = \{a, b\}$ and let $L = \{ w \in \Sigma^* \mid w$ contains two $b$'s separated by exactly five characters $\}$. Write a regular expression for $L$.

iv. Let $\Sigma = \{a, b\}$ and let $L = \{ w \in \Sigma^* \mid w$ is a nonempty string whose characters alternate between $a$'s and $b$'s $\}$. Write a regular expression for $L$.

v. Let $\Sigma = \{a, b, c\}$ and let $L = \{ w \in \Sigma^* \mid w$ contains every character in $\Sigma$ exactly once $\}$. Write a regular expression for $L$.

**Problem Two: State Elimination**

Below is an NFA for a language from last week's packet of problems:

Using the state-elimination algorithm, convert this NFA into a regular expression. (You could just directly design a regular expression for this language, but we want you to specifically use the state elimination algorithm).
Problem Three: The Myhill-Nerode Theorem

The Myhill-Nerode theorem says the following:

Let $L$ be a language over $Σ$. If there is a set $S ⊆ Σ^*$ such that

- $S$ contains infinitely many strings, and
- any two distinct strings $x, y ∈ S$ are distinguishable relative to $L$ (that is, $x ≡_L y$),

then $L$ is not a regular language.

Below is a (slightly modified) version of the proof of the Myhill-Nerode theorem from lecture:

**Proof:** Let $L$ be an arbitrary language over $Σ$. Let $S ⊆ Σ^*$ be an infinite set of strings with the following property: if $x, y ∈ S$ and $x ≠ y$, then $x ≡_L y$. We will show that $L$ is not regular.

Suppose for the sake of contradiction that $L$ is regular. This means that there must be some DFA $D$ for $L$. Let $k$ be the number of states in $D$. Since $S$ is an infinite set, we can choose $k+1$ distinct strings from $S$ and run each of those strings through $D$. Because there are only $k$ states in $D$ and we've chosen $k+1$ distinct strings from $S$, by the pigeonhole principle we know that at least two strings from $S$ must end in the same state in $D$. Choose any two such strings and call them $x$ and $y$.

Since $x ∈ S$ and $y ∈ S$ and $x ≠ y$, we know that $x ≡_L y$. Consequently, by our earlier theorem, we know that $x$ and $y$ must end in different states when run through $D$. But this is impossible – we chose $x$ and $y$ specifically because they end in the same state when run through $D$. We have reached a contradiction, so our assumption must have been wrong. Thus $L$ is not a regular language. ■

This question explores the theorem in a bit more detail.

i. What is the formal definition of the statement $x ≡_L y$? Explain it in plain English. Give an example of two strings $x$ and $y$ along with a language $L$ where $x ≡_L y$ holds.

ii. The proof hinges on the fact that if $x ≡_L y$, then $x$ and $y$ cannot end in the same state when run through any DFA for a language $L$. We sketched a proof of this in class. Explain intuitively why this is the case.

iii. Explain, intuitively, why $S$ has to be an infinite set for this proof to work.

iv. Does anything in the proof require that $S$ be a subset of $L$?
Problem Four: Nonregular Languages Warmup

Let $\Sigma = \{1, \geq\}$ and consider the language $L = \{1^n \geq 1^n | m, n \in \mathbb{N} \text{ and } m \geq n \}$.

i. Give some specific examples of strings from the language $L$.

ii. Without using the Myhill-Nerode theorem, give an intuitive justification for why $L$ isn't regular.

iii. Use the Myhill-Nerode theorem to prove that $L$ isn't regular. You'll need to find an infinite set of strings that are pairwise distinguishable relative to $L$. As a hint, see if you can think of some strings that would have to be treated differently by any DFA for $L$, then see what happens if you gather all of them together into a set.

Problem Five: Nonregular Languages

Here are some more problems to help you get used to proving that certain languages aren't regular.

i. Let $\Sigma = \{a, b\}$ and let $L = \{a^n b^m | n, m \in \mathbb{N} \text{ and } n \neq m \}$. Explain why this language is not the complement of the language $\{a^n b^n | n \in \mathbb{N} \}$.

ii. Let $\Sigma = \{a, b\}$ and let $L = \{a^n b^m | n, m \in \mathbb{N} \text{ and } n \neq m \}$. Prove that $L$ is not regular.

iii. Let $\Sigma = \{a\}$ and let $L = \{w \in \Sigma^* | w \text{ is a palindrome }\}$. Prove that $L$ is regular.

iv. Let $\Sigma = \{a, b\}$ and let $L = \{w \in \Sigma^* | w \text{ is a palindrome }\}$. Prove that $L$ is not regular.