1. (25 points) Give an NFA (both a state diagram and a formal description) recognizing the language $0^*1^*0^+$ with three states. The alphabet is $\{0, 1\}$. You don’t need to have a formal proof.

2. (25 points) This question studies the number of states in a DFA equivalent to an NFA. Recall that in class we showed an NFA with 4 states that recognizes the language which consists of all binary strings that have a 1 in the third position from the end. For any integer $k$, it is easy to generalize this construction to an NFA with $k + 1$ states that recognizes the language which consists of all binary strings that have a 1 in the $k$'th position from the end. The general transformation from an NFA to a DFA will give us a DFA with at most $2^{k+1}$ states recognizing the same language.

Show that, any DFA that recognizes the same language must have at least $2^k$ states.

**Hint:** start by looking at the following two strings: $10^{k-1}$ and $0^k$. Observe that when a DFA takes them as inputs, it must end up at different states, since one string is accepted and the other is rejected.

3. (25 points) Say that string $x$ is a prefix of string $y$ if a string $z$ exists where $xz = y$ and that $x$ is a proper prefix of $y$ if in addition $x \neq y$. Let $A$ be a regular language. Show that the class of regular languages is closed under the following operation.

$$\text{NOEXTEND}(A) = \{w \in A | w \text{ is not the proper prefix of any string in } A\}$$

**Hint:** Think about when a string $w \in A$ can be the proper prefix of another string in $A$, then modify the states of the machine to avoid this.

4. (25 points) Let $\Sigma = \{0, 1\}$.

(a) Write a regular expression for the language $L$ consisting of all strings in $\Sigma^*$ with exactly one occurrence of the substring $111$.

(b) Write a regular expression for the language $L$ consisting of all strings in $\Sigma^*$ that do not end with $10$.

You don’t need to have a formal proof.