You have 90 minutes to complete this exam. You may state without proof any fact taught in class or assigned as homework.

1. Give a simple verbal description of the language recognized by the following NFA with alphabet \{0, 1, 2\}:

2. Draw NFAs for the following languages, taking full advantage of nondeterminism:
   a. binary strings that start with a 1 or have a 1 in the third position from the end;
   b. binary strings that contain 01 or 10 but not both.
3 Prove that the following languages over the binary alphabet are regular:

(2 pts) a. even-length strings that contain 0101;
(2 pts) b. strings in which every 1 is adjacent to a 0;
(2 pts) c. strings in which the substring 01 occurs an even number of times.
For languages $A$ and $B$ over a given alphabet $\Sigma$, define $A \circ B$ to be the set of all strings in $A$ that do not contain a substring that is in $B$. Prove that regular languages are closed under the $\circ$ operation.

Let $L$ be a given regular language. Define $L^\dagger$ to be the set of all strings obtained by taking a nonempty string in $L$ and removing its last symbol. Prove that $L^\dagger$ is regular.
Describe an algorithm that takes as input an NFA $N$ and outputs the minimum length of a string rejected by $N$. If no such string exists, the algorithm should output $\infty$. Your algorithm must run in finite time.

Let $D$ be a given DFA. Let $W$ be the set of all strings $w$ such that every state of $D$ is visited while processing $w$. Prove that $W$ is regular.
SOLUTIONS
You have 90 minutes to complete this exam. You may state without proof any fact taught in class or assigned as homework.

1. Give a simple verbal description of the language recognized by the following NFA with alphabet \{0, 1, 2\}:

Solution: nonempty strings in which the first symbol occurs only once.

2. Draw NFAs for the following languages, taking full advantage of nondeterminism:

   a. binary strings that start with a 1 or have a 1 in the third position from the end;
   b. binary strings that contain 01 or 10 but not both.

Solution.
Prove that the following languages over the binary alphabet are regular:

(a) even-length strings that contain 0101;
(b) strings in which every 1 is adjacent to a 0;
(c) strings in which the substring 01 occurs an even number of times.

Solution.

(a) This language is given by the intersection \( A \cap B \), where \( A \) is the set of even-length strings and \( B \) is the set of strings that contain 0101. We proved in class that \( A \) and \( B \) are both regular. By the closure properties of regular languages, \( A \cap B \) is regular as well.

(b) This language is recognized by the following NFA:

![NFA Diagram]

(c) This language is recognized by the following DFA:

![DFA Diagram]
For languages \( A \) and \( B \) over a given alphabet \( \Sigma \), define \( A \circ B \) to be the set of all strings in \( A \) that do not contain a substring that is in \( B \). Prove that regular languages are closed under the \( \circ \) operation.

\[
A \circ B = A \setminus (\Sigma^* B \Sigma^*).
\]

**Solution.** Let \( A \) and \( B \) be regular. We have \( A \circ B = A \setminus (\Sigma^* B \Sigma^*) \). Here \( A \) and \( B \) are regular by hypothesis, and \( \Sigma \) is regular because it is finite. Since regular languages are closed under Kleene star, concatenation, and set difference, we conclude that \( A \circ B \) is regular.

Let \( L \) be a given regular language. Define \( L^\dagger \) to be the set of all strings obtained by taking a nonempty string in \( L \) and removing its last symbol. Prove that \( L^\dagger \) is regular.

**Solution.** Let \( D = (Q, \Sigma, \delta, q_0, F) \) be a DFA for \( L \). Then \( L^\dagger \) is recognized by the DFA \( (Q, \Sigma, \delta, q_0, F^\dagger) \), where \( F^\dagger = \{ q \in Q : \delta(q, \sigma) \in F \text{ for some } \sigma \} \). The new DFA operates like the old one but accepts the input if and only if it can be extended by a single character to a string that \( D \) would accept.
Describe an algorithm that takes as input an NFA $N$ and outputs the minimum length of a string rejected by $N$. If no such string exists, the algorithm should output “$\infty$.” Your algorithm must run in finite time.

**Solution.** Convert $N$ to an equivalent DFA $D$ via the construction given in class. Then, view $D$ as a directed graph and run breadth-first search from the initial state to compute the distance to the nearest rejecting state. If it so happens that no rejecting state is reachable in $D$ from the initial state, output “$\infty$.”

Let $D$ be a given DFA. Let $W$ be the set of all strings $w$ such that every state of $D$ is visited while processing $w$. Prove that $W$ is regular.

**Solution.** Let $D = (Q, \Sigma, \delta, q_0, F)$ be the given DFA. Then $W$ is recognized by the DFA $(Q \times \mathcal{P}(Q), \Sigma, \Delta, (q_0, \{q_0\}), Q \times \{Q\})$, where $\Delta((q, S), \sigma) = (\delta(q, \sigma), S \cup \{\delta(q, \sigma)\})$. The DFA for $W$ operates just like $D$ but additionally keeps track of the set of states visited so far, accepting if and only if all states have been visited.