1. Be sure to know how to define (or state in the case of theorems) each of the following. (You must give definitions and statements like the ones presented in class. Check your notes!)

- difference of two languages
- symmetric difference of two languages
- nondeterministic finite automaton \( M = (Q, \Sigma \cup \{\lambda\}, \delta, q_0, F) \) (i.e., the 5-tuple definition including the meaning of each member of the tuple)
- \( \delta^*(q, \lambda) \) for an nfa (with \( \lambda \)- transitions) where \( q \) is a state and \( \lambda \) is the empty string
- \( \delta^*(q, a) \) for an nfa (with \( \lambda \)- transitions) where \( q \) is a state and \( a \) is an input symbol
- language of an nfa \( M \)
- equivalent finite automata
- Rabin-Scott Theorem
- regular expression over an input alphabet \( \Sigma \)
- primitive regular expression
- Kleene’s Theorem
- production
- \( u \xrightarrow{*} v \)
- a grammar \( G = (V, T, S, P) \)
- the language \( L(G) \) of a grammar \( G = (V, T, S, P) \)
- equivalent grammars

2. Consider the finite automata \( M_1 = (Q_1, \Sigma, \delta_1, q^1_0, F_1) \) and \( M_2 = (Q_2, \Sigma, \delta_2, q^2_0, F_2) \) in Figure 1. Construct the ‘cross product’ automaton \( M = (Q_1 \times Q_2, \Sigma, \delta, (q^1_0, q^2_0), F_1 \times F_2) \) described in class. What language does your automaton accept?

3. If \( L_1 \) and \( L_2 \) are languages, let \( L_1 \odot L_2 \) be defined as

\[
L_1 \odot L_2 = \{ w \in \Sigma^* : w \notin L_1 \text{ and } w \notin L_2 \}.
\]

Prove that if \( L_1 \) and \( L_2 \) are regular languages, so is \( L_1 \odot L_2 \).

4. Consider the nondeterministic finite automaton \( M \) in Figure 2. Describe \( L(M) \).
5. Let $\Sigma = \{a, b, c, d\}$. Construct a nondeterministic finite automaton to accept the finite language $L = \{\lambda, ab, bcdd, acba, c\}$.

6. Given a finite automaton $M$ (deterministic or nondeterministic), construct an nfa $M'$ that is equivalent to $M$ and has exactly one final state.

7. For the nondeterministic automaton $M$ in Figure 3, construct an equivalent deterministic finite automaton using the technique discussed in class.

8. Construct regular expressions for each of the following languages:
   (a) $L_1 = \{w \in \{0, 1\}^* : w$ consists of an alternating sequence of 0s and 1s$\}$
   (b) $L_2 = \{w \in \{a, b, c\}^* : w$ contains at least one $a$ and at least one $b$\}$
   (c) $L_3 = \{w \in \{0, 1\}^* :$ the tenth symbol from the right is a 1$\}$
   (d) $L_4 = \{w \in \{0, 1\}^* :$ every pair of adjacent 0s appears before any pair of adjacent 1s$\}$
   (e) $L_5 = \{w \in \{0, 1\}^* :$ the number of 0s is divisible by 5$\}$

9. Give simple English descriptions of the languages given by the following regular expressions:
   (a) $(1 + \lambda)(00*1)^*0^*$
   (b) $(0*1^*)^*000(0 + 1)^*$
   (c) $(0 + 10)^*1^*$

10. Construct nfas accepting the languages given by the following regular expressions:
    (a) $01^*$
    (b) $00(0 + 1)^*$

11. Suppose $M = (Q, \Sigma, \delta, q_0, F)$ is an nfa such that there are no transitions into $q_0$ and out of $q_f$ where $q_f$ is a unique final state. Describe the language accepted by each modification of $M$ in terms of $L = L(M)$:
    (a) The automaton constructed from $M$ by adding a $\lambda$–transition from $q_f$ to $q_0$.
    (b) The automaton constructed from $M$ by adding a $\lambda$–transition from $q_0$ to every state reachable from $q_0$ along a path whose labels may include symbols from $\Sigma$ as well as $\lambda$.
    (c) The automaton constructed from $M$ by adding a $\lambda$–transition to $q_f$ from every state that can reach $q_f$ along some path.
    (d) The automaton constructed from $M$ by doing both (b) and (c).

12. Suppose $r$ and $s$ are regular expressions. Is it true that $(r + s)^* = r^* + s^*$. If so, prove it. If not, present a counterexample.
13. Suppose we try to apply the Pumping Lemma for regular languages to the language of the regular expression \((00 + 11)^*\). Show how the adversary wins no matter what we do!

14. Find a grammar that generates the language \(L = \{w01 : w \in \{0, 1\}^*\}\).

15. Consider the grammar \(G = (\{S\}, \{(\), \}, S, P)\) where \(P = \{S \rightarrow SS \mid (S) \mid \lambda\}\). Determine \(L(G)\).

16. Consider the grammar \(G = (\{S\}, \{a, b\}, S, P)\) where \(P = \{S \rightarrow aSbS \mid bSaS \mid \lambda\}\). Give a simple English description of \(L(G)\).

17. Consider the grammar \(G = (\{S, A, B\}, \{0, 1\}, P)\) where \(P = \{S \rightarrow A1B, A \rightarrow 0A \mid \lambda, B \rightarrow 0B \mid 1B \mid \lambda\}\). Determine \(L(G)\).