1. The operator * in regular expressions is called **Kleene closure**

2. True or False. Justify your answer in either case.

   The set of languages accepted by nondeterministic finite automata is a proper subset of the set of languages accepted by deterministic finite automata.

   **False. Both classes are the same, the class of regular languages.**

3. True or False. The following grammar is ambiguous.

   \[ S \rightarrow aSbS | bSaS | \epsilon \]

   **True.** \[ S \Rightarrow aSbS \Rightarrow abS \Rightarrow ababS \Rightarrow abab \] and \[ S \Rightarrow aSbS \Rightarrow abSaSbS \Rightarrow ababS \Rightarrow abab \] are 2 different left-most derivations for abab.

4. List the grammars in the **Chomsky hierarchy** and show the containment relationship of their corresponding language classes.

   Regular languages are contained in context-free languages which are contained in context-sensitive languages which are contained in recursively enumerable languages. Each is defined by a grammar class of the same name, except that recursively enumerable languages are defined by unrestricted grammars.

5. Which of the following languages is not regular.

   (a) \[ L = \{ w \mid w \in \{0, 1\}^*, \text{the number of 0's + the number of 1's is odd} \} \]
   (b) \[ L = \{ 0^i1^j \mid i \geq 0, j \geq 0, \text{if } i \text{ is odd, } j \text{ is even, if } i \text{ is even, } j \text{ is odd} \} \]
   (c) \[ L = \{ w#w^R \mid w \in \{0, 1\}^* \} \]
   (d) \[ L = \{ w#w^R \mid w \in \{0, 1\}^*, |w| \leq 4115 \} \]
   (e) \[ L \text{ is the set of all string constants in a programming language like C, C++ or Java (e.g., \text{"hello"}, \text{"!#$%&*"}, \text{"Am I regular?"}).} \]

   **(c) is not regular. It is not possible to compare a string of arbitrary length against its reverse.**

6. Write a finite automaton (deterministic, nondeterministic, or nondeterministic with \(\epsilon\)-transitions) to recognize the language over \(\{a, b, c\}\)\(^*\) such that every string of \(a\)'s must be followed by at least one \(b\) and every string of \(b\)'s must be followed by at least one \(c\).

   \[ M = (\{q_0, q_1, q_2\}, \{a, b, c\}, \delta, q_0, \{q_0\}) \]
   \[ \delta(q_0, a) = q_1 \]
   \[ \delta(q_0, b) = q_2 \]
   \[ \delta(q_0, c) = q_0 \]
   \[ \delta(q_1, a) = q_1 \]
   \[ \delta(q_1, b) = q_2 \]
   \[ \delta(q_2, b) = q_2 \]
   \[ \delta(q_2, c) = q_0 \]
7. Using the pumping lemma for regular languages, prove that the language \( L = \{b_i\#b_{i+1}^R \mid b_n \text{ is the binary representation of } n\} \) is not regular. Note that \( L \) includes 0\#1, 1\#01, 10\#11, 11\#001, etc.

Let \( w = 1^n\#0^n1 = xyz \). By the pumping lemma, \( xyyz \) will be in \( L \). But it cannot be in \( L \) since pumping any string containing the first \( n \) symbols will result in a string with more 1’s to the left of \# than there are 0’s to the right of \#.

8. Write a context-free grammar for the language \( L = \{b_i\#b_{i+1}^R \mid b_n \text{ is the binary representation of } n\} \). Note that \( L \) includes 0\#1, 1\#01, 10\#11, 11\#001, etc.

Note that for any binary number \( b_i = a_1a_2...a_k01^m \), \( b_{i+1} = a_1a_2...a_k10^m \), so this language is \( w0^m\#0^m1w^R \). The grammar therefore has productions:

\[
S \rightarrow A \mid B1 \\
A \rightarrow 0A0 \mid 1A1 \mid 0B1 \mid 0#1 \\
B \rightarrow 1B0 \mid 1#0 \\
B \Rightarrow 1^m\#0^m (m \geq 1). \quad A \Rightarrow w0Blw^R \Rightarrow w0^1m\#0^m1w^R (m \geq 0). \quad S \rightarrow B1 \text{ takes care of the case where } b_i \text{ contains no } 0.
\]