Quiz 1

1. Draw a DFA that accepts \( L = \{ x \in \{1,2\}^* : x \text{ has exactly two 2's} \} \).

2. List, in order, the lexicographically-first four strings of \((111)^*(11111)^*\).

3. Write a regular expression for the language \((aa)^*\). The complement is relative to the alphabet \(\Sigma = \{a\}\).

4. Every NFA-acceptable language can be accepted by an NFA with just a single final state.  
   \[ \text{True} \quad \text{False} \]

5. Every subset of a regular language is regular.  
   \[ \text{True} \quad \text{False} \]

6. \(L^*\) is infinite.  
   \[ \text{True} \quad \text{False} \]

7. If \( M = (Q, \Sigma, \delta, q_0, F) \) is a DFA and \( F = Q \) then \( L(M) = \Sigma^* \).  
   \[ \text{True} \quad \text{False} \]

8. If \( L \) is accepted by an \( n \)-state NFA then \( L \) is accepted by some \( 3^n \)-state DFA.  
   \[ \text{True} \quad \text{False} \]

9. If \( L \) is a not-regular language and \( F \) is a finite language then \( L \cap F \) is a regular language.  
   \[ \text{True} \quad \text{False} \]

10. \((L^*)^* = L^*\).  
    \[ \text{True} \quad \text{False} \]

11. For \( \alpha \) a regular expression, there is an algorithm to decide if \( x \in L(\alpha) \) that is efficient enough to run in a reasonable amount of time on reasonable length \( x, \alpha \).  
    \[ \text{True} \quad \text{False} \]

12. Let \( M = (Q, \{0,1\}, \delta, q_0, F) \) be a DFA and suppose that \( \delta^*(q_0, x) = \delta^*(q_0, y) \). Then \( x \in L(M) \) if and only if \( y \in L(M) \).  
    \[ \text{True} \quad \text{False} \]