Quiz 2

Try to get each question fully right — likely no partial credit will be given.

1. Using the procedure explained in class and in your text, convert the following regular expression into an NFA for the same language: 
\[(ab)^*\]. Do not simplify.

2. Draw a smallest DFA that accepts 
\[L = \{x \in \{0, 1\}^* : \text{the number that } x \text{ represents, in binary, is divisible by } 3\} = \{0\}^*\{\varepsilon, 11, 110, 1001, \ldots\}\]. (smallest = fewest states)

3. Every DFA-acceptable language can be accepted by an DFA with just a single final state.
   Explain: True False

4. If \(\alpha\) and \(\beta\) are regular expressions then there is a regular expression for \(L(\alpha) \cap L(\beta)\).
   Explain: True False

5. If \(M = (Q, \Sigma, \delta, q_0, F)\) is an NFA and \(F = Q\) then \(L(M) = \Sigma^*\).
   Explain: True False

6. If \(L\) is accepted by an \(n\)-state NFA then \(L\) is accepted by some \(n\)-state NFA.
   Explain: True False

7. If \(L\) is DFA-acceptable and \(F\) is finite then \(L \cap F\) is a DFA-acceptable.
   Explain: True False

8. Carefully state the **pumping lemma** for regular languages. (Any form of the pumping lemma is fine. Don’t use the word “pumps.” No credit if quantifiers are wrong on ambiguous.)