Theory of Computation - Assignment 1 - Due: 17-Feb-2025, 11 PM

1. The 'finite' in FSA refers to
   (a) Finite number of states
   (b) Length of the string is finite
   (c) Both (A) and (B)
   (d) None
2. The language accepted by FSA can be
   (a) Finite or Infinite
   (b) Must be Finite (always)
   (c) If it is infinite, then it must have infinite number of states
   (d) Must be Infinite (always)
3. How many different FSAs are possible for the language $L = \Sigma^*, \Sigma = \{0, 1\}$
   (a) Finitely many
   (b) Infinitely many
   (c) No FSA exists
4. Are there FSAs in which every state is a final state.
   (a) YES
   (b) NO
5. Let $\Sigma = \{0, 1\}$. Let L be the language defined over Σ such that the number of 0's is a multiple of 3 and the number of 1's is a multiple of 4. The number of states is
   (a) 3
   (b) 4
   (c) 7
   (d) 12
6. How many DFAs are possible for the language $\{x \mid x$ begins with 0 and ends with 1 $\}$.
   (a) It is unique and exactly one
   (b) NO DFA exists
   (c) at least 3 FAs
   (d) None of the above
7. Tick all that are true. Let $\Sigma = \{1\}$. For the regular expression $(11111 + 111)^*$,
   A. the language $L = \{\epsilon, 111, 11111, 1^8, 1^9, 1^{10}, 1^{11}, ...\}$.
   B. there exists a minimal DFA with 9 states
   C. there exists a DFA with 8 states
   D. there is no NFA with 7 states.

8. Consider a DFA with 4 states such that all states are final states. Then, which of the following are true
A. $L = \Sigma^*$.
B. Any minimal DFA has exactly one state.
C. $L \neq \Sigma^*$.
D. there exists an equivalent epsilon NFA with 4 states.

9. How many different two state DFAs are possible for $\Sigma = \{a, b, c\}$. A. 256
B. 512
C. 128
D. None of the above

10. Let L be a language over $\{a, b\}$ with the property that all strings in L are of odd length. Which of the following is (are) the regular expressions for L.
A. $(a + b)(aa + ab + ba + bb)^*(b + a)$
B. $(a + b)(aa + ab + ba + bb)^*$
C. $(a + b)(aa + ab + ba + bb)^* + (aa + ab + ba + bb)^*(b + a)$
D. $(aa + ab + ba + bb)^*(b + a)$

11. Tick all that are true. The regular expression for the set of strings over $\{0, 1\}$ not containing 11 as a substring.
A. $(00 + 01 + 10)^*$
B. $0^* + 0^*1 + (0^*10^*)^*$
C. $\epsilon + 1 + 0^* + (0^*10^*)^*$
D. $(10 + 0)^* + (10 + 0)^*1$.