National Institute of Technology, Silchar

Subject Code : CS-204, Subject: Theory of Computation

Semester: 4th . Branch: Computer Sci. & Engg.

Duration: One Hour & Fifteen Minutes. Total Marks: 30

Figure in the right hand margin indicates full marks for the question.

All questions are compulsory (NO-NEGATIVE Marks for wrong Answers)

NOTE:

Fill google-form and submit your response along with your name, institute email ID and scholar No. in the classwork section at Google Classroom.

Q1. When converting NFAs to DFAs using the subset construction, we noted that the number of states in a DFA can become exponentially larger than the number of states in the corresponding NFA. However, the DFA that results from the subset construction is not necessarily a minimal DFA (least number of states) for the language; hence it can be subject to the DFA minimization algorithm discussed in class. The question that arise naturally is whether NFAs really help in representing languages exponentially more compactly than minimized DFAs. **6**

What do you think?

a) Yes, NFAs really help in representing languages exponentially more compactly than minimized DFAs.

b) No, NFAs do not help in representing languages exponentially more compactly than minimized DFAs.

c). Can't say for sure.

d). The question itself is wrong.

Q2. Let $\sum$ be an alphabet. Let $L_1, L_2 \subseteq \sum^*$ be languages such that L_1 is regular and L_2 is not. In addition, we know that: L_2 is not $\subseteq L_1$; L_1 is not $\subseteq L_2$; and $L_1 \cap L_2$ is an infinite set. We are interested in the regularity of $L_1 \cap L_2$. **6**

What do you think?

a) $L_1 \cap L_2$ is necessarily regular.

b) $L_1 \cap L_2$ is necessarily non-regular.

c). Can't say for sure.

d). The question itself is wrong.

Q3. Which of the following languages is\are Context-free language(s)? **6**

a) $L = \{ 0^i 1^j 2^k : i < j < k\}$.

b) $L = \{ w \in (0 + 1 + 2)^* :$ w does not contain the same number of all three symbols$\}$.

c) $L = \{ uawb : u, w \in (a + b)^* , |u| = |w| \}$.

d) $L = \{ b_i \# b_{i+1} : b_i$ is i in binary, $i \geq 1\}$ when $\Sigma = \{0, 1, \#\}$.

Q4. Let $L \subseteq (0+1)^*$ be the language {(M,x) : Turing machine M on input x enters every state of M at least once}. Assume M to be a standard single-tape Turing machine encoded in binary as we discussed in class. The string x is also assumed to be a binary string. **6**

What do you think?

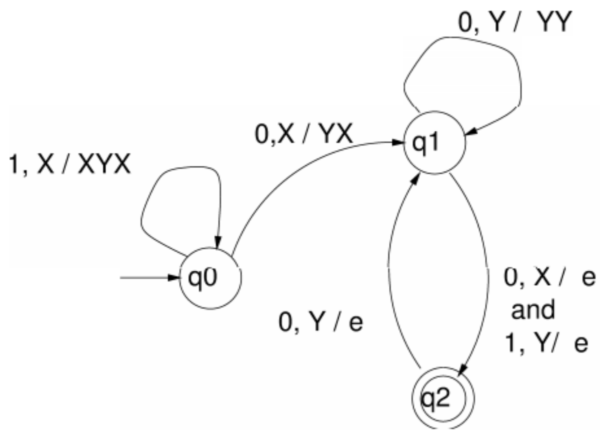
a) There necessarily exists a membership algorithm for L.

b) There exists no membership algorithm for L.

c) Can't say for sure it's undecidable.

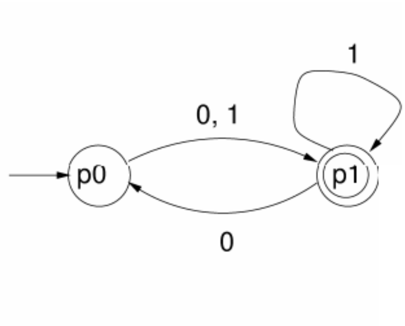
d) The question itself is wrong.

Q5. Consider the Push-down acceptor (PDA) and Deterministic finite acceptor (DFA) given below. The PDA is assumed to start operation with X as the only symbol in the stack. Let L_1 be the language accepted by the PDA by final state and let L_2 be the language accepted by the DFA. The alphabet for both languages is $\Sigma = \{0,1\}$. Define $L_1 / L_2 = \{ w \in \Sigma^* :$ there exists $u \in L_2$ such that $wu \in L_1 \}$. **6**



Push Down Automata

Arc labels: input, stack-top / replacement of stack-top



Deterministic Finite Automata

What do you think? The language $L = L_1 / L_2$ is :

a) Regular language.

b) Not Regular but a Context-free language.

c) Not Context-free but a Context-sensitive language.

d) Not Context-sensitive but a Recursively-enumerable language.

*****