TOC-IV Sem CSE, 1st Mid term examination

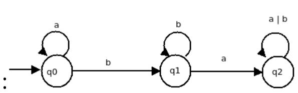
Solution Hints

March 27, 2019

Part A: Attempt All questions

1. 1A: True. This is because, in NFA for a single symbol there can be multiple transitions, hence for a same string there can be multiple paths. Each such path is like a DFA solving it.
2. 1B: Answer: Regular expression
3. 1C: Answer: To recognize tokens, as lexical analyzer in compiler, to recognize string patterns.
4. 1D: Answer: {a, ab, abb, abbb, ...}
5. 1E: Answer: True, as they generate the same language.
6. 1F: Answer: 2.
7. 1G: Answer:True
8. 1H: Answer: Exchange the final and non-final states.
9. 1I: Answer: False
10. 1J: Answer: 256

Part B: Attempt Any two questions

1. Q2: As shown here: 

2. Q2: As shown Table here:

state	Input (a)	Input (b)
q_0	q_0	q_1
q_1	q_2	q_1
q_2	q_2	q_2

3. Q3: (a) $a(b + c + d)$

4. Q3:(b) $a(b + c) + b(b + c)$

5. Q3:(c) ab^*

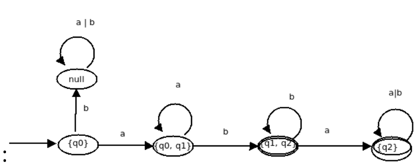
6. Q4: Assume that length of string is longer than number of states. Hence, there is a loop, for part of string a^k and k>=1. Hence, $\delta^*(q_0, a^{n-k}b^n)$ leads to accept state. In addition, the string passing through the loop is accepted at state $\delta^*(q_0, a^n b^n)$. Since this machine accepts two different languages, hence it is not an automata.

7. Q4: The other approach for this question is as follows: Let us assume that two strings a^k and a^m are indistinguishable. That is, they lead the FA from q_0 to same state ($a^k \approx a^m$) for $m \neq k$. Now, we supply string a^m which makes each string as $a^k a^m$ and $a^m a^m$. We note that $a^k a^m \neq a^m a^m$. That is, these states are distinguishable. Such states can be infinitely large (for every value of k). Since the automata has infintley large number of states, it is not a FA.

8. Q5: Answer shall comprise, a physical model of Moore machine, two tapes (one input and other output), the meaning of the tuples in $(Q, \Sigma, \delta, q_0, \Gamma, \lambda)$, and an example of transition diagram to demonstrate the generation of an output string for a given input string.

Part C: Attempt Any two questions

1. Q6: Pumping lemma is used to show if a given language is non-regular. It is useful only for infinite languages. It cannot tell, if a given language is regular. To test the non-regularity of a language, we breakup the given string of the language as $w = xy^k z$, with $k \geq 0$, $y \geq 1$. Then for some value of $k \geq 0$ we should be able to show that $w \notin L$, that is, it does not belong to the language. For the given language $\{a, aaaa, ...\}$ the proof is as follows. Let w, the input follows the square law, i.e. $|w| = n^2$, and $|y| = m \geq 1$, for m, n. Hence, $|w| = |xz| + |y| = n^2 = n^2 - m + m$. Let $k = 2$, hence, $|xy^2z| = |xz| + |y^2| = (n^2 - m) + 2m = n^2 + m \leq n^2 + n$, as m can be maximum n. Hence, $|w| = n^2 \leq n^2 + n$. The next perfect square is $n^2 + 2n + 1 = (n+1)^2$. So, $n^2 \leq n^2 + n < (n+1)^2$. Since w falls between two perfect squares, the w is not always square. This goes against our assumption that it is a perfect square. Hence, as per the pumping lemma, the language is not regular.

2. Q7: equivalent DFA: 

3. Q8: The answer shall comprise, diagram of FA, mathematical model $(Q, \Sigma, \delta, q_0, F)$ and their definition. Apart from this, the working and language recognition process needs to be described.