t.me/vitquestionpapers_2k19

VIT UNIVERSITY

VELLORE • CHENNAI

www.vit.ac.in

School of Computer Science and Engineering

A1+TA1+TAA1-Slot CAT-I (Aug-2018)

Subject: ToC and Compiler Design – CSE2002

Time: 1 Hr 30 Mins **Max.Marks:50**

Answer ALL questions

(5 X 10 = 50 marks)

1. State true or false for the given argument? **[Each 1M]**
   (a) For any regular languages L_1 and L_2 over Σ, $L_1 - L_2$ is also regular.
   (b) If $L \subseteq \Sigma^*$ and L is finite, then $\Sigma^* - L$ is regular.
   (c) If $L, L' \subseteq \Sigma^*$, and both L and L′ are infinite, then $L \cap L'$ is also infinite.
   (d) The transition function of a DFA is of the form $\delta: Q \times \Sigma \rightarrow 2^Q$.
   (e) If L is regular, then $L' = \{aw \mid w \in L\}$ is regular
   (f) The language of the regular expression $(\varphi)^*a^*$ is empty
   (g) For any NFA with n states, there is a DFA with at most 2^n states that accepts the same language as the NFA
   (h) The set of all words mentioned in all the pages on the Internet at this moment is a regular language.
   (i) If L is regular and $L' \subseteq L$, then L′ is regular
   (j) Every subset of a regular set is regular

2. (a) A new programming language is being designed that includes decimal numbers with no exponent. A decimal number has a period to separate the integer and fractional parts of the number and must have at least one digit both before and after the decimal point. Furthermore, a decimal number must not have any superfluous leading or trailing 0's. Some examples of legal decimal numbers are 3.14, 0.01, 17.0, 0.0, 123.00321. Some examples of illegal decimal numbers are 00.0, .5, 17, 17., 003.01, 0.0012300 (this last example is illegal because of the trailing 0's; the 0 before the decimal point is required).
   Write a regular expression (or collection of regular expressions) to generate legal decimal numbers. **[4]**

   (b) There are two rooms A and B with lights, but with a single control switch for both rooms. Hence lights in rooms are both on or both off at any point. The goal is to build an automatic control system that manages this switch. There is a sensor that detects motion in the two rooms and sends data to the controller; the controller reads these two signals and then instructs whether the switch should be turned on or off. We would like the controller to turn the lights on when motion is detected in either room, and turns them off if both rooms are empty for two consecutive signals from the sensor. Assume that the system starts from the state when lights are off. The sequence of events and actions is represented by the alphabet $\Sigma = \{yes, no, on, off\}$. Sequences accepted are of the form: $r_1\, s_1\, t_1\, r_2\, s_2\, t_2 \ldots r_n\, s_n\, t_n$ where each $r_i \in \{yes, no\}$ and stands for the signal coming from Room A, each $s_i \in \{yes, no\}$ stands for the signal coming from room B, and each $t_i \in \{on, off\}$ stands for the instruction the controller gives to the switch. **[6]**

   Design a DFA that accepts precisely the sequences that conform to the behavior of the controller. Example of good sequences:
   (a) no, no, off
   (b) yes, yes, on, yes, no, on, no, no, ~~on~~ off, no, no, off
   (c) no, yes, on, no, no, on, no, yes, on
   (d) yes, no, on, no, no, on, no, yes, on

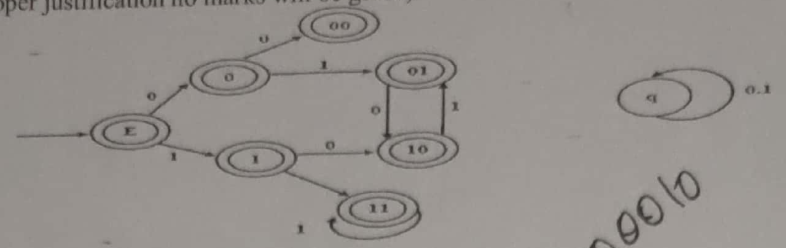
   Example of bad sequences:
   (a) yes, no, on, no incomplet
   (b) no, no, yes ∉ risiti
   (c) yes, yes, off

(D-{0}) or 0) D* or (D*

D-{0} D* or 0.0 or D* (D-{0})

Scanned by CamScanner

3. (a) Consider the set of strings on {0,1} in which, every substring of 3 symbols has at most two zeros. For example, 001110 and 011001 are in the language, but 100010 is not. All strings of length less than 3 are also in the language. A partially completed DFA that accepts this language is shown below. (Without proper justification no marks will be given) [4]



100010

The missing arcs in the DFA are

(A)

	00	01	10	11	q
00	1	0			
01				1	
10	0				
11			0		

(B)

	00	01	10	11	q
00		0			1
01		1			
10				0	
11		0			

(C)

	00	01	10	11	q
00		1			0
01		1			
10			0		
11		0			

(D)

	00	01	10	11	q
00		1			0
01				1	
10	0				
11			0		

(b) The symmetric difference of languages L and M, which we shall denote SD(L,M), is the set of strings that are in exactly one of L and M. For example, if L = {00, 101} and M = {11, 00}, then SD(L,M) = {11, 101}. [6]

(a) Suppose L = L(0*1*) and M = L(1*0*). What are all the strings of length 3 or less in SD(L,M)?

(b) Write a regular expression for SD(L,M).

(c) Write a formula for SD(L,M) in terms of familiar operations

4. (a) Give state diagrams of NFAs with the specified number of states recognizing each of the following languages. In all parts the alphabet is {0,1}. [4]

(a) The language {w | w ends with 001 with three states}

(b) The language 1*(001+)* with three states

(b) For languages A and B, let the shuffle of A and B be the language $\{w \mid w = a_1b_1 \cdots a_kb_k$, where $a_1 \cdots a_k \in A$ and $b_1 \cdots b_k \in B$, each $a_i, b_i \in \Sigma^*\}$. Show that the class of regular languages is closed under shuffle. [6]

5. (a) Give the formal idea for the following statements: [6]

(a) Prove that if we add a finite set of strings to a regular language, the result is a regular language.

(b) Prove that if we remove a finite set of strings from a regular language, the result is a regular language.

(c) Prove that if we add a finite set of strings to a nonregular language, the result is a nonregular language

(b) For the given expression id=(id1*id2)-(id3*id4) show the output of syntax, semantic, code optimization phase of the compiler? [4]