Roll No. ..... Total Pages : 5

301503

December, 2019

B.Tech. (IT/CE) 5th SEMESTER

Formal Languages Automata & Complier Design

(PCC-CS-502)

Time : 3 Hours] [Max. Marks : 75

Instructions :

1. *It is compulsory to answer all the questions (1.5 marks each) of Part-A in short.*
2. *Answer any four questions from Part-B in detail.*
3. *Different sub-parts of a question are to be attempted adjacent to each other.*

PART - A

1. (a) Represent the set $\{1^{2n+1} \mid n \geq 0\}$ by a regular expression and describe the regular expression $a(a + b)^*b$ in a set. (1.5)

(b) Find all strings of length 4 or less for the following regular expression :

(a*b + b*a). (1.5)

(c) Differentiate between DFA and NDFA. (1.5)

(d) Consider a grammar G whose productions are:
$S \rightarrow 0B \mid 1A$, $A \rightarrow 0 \mid 0S \mid 1AA$, $B \rightarrow 1 \mid 1S \mid 0BB$.
Find the left most and right most derivation for the string 00110101 in this grammar. (1.5)

(e) Justify that the some of the infinite sets are countable. (1.5)

(f) What do you mean by bootstrapping and define cross compiler. (1.5)

(g) Explain whether the following grammar is ambiguous or not?
$E \rightarrow E+E \mid E*E \mid (E) \mid a$. (1.5)

(h) What do you mean by Parser? What are the various types of Parsers available? (1.5)

(i) Differentiate between syntax and semantic. What do you mean by semantic actions? (1.5)

(j) What do you mean by optimization? Why this is an optional phase in the compiler design? (1.5)

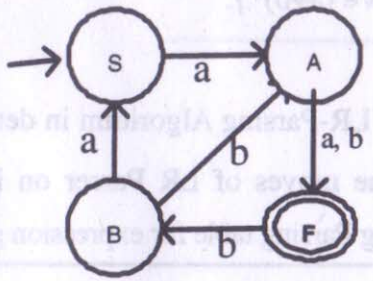
PART - B

2. (a) Prove that $L = \{0^n1^n \mid n \text{ is an integer}\}$ is not regular. (5)

(b) Design a DFA for the following regular expression :
$P = (11)^*(00)^*$
Afterwards, convert that DFA in a way that it would accept reverse of given regular expression (P^R). (10)

3. (a) Briefly discuss the functions of all the phases of a compiler. (5)

(b) Find the regular expression corresponding to given DFA in the Figure using Arden's Theorem. (10)



4. Construct the predictive parsing table for the following grammar G :

$E \rightarrow T\ E'$

$E' \rightarrow +\ T\ E'$

$E' \rightarrow \wedge$

$T' \rightarrow F\ T'$

$T'' \rightarrow *\ F\ T'$

$T'' \rightarrow \wedge$

$F \rightarrow (E)$

$F \rightarrow id$ (15)

5. (a) Design a Turing Machine for deciding the language $M = \{w\#w \mid w \in (a+b)^*\}$. (7.5)

(b) Design a PDA which will recognize the elements of following set:
$(ww^r \mid w \in (a+b)^*\}$. (7.5)

6. (a) Explain LR-Parsing Algorithm in detail. (7.5)

(b) Show the moves of LR Parser on id*id+id for the following Parsing table for expression grammar : (7.5)

STATE	ACTION						GOTO		
	Id	+	*	(	)	$	E	T	F
0	S5			S4			1	2	3
1		S6				Acc			
2		R2	S7		R2	R2			
3		R4	R4		R4	R4			
4	S5			S4			8	2	3
5		R6	R6		R6	R6			
6	S5			S4				9	3
7	S5			S4					10
8		S6			S11				
9		R1	S7		R1	R1			
10		R3	R3		R3	R3			
11		R5	R5		R5	R5			

7. Write short notes on :

(a) Symbol Table. (5)

(b) Intermediate Code Generation. (5)

(c) Machine Code Generation and Optimization. (5)