

Reg.No

--	--	--	--	--	--	--	--	--	--	--	--



SNS College of Technology, Coimbatore-35.

(Autonomous)

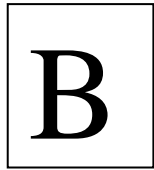
B.E/B.Tech- Internal Assessment -II

Academic Year 2023-2024 (ODD)

Fifth Semester

Computer Science and Engineering

19CSB301 – Automata Theory and Compiler Design

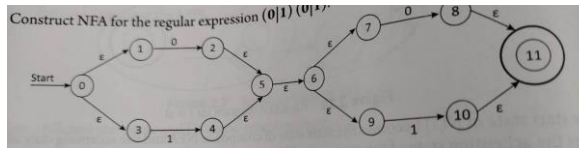


Time: 1.5 Hours

Maximum Marks: 50

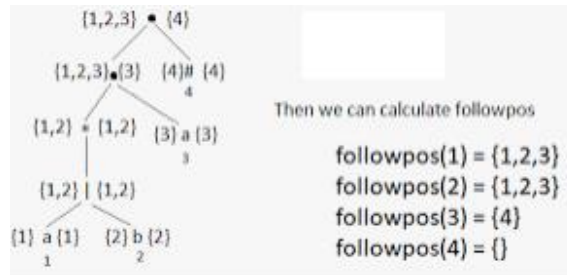
Part-A (5 x 2 =10 Marks)

1. Draw the ϵ -NFA for the regular expression $(0^*|1^*)$



CO Blooms
CO2 Rem

2. Construct the Syntax tree for $(a|b)^*$ and find the first pos and last pos of operands and operators



CO2 App

3. Eliminate immediate left recursion for the following Grammar

$E \rightarrow E+T$

$T \rightarrow T * F$

$F \rightarrow (E) | id$

CO3 App

Eliminating the immediate left recursion from the productions for E and then for T, we obtain

$E \rightarrow TE'$

$E' \rightarrow + TE' | \epsilon$

$T \rightarrow FT'$

$T' \rightarrow * FT' | \epsilon$

$F \rightarrow (E) | id.$

4. Perform left factoring for the grammar.

$A \rightarrow qB | qC$

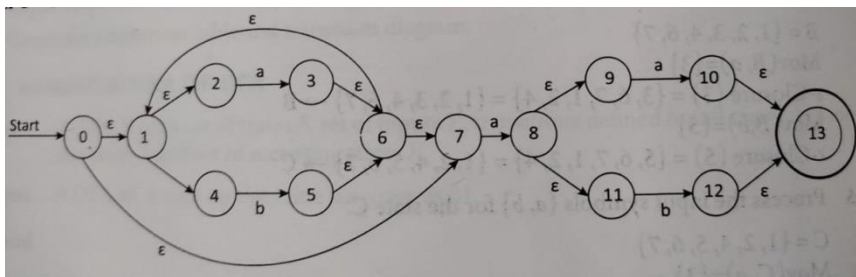
CO3 APP

5. Write the grammar (CFG) for the regular expression $a^*b^*c^*$

CO3 Rem

Part-B (2x13+14=40 Marks)

6. a. Construct the ϵ -NFA to DFA for the given regular expression $(a|b)^*a(a|b)$ using Thompsons & Subset Construction 13 CO2 A pp



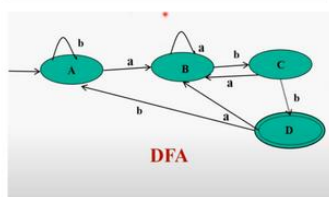
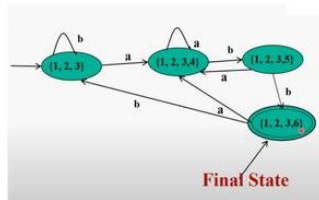
States	Input	
	a	b
A	B	C
B	D	E
C	B	C
D	D	E
E	B	C

(or)

b. Convert the Regular Expression $(a|b)^*abb$ to DFA (Direct Method) 13 CO2 A pp

DFA Construction

	Node	followpos
a	1	1,2,3
b	2	1,2,3
a	3	4
b	4	5
b	5	6
#	6	-



7. a. Construct the predictive parsing table for the following grammar and hence check whether the string (a,a) is accepted or not. 13 CO3 A na

$S \rightarrow (L)|a$

$L \rightarrow L,S|S$

Remove left recursion $L \rightarrow S L'$
 $L' \rightarrow , S L' / \epsilon$
 The grammar
 $S \rightarrow (L)$
 $S \rightarrow a$
 $L \rightarrow S L'$
 $L' \rightarrow , S L' / \epsilon$

Find FIRST for all the non-terminals.
 Non-terminals = $\{S, L\}$
 Terminals = $\{ (, , a \}$
 $FIRST(S) = \{ (, a \}$
 $FIRST(L) = \{ FIRST(S) \} = \{ (, a \}$
 $FIRST(L') = \{ , , \epsilon \}$

Find FOLLOW for all the non-terminals.
 $FOLLOW(S) = \{ S \text{ followed by non-terminal } L', \$ \}$
 $= \{ FIRST(L') = \{ , , \epsilon \}, \text{ eliminate } \epsilon, \text{ find } FOLLOW(L) \}$
 $= \{ , , \$ \}$
 $FOLLOW(L) = \{ \}$
 $FOLLOW(L') = \{ L' \text{ does not followed by terminal and non-terminal} \}$
 $= \{ FOLLOW(L) \}$
 $= \{ \}$

	()	a	,	\$
S	$S \rightarrow (L)$		$S \rightarrow a$		
L	$L \rightarrow S L'$		$L \rightarrow S L'$		

(or)

- b. Construct the CLR parsing table for the following grammar. check whether the string (a) is accepted or not. 13 CO3 A pp

CFG $A \rightarrow (A)|a$

Step 1: Write the augmented Grammar

Step 2: Compute Closure function

Step 3: Find LR(0) items

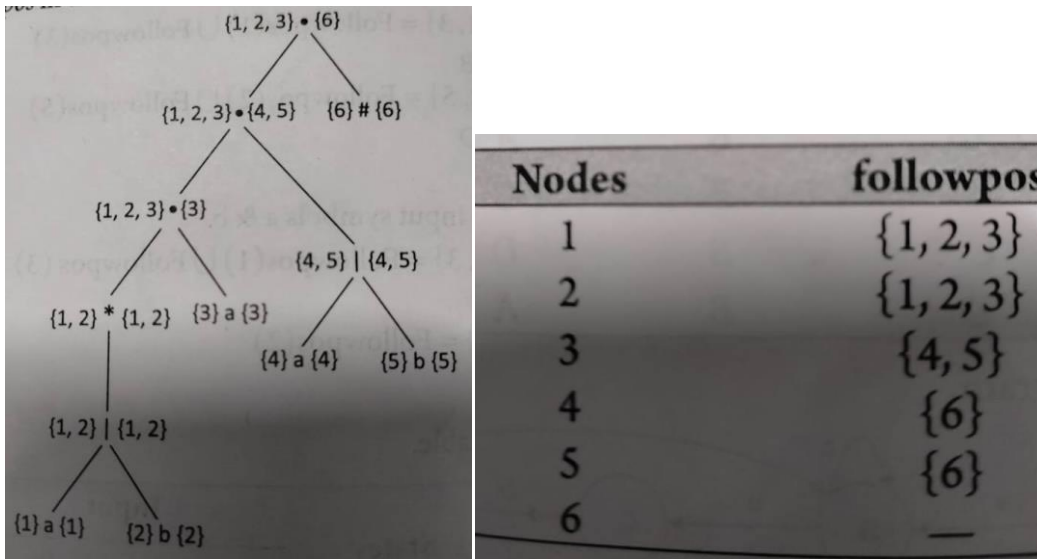
Step 4: Find Goto function

Step 5: CLR Parsing Table construction

Step 6: Input string Parsing

8. a. Construct the DFA from the given Regular Expression $(a/b)^*a(a/b)$ using Direct Method

14 CO2 A pp



(or)

b. Construct the SLR parsing table for the following grammar and check whether the string “adad” is accepted or not.

14 CO3 A na

$S \rightarrow CC$

$C \rightarrow aC$

$C \rightarrow d$

Step 1 :Find the augmented grammar G

Step 2:Find LR(0) Items

Step 3:Constuct Parsing Table

Step 4:Parse the string

States	Action			goto	
	c	d	\$	S	C
0	s3	s4		1	2
1			acc		
2	s3	s4			5
3	s3	s4			6
4	r3	r3	r3		
5	r1	r1	r1		
6	r2	r2	r2		

Stack	Input string	Action
0	cdcd \$	Shift 3
0c3	dcd \$	Shift 4
0c3d4	cd \$	Reduce r3 $C \rightarrow d$
0c3C6	cd \$	Reduce r2 $C \rightarrow cC$
0C2	cd \$	Shift 3
0C2c3	d \$	Shift 4
0C2c3d4	\$	Reduce r3 $C \rightarrow d$
0C2c3C6	\$	Reduce r1 $S \rightarrow CC$
0C2c3C6	\$	Reduce r2 $C \rightarrow cC$
0C2C5	\$	Reduce r1 $S \rightarrow CC$
0S1	\$	Accept

**Und-Understanding Rem-Remembering App-Applying
Ana-Analysis Cre-Creating Eva-Evaluating**

Prepared by

Verified by

HOD