JAMIA HAMDARD
Department of Computer Science & Engineering
School of Engineering Sciences & Technology
New Delhi-110062

Odd Semester Examination 2024 [BACKLOG]

Course Name: Formal Languages & Compiler Design **Paper Code: MCA 301**
Time: 3 Hours **Maximum Marks: 75**
Follow proper numbering for answering the questions.

Q. No.	Questions	Marks	CO	BL
	Section A ***All questions are compulsory.***	7×5=35		
1	**i.** Construct NFA for all strings ending with "a" ; alphabet is{a,b}. **ii.** Convert the regular expression abb(a | b)* to DFA. Alphabet is{a,b}	5	CO1	L2
2	Define First and Follow. Find out the first() and follow() for all the symbols of given grammar, G. Grammar: **S → E** **E → aTF b** **F →*aF | GH | ε** **T → (E) | ε** **G → g | ε** **H → h| ε**	5	CO4	L3
3	**i.** Find the number of tokens for the following C code: **int min(a,b)** **{ int a, b;** **return a < b? a : b;** **}** **ii.** Identify the type of grammar according to Chomsky classification Grammar: **A → aBb** **A → b** **B → a**	5	CO3	L3
4	Which phase(s) of compiler: **i.** Is/are considered frontend **ii.** Checks for type mismatch **iii.** Is/are independent of the underlying machine **iv.** Removes comment statements **v.** Produces a parse tree	5	CO2	L1

5	Explain various errors encountered in different phases of compiler.	5	CO3	L2
6	Eliminate left recursion from the following grammar: **S → Ar | At | w| x | z** **A → Aa | Ab | Sc | d | ε**	5	CO2	L3
7	Differentiate between Top Down parsing and Bottom Up Parsing.	5	CO2	L2
Attempt any Four questions.	**Section B**	4×10=40		
1	Construct the SDT to convert infix expression: a+b*c into postfix using the following grammar: **E→ E+T / T** **T → T*F / F** **F→ id**	10	CO3	L4
2	**i.** Define basic blocks & control flow graph. **ii.** Draw the control flow graph for the following 3 address statements: **1)** F=1 **2)** I=2 **3)** If I >X goto 8 **4)** T= F * I **5)** Z= I+1 **6)** I = Z **7)** goto 3 **8)** goto 1	10	CO6	L4
3	Represent **x=a-a / (b-c) + ((b-c) + (d^e^f))) using** following techniques: [^ represents Power operator] **i.** Syntax tree **ii.** DAG **iii.** Postfix **iv.** Quadruple	10	CO5	L3
4	Grammar G: **S→ Xa / bXc / Yc / bYa** **X→ d** **Y→ d** Construct SLR(1) parsing table & check whether the above grammar is SLR(1) or not.	10	CO4	L4
5	What is an activation record? Explain with diagram and example.	10	CO5	L2
6	Write a short note on: **i.** Ambiguous grammar **ii.** Constant Propagation & Dead Code Elimination	10	CO1, CO5	L2

BL-Bloom's Taxonomy Levels (1- Remembering, 2- Understanding, 3- Applying, 4- Analysing, 5- Evaluating, 6- Creating)