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

Even Semester Examination 2023

Course Name: Compiler Design **Paper Code: BTCSE 602**
Time: 3 Hours **Maximum Marks: 75**
Follow proper numbering for answering the questions.
Course Outcome (CO):
CO1. Evaluate the basic concepts of formal languages and their application to Compiler Design.
CO2. Develop a familiarity with fundamental principles of compiler design.
CO3. Demonstrate the process of translating a high-level language to executable code.
CO4. Analyze different parsing techniques and algorithms.
CO5. Generate intermediate code for statements in the high-level languages.
CO6. Apply techniques for code optimization.
CO7. Implement a complete compiler for a small programming language.

Q. No.	Questions	Marks	CO	BL
	Section A ***All questions are compulsory.***	7×5=35		
1	Grammar G: **S → E** **E → aTF b** **F →*aF | GH | ε** **T → (E) | ε** **G → g | ε** **H → h| ε** Evaluate First() and Follow() for all the symbols of above grammar, G.	5	CO4	L3
2	Find the number of tokens for the following C code: ***int main()*** ***{ int a = 1;*** ***while (a <= 10)*** ***{ printf ("%d",a);*** ***a + + ; } }***	5	CO3	L3
3	Considering alphabet to be {a,b} : **i.** Construct a NFA for all strings starting with "a" & ending with "b". **ii.** What language does the following regular expression generate? (a|b)*ab(a|b)*	5	CO2	L3

4	Optimize the given code using machine independent code optimization techniques and explain in brief the techniques used.
	int x= 5+7+c;
	z=a+y;
	printf("%d,%d",z,y);
	t1=x*z;
	t2=a+b;
	t3=p%t2;
	t4=x*z;
	int s= a+23;
	t5=a-z;
		5	CO6	L1, L2

5	Define left recursion in grammar. What is the need of eliminating left recursion from the grammar? Eliminate left recursion from the following grammar: $S \rightarrow Ar \mid At \mid As \mid w \mid x \mid z$ $A \rightarrow Aa \mid Ab \mid Sc \mid d \mid \varepsilon$	5	CO2	L3
6	Differentiate between top down and bottom up parsing technique. Grammar G: $X \rightarrow AB$ $B \rightarrow X \mid \varepsilon$ $A \rightarrow m$ Construct the LL(1) parsing table for the above grammar, G.	5	CO3	L2
7.	What is activation record? Explain.	5	CO1	L2
	Section B ***Attempt any Four questions.***	4×10=40		
1	What are the different phases of compiler? Explain in detail. Write down the output of each phase for the expression **a=b+c*50**.	10	CO3	L3
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 / (c^d)+((b-c)+((b-c)*(b^c^d)))** using following techniques: [^ represents Power operator] i.Directed Acyclic Graph ii. Three Address Code iii. Postfix iv. Triple	10	CO5	L3
4	Grammar G: **S→ Aa / bAc / dc / bda** **A→ d** Construct the LALR(1) parsing table & find out whether the above grammar is LALR(1) or not.	10	CO4	L4
5	i. Differentiate between synthesized and inherited attributes. ii. Consider the grammar with the following translation rules and E as the start symbol. **E -> E1 #T {E.value = E1.value * T.value}** **| T {E.value = T.value}** **T -> T1 & F {T.value = T1.value + F.value}** **|F {T.value= F.value}** **F -> num {F.value = num.value}** Compute E.value for the root of the parse tree for the expression: 2 # 3 & 5 # 6 &4.	10	CO4	L3
6	Write a short note on any two of the following: i. Semantic Analysis ii. LR(0) and LR(1) items iii. Inheritance & Polymorphism	10	CO1, CO3	L1, L2

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