U.S.N.										

B.M.S. College of Engineering, Bengaluru-560019

Autonomous Institute Affiliated to VTU

June 2025 Semester End Main Examinations

Programme: B.E. **Semester: V**
Branch: Computer Science & Engineering **Duration: 3 hrs.**
Course Code: 22CS5PCCPD **Max Marks: 100**
Course: Compiler Design

Instructions: 1. Answer any FIVE full questions, choosing one full question from each unit.
2. Missing data, if any, may be suitably assumed.

Important Note: Completing your answers, compulsorily draw diagonal cross lines on the remaining blank pages. Revealing of identification, appeal to evaluator will be treated as malpractice.

		UNIT - I	CO	PO	Marks
1	a)	Outline the various phases of a compiler with any suitable example	CO2	PO2	**10**
	b)	Write a lex program to tokenize the following line. What are the tokens that are identified? int a,b,sum=0;	CO1	PO1	**10**
		OR			
2	a)	Design a transition diagram for relational operators such as < , <=, > , >=,== ,!= . Emulate the same with a simple pseudocode.	CO2	PO2	**10**
	b)	Describe input buffer scheme used in lexical analysis with look ahead code with sentinels	CO1	PO1	**10**
		UNIT - II			
3	a)	Give the algorithm for finding FIRST and FOLLOW Set. Find FIRST and FOLLOW set of all variables given below S → ACB | CbB | Ba A→ da|BC B → g | ε C → h | ε	CO1	PO1	**12**
	b)	Verify if the following grammar is ambiguous? E → E+E | E-E | E*E | (E) | a | b If ambiguous convert the same to unambiguous grammar with following associativity and precedence ( ) having highest priority + having second highest priority and right associative - having third highest priority and right associative * having least priority and left associative	CO2	PO2	**08**
		OR			
4	a)	Construct LR(0) Item set , SLR Parsing table for the following grammar. Validate with appropriate reasoning if the given grammar is SLR or not S → L=R | R L → *R | id R→ L	CO1	PO1	**12**
	b)	Analyze if the following grammar is suitable for Predictive parsing. With appropriate justification	CO1	PO1	**8**

			E → 5+T | 3-T T→ V | V*V | V+V V→ a | b			
			UNIT - III			
	5	a)	Write SDD for a desk calculator having operations +, * for a top down parser. Show the dependency graph for the input 1+2*3.	*CO1*	*PO1*	**10**
		b)	Write the SDT for int [2][3]. Show the annotated parse tree.	*CO1*	*PO1*	**10**
			OR			
	6	a)	Write the grammar and design the syntax directed definition for a simple desk calculator using bottom up approach. Also outline the type of attribute in each production.	*CO1*	*PO1*	**7**
		b)	Write the SDD for a simple type declaration statement and design the annotated parse tree for the input string "float a,b,c".	*CO1*	*PO1*	**8**
		c)	Explain in detail about S-attributed and L-attributed definition with an example.	*CO1*	*PO1*	**5**
			UNIT - IV			

	7	a)	Analyze the following code snippet and give its equivalent three address code
			while(a<c and b>d)
			{
			if a=1
			then c=c+1
			else
			while (a <= d)
			a=a+b
			}
				CO2	*PO2*	**5**

		b)	Give DAG for the expression (a+b)*(a+b+c)	*CO2*	*PO2*	**3**
		c)	Outline with a simple diagram and with appropriate SDD for flow control statement such as if,if-else ,while	*CO1*	*PO1*	**12**
			OR			
	8	a)	What is meant by three address code? Construct DAG and equivalent three address code for ((x+y)-((x+y)*(x-y))) + ((x+y)*(x-y)) .	*CO2*	*PO2*	**10**
		b)	Give SDT for various Boolean expressions	*CO1*	*PO1*	**10**
			UNIT - V			
	9	a)	List issues in design of code generator	*CO1*	*PO1*	**10**
		b)	Give intermediate code, basic block and flow graph for the code snippet given below *for i from 1 to 10 do* *for j from 1 to 10 do* *a[i,j]=0.0;* *for i from 1 to 10 do* *a[i,i] = 1.0;*	*CO2*	*PO2*	**10**

			OR			
	10	a)	Convert the following 3 address code to assembly target instruction i)*p=y ii)a[j]=c	*CO2*	*PO2*	**5**
		b)	Analyze the following three address code and generate the equivalent optimal machine code. Also show the address descriptor and register descriptor values for each instruction. Assume a,b,c,d are live on exit from this block and T and U are temporary variables in this block. Also consider only two register R1 and R2 available in this block. T=a*b U=a-c a=d	*CO1*	*PO1*	**5**
		c)	Consider the given program instruction. Write the cost of each instruction. LD R0,R1 LD R0,M LD R1,*100(R2) LD R1,*R2 MUL R0,R0,8 LD R1,a(R0) ST b,R1 LD R0,p ST x,R1 BLTZ *R3,R0	*CO2*	*PO2*	**10**

****