B.E. COMPUTER SCIENCE AND ENGINEERING
THIRD YEAR SECOND SEMESTER EXAM 2025
Subject: ***Compiler Design***

Time : Three hours **Full Marks: 100**

Group-1 [CO1] 20 marks	Answer Question 1 and any one from the rest: 1. (a) What is the purpose of a compiler? (b) Discuss the different phases of the back-end of a compiler highlighting the significance of each phase. (c) What are the categories of formal languages? Define a formal grammar. 2+4+4=10 2. Write a regular expression for a signed quaternary number (base-4) (there should not be any leading zero in the number).Construct a DFA for itdirectly from the regular expression using firstpos, lastpos and followpos. 2+8=10 3. (a) Write a regular expression that can recognize the positive integer constants in C. Some examples of acceptable integer constants are given below: 45 O123 (octal) 30u (unsigned) 50l (long) (b) Construct an NFA for the above regular expression using McNaughton–Yamada–Thompson algorithm. Show every steps of the construction. 2+8=10
Group-2 [CO2] 40 marks	Answer Question 4 any one question from the rest: 4. (a) Consider the following context-free grammar: S →S + S | S S | (S) | S * | a (i) Give leftmost and rightmost derivations for the string (a + a) * a (ii) Draw a parse tree for the string. (iii) Is the grammar ambiguous or unambiguous? Justify your answer. (b) While constructing a parsing table for a top-down parser, why do you need to construct the FIRST and FOLLOW sets? Explain your answer with examples. (c) How do you recognize that a grammar is LL(1)? Give an example. (d) Two popular bottom-up parsing techniques are SLR and Canonical LR. What are the differences between these parsing techniques? Which one is more powerful and why? (3+2+3)+4+4+4=20 5. (a) Consider the grammar: *S --> AB* *S --> BA* *A -->aaB* *A --> a* *B -->bbA* *B --> a* (Terminals = *{a, b}*, Non-terminals = *{S, A, B}*, Start Symbol = *S*) Construct LR(0) item set for the above grammar. (b) Construct the SLR parsing table for the above grammar. (c) Show the actions of the parser for the input string: (bbaaa). (d) What is a viable prefix? Show at least 2 viable prefixes for the above parsing. 6+8+4+2=20 6. Consider the grammar: *S →SS* + | *S S* * | *a* (*S*is the start symbol;+, * and *a*are terminals) (a) What type of conflicts arise in LR parsing and why? (b) Construct the canonical LR set of items for the above grammar. (b) Construct the LALR set of items for the above grammar. (c) Discuss how the lookahead is used in LR(1) parsing (Definition is not required. Explain with an example). 4+6+6+4=20

Ref. No.:**Ex/CSE/PC/B/T/323/2025**

Group-3
[CO3]
20 marks
	Answer any twoquestions:
	7. (a) What is the importance of semantic analysis phase in compilers? What do you mean by *Syntax Directed Definition*?
	What are the different types of *Syntax Directed Definitions?*
	(b) Consider the following grammar for expressions involving operator '+' and integer or floating point operands. Floating point numbers are distinguished by having a decimal point.
	E → E + T | T
	T →num.num | num
	Give an SDD to determine the type of each term T and expression E.
	5+5=10
	8. (a) What is the use of a symbol table? In which phases of compiler a symbol table is used?
	(b) What are the operations performed on a symbol table when it is implemented as multiple hash tables?
	(c) What is a three-address code? Write a three address code for the following expression:
	for(i = 1; i<=10; i++)
	{
	a[i] = x * 5;
	}
	2+3+5=10
	9. (a) Write the semantic rules to generate three address codes for an assignment statement for arithmetic expressions which will use binary operators '+' and '*'.
	(b)Show how the code for the statement 'x=a+b*c+d' can be generated with the semantic rules.
	4+6=10

Group-4
[CO 4&5]
20 marks
	Answer any one questions:
	10. (a) What is a basic block? Write an algorithm to identify a basic block.
	(b) Consider the following three-address code:
	p = q + r
	s = p + q
	u = s * v
	if (u > 100)
	v = r + u
	else
	q = s * u
	q = v + r
	Construct the flow graph of the above piece of code.
	(c) Find the liveness and next use for each variable at each statement and at the entry of each basic block.
	(d) Construct the Register Interference Graph.
	(e) What is the difference between Useless code elimination and Unreachable code elimination? Give examples of both.
	4+3+6+3+4=20
	11. Consider the following basic block:
	A := 5;
	F := A + 1;
	E := F + 5;
	B := F * A;
	PRINT B + E + A;
	D := E + 5;
	PRINT E;
	C := D + B;
	PRINT E + C;
	(a) Construct the register interference graph for the block.
	(b) How many colors are necessary to color the graph optimally without register spills? Show such an optimal coloring.
	(c) With an example of each of the following code optimization techniques, explain their advantages:
	(i)Constant propagation, (ii) Loop invariant code motion, (iii) Loop fission, (iv) Loop fusion, (v) Loop unrolling and jam
	6+4+10=20