VIT
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

REG.NO.:

SLOT: E2+TE2

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING
CONTINUOUS ASSESSMENT TEST - II
FALL SEMESTER 2025-2026

Programme Name & Branch	: B. Tech CSE
Course Code and Course Name	: BCSE307L – Compiler Design
Faculty Name(s)	: Prof. Sathya K
Class Number(s)	: VL2025260104779
Date of Examination	: 09.10.2025 9.30AM to 11.00AM
Exam Duration	: 90 minutes

Maximum Marks: 50

General instruction(s): Answer All Questions

Course Outcomes:

CO2: Develop language specifications using context free grammars (CFG).
CO3: Apply the ideas, the techniques, and the knowledge acquired for the purpose of developing software systems
CO4: Constructing symbol tables and generating intermediate code.
CO5: Obtain insights on compiler optimization and code generation

Q. No	Question	M	CO	BL
1.	Consider the language: $L(G) = \{ww^R \mid w \in \{a, b\}^*\}$ with the grammar: $S \rightarrow aSa \mid bSb \mid a \mid b$ a) Construct the canonical LR (1) item sets for this grammar. **(5 Marks)** b) Suppose we modify the grammar to allow empty string (ε): **(5 Marks)** $S \rightarrow aSa \mid bSb \mid a \mid b \mid \epsilon$. Explain how the CLR item sets and conflicts change.	10	CO2	2
2.	Consider the grammar $E \rightarrow E + T \mid E - T \mid T$ $T \rightarrow T * F \mid T / F \mid F$ $F \rightarrow (E) \mid id \mid digit \mid id [E1] \mid id [E1][E2]$ Define a Syntax-Directed Translation (SDT) and Generate postfix notation as synthesized attribute. Also, annotate the parse tree for the expression: B[1][3] := (A[2] + 5) * 2	10	CO3	3

3.	a) Given the grammar
	$E \rightarrow E + T \mid E - T \mid T$
	$T \rightarrow T * F \mid T / F \mid F$
	$F \rightarrow (E) \mid id \mid num$
	Construct the syntax tree for an expression a * 3 * (2 + b) * 4 using the SDD. **(5 Marks)**
	b) Generate the three-address code for the following C program **(5 Marks)**
	switch (x) {
	case 1:
	if (y > 10)
		10	CO4	3

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING
CONTINUOUS ASSESSMENT TEST - II
FALL SEMESTER 2025-2026

	z := y + 5;
	else
	z := y - 5;
	break;
	case 2:
	z := y * 2;
	break;
	default:
	z := 0;			

4.	Given the input code:
	while a < 10 do
	if b == 0 then
	c := c + 1
	else
	while d < 5 do
	d := d + 1
	Illustrate the steps of generating the intermediate code with backpatching.
		10	CO4	3

5.	Consider the following C code.
	int main () {
	int x = 5, y = 10;
	int result = 0;
	int unusedVar = x + 0;
	
	if (x > 0 && y > 0) {
	result = x * 2 + y * 2;
	}
	
	if (x < 0 || y < 0) {
	result += 100;
	}
	
	printf("Result = %d\n", result);
	return 0;
	}
	Highlights the Principal Sources of Optimization and rewrite the optimized code.
		10	CO5	2

****