	ITER, SIKSHA 'O' ANUSANDHAN (Deemed to be University)		Assignment
Branch	**CSE/CSIT**	Programme	**B.Tech**
Course Name	**Compilers : Principles, Techniques and Tools**	Semester	**6th**
Course Code	**CSE 3739**	Academic Year	**2024-25**

ASSIGNMENT - II

Submission due date: 17/05/2025

Learning Level (LL)			
	L1: Remembering	**L3:** Applying	**L5**: Evaluating
	L2: Understanding	**L4**: Analysing	**L6**: Creating

Q's	Questions	COs	LL
1	Consider the grammar **S→aAd|bBd|aBe|bAe** **A→c** **B→c** i) Create the LR(1) canonical sets of items. ii) Construct the CLR(1) parsing table for the grammar. iii) Construct the LALR(1) parsing table for the grammar. Is the grammar LALR(1) ?Justify.	**CO3**	**L6**
2	Consider the following grammar and their syntax Directed Translation (SDT) rules. **Productions** **Sementic Rules** S → S * A {S.val = S.val × A.val} S → A {S.val = A.val} A → A + B {A.val = A.val - B.val} B → (S) {B.val = 2} A → B {A.val = B.val} B → id {B.val = id.lexval} Construct the annotated parse tree and evaluate the value of the expression 4*6+3*7	**CO4**	**L6**
3	A shift reduce parser carries out the actions specified within braces immediately after reducing with the corresponding rule of grammar. **S→aaW {print '1' }** **S→b {print '2' }** **W→Sc {print '3' }** Construct the annotated parse tree and find the translation of **aaaabcc** using the syntax directed translation scheme described by the above rules?	**CO4**	**L6**
4	Consider the expression **a + a * (b-c) + (b-c) * d** Generate the **three-address code (TAC)** for the given expression.	**CO5**	**L3,L4**
5	Represent the three-address code that is generated in Q.4, using the following intermediate code formats: i) Quadruples ii) Triples iii) Indirect Triples	**CO5**	**L4,L6**

Course Outcomes	By the end of the course, through lectures, readings, home works, assignments, and exams, students will be able to:	
	CO1	Understand the overview of programming languages, language processors and the structure of a compiler
	CO2	Acquire knowledge in theory of computation and their role in designing different types of tokens generated by lexical analyzer
	CO3	Understand the role of Parser(s) (LL, SLR, CLR and LALR) and its types i.e. Top-down and Bottom-up parsers.
	CO4	Apply and evaluate syntax directed translation schemes, synthesized

		attributes, inherited attributes, and different techniques for symbol table organization
	CO5	Analyze the generation of various intermediate codes and the process of their optimization.
	CO6	Understand the target machine's run time environment, and its instruction set for code generation and techniques used for code optimization