C. U. SHAH UNIVERSITY- Wadhwan City

FACULTY OF TECHNOLOGY AND ENGINEERING
DEPARTMENT OF COMPUTER ENGINEERING
B.TECH. SEMESTER: - VII

Subject Name: – Translator Design
Subject Code: - 4TE07TDE1

Teaching & Evaluation Scheme: -

Subject Code	Subject Name	Teaching Scheme (Hours)				Credits	Evaluation Scheme							
		Th	Tu	Pr	Total		Theory				Practical			Total
							Session Exam		University Exam		Internal		University	
							Marks	Hours	Marks	Hours	Pr/ Viva	TW	Pr	
4TE07TDE1	Translator Design	03	00	02	05	04	30	1.5	70	3.0	---	20	30	150

Objective: To introduce students with the techniques used in designing and writing compilers.

Prerequisite: Basic knowledge of data structure and algorithms and theory of computation.

Sr. No.	Course Content	Total Hours
1.	**Introduction to Translators:** Language Processing Activities, Fundamentals of language processing, Fundamentals of language specification, Language processor Development tools, Data Structure for language processing, editors, Debuggers.	04
2.	**Macro and Macro Processors:** Macro Definition and call, Macro Expansion, Nested Macro calls, Advanced macro facilities, Design of Macro Processors, Design of Macro Assembler.	03
3.	**Introduction to Compiler:** The phases of a Compiler, Cousins of the Compiler, grouping of Phases, Front-end and Back-end of compiler, pass structure, Error Handling.	04
4.	**Lexical Analysis:** Role of Lexical Analyzer, Input Buffering, Specification and recognition of Tokens, A Language for Specifying Lexical Analyzers, Finite Automata, From a Regular Expression to an NFA, Design of a Lexical Analyzer Generator, Optimization of DFA, Hard Coding and Automatic Generation Lexical Analyzers.	06
5.	**Syntax and Semantic Analysis:** Top-Down Parsing Methods, Brute-Force approach, Recursive descent and predictive parsers, Bottom-Up Parsing, shift –reduce parsing, Operator-Precedence Parsing, LR Parsers, Using Ambiguous Grammars, Parser Generators, Error handling in parser, Automatic Generation of Parsers, Syntax-Directed Definitions and translation schemes Construction of Syntax Trees, Bottom-Up Evaluation of S-Attributed Definitions, L-Attributed Definitions, Type Checking, type systems, type expressions, type conversion and overloading.	10
6.	**Intermediate Code Generation:** Intermediate languages, syntax trees, postfix code, triples and indirect triples, Syntax Directed Translation Mechanisms And Attributed Mechanisms And Attributed Definition.	04
7.	**Memory Management:** Source Language Issues, Storage Organization, Storage organization and Allocation Strategies, and Access to Non local Names, Parameter	04

	Passing, Symbol Tables, Language Facilities for Dynamic Storage Allocation, Dynamic Storage Allocation Techniques.	
8.	**Code Generation:** Issues in the Design of a Code Generator, The Target Machine, Run-Time Storage Management, Basic Blocks and Flow Graphs, Next-Use Information, A Simple Code Generator, Register Allocation and Assignment, The DAG Representation of Basic Blocks, Peephole Optimization, Generating Code from DAGs	**05**
9.	**Code Optimization:** Sources of code optimization, optimization of basic blocks, data flow analysis, code improving transformations	**03**
10.	**Assembler, Linker and Loader:** Elements of Assembly language Programming, A simple Assembly scheme, One pass and two pass assembler, Relocation of linking concepts, design of linker, self relocating programs, Loaders, A suitable case study for linker	**05**
	Total	**48**

Outcomes:

At the end of the course, students will be able to design and develop compiler for any kind of languages as well as students would be able to generate their own parser. Student will also be able to optimize code and to apply various error recovery mechanisms.

Books Recommended:

1. Compiler Tools Techniques - A.V.Aho, Ravi Sethi, J.D.Ullman, Addison Wesley
2. The Theory and Practice of Compiler Writing - Trembley J.P. And Sorenson P.G. Mcgraw-Hill
3. System Programming and Operating Systems, 2nd Edition by D. M. Dhamdhere, Tata Mcgraw-Hill (2009)
4. Modern Compiler Design - Dick Grune, Henri E. Bal, Jacob, Langendoen, WILEY India
5. Compiler Construction - Waite W.N. And Goos G., Springer Verlag
6. Compiler Construction-Principles and Practices - D.M.Dhamdhere, Mcmillian
7. Principles of Compiler Design, V. Raghavan, McGrawHill
8. Systems Programming by John J. Donovan