INSTITUTE	**FACULTY OF TECHNOLOGY**
PROGRAM	**BACHELOR OF TECHNOLOGY (COMPUTER ENGINEERING)**
SEMESTER	**7**
COURSE TITLE	**COMPILER DESIGN**
COURSE CODE	**01CE0714**
COURSE CREDITS	**4**

Objective:

1 The objective of this course is to provide students with a comprehensive understanding of the fundamental methodologies, theory, and tools that enable the practise and act of compiler creation.

Course Outcomes: After completion of this course, student will be able to:

1 Explain the process of program translation and the various phases and tools involved in compiler construction.

2 Apply lexical analysis techniques to convert regular expressions into finite automata and perform lexical token recognition.

3 Analyze and differentiate parsing techniques for context-free grammars, and construct parsers using top-down and bottom-up methods

4 Design intermediate code representations and implement syntax-directed translation schemes with proper memory management strategies

5 Apply code optimization techniques and generate efficient machine-level code using flow graphs and code generation strategies.

Pre-requisite of course:NA

Teaching and Examination Scheme

Theory Hours	Tutorial Hours	Practical Hours	ESE	IA	CSE	Viva	Term Work
3	0	2	50	30	20	25	25

Contents : Unit	Topics	Contact Hours
1	**Introduction to Compilers:** Outline of Process Of translation, Phases of Compiler, Cousins of Compiler, Types of compiler, Symbol Table, Compiler Constructions tools, Overview of LEX and YACC	5

MR. HARDIK AJAYBHAI DOSHI
Digitally signed by (Name of HOD)

DR. RAJENDRASINH BAHADURSINH JADEJA
Digitally signed by (Name of Dean/ Principal)

Contents : Unit	Topics	Contact Hours
2	**Lexical Analysis** Role of Lexical analyzer, Input buffering techniques, Specification and recognition of tokens, Conversion of Regular expressions to finite automata using Thompson's method, Conversion of NDFA to DFA, Minimization of DFA, Conversion of Regular Expression to DFA using Syntax Tree Method	8
3	**Parsing** Role of Parser and basics of Context Free Grammar, Left recursion, Left Factoring,, ambiguous grammar, ambiguous grammar, Top down and Bottom up Parsing, Recursive Descent Parser, Predictive Parser, LL(1) Parsing, Operator Precedence Parser, LR Parsing - LR (0), Simple LR Parsing, Canonical LR Parsing, Look ahead LR Parsing	12
4	**Intermediate Code Generation and Memory Management** Type of Intermediate forms, Directed Acyclic Graph, Quadruple,, Triples, Indirect triples,, Syntax Directed Definition - Synthesized Attributes, Inherited Attributes, Syntax Directed Translation, Activation record and parameter passing method, Activation record and parameter passing method	7
5	**Code Optimization and Generation** Introduction to optimization and Optimization Techniques, Peephole optimization, Overview of Global Data Flow analysis, Basic blocks and Flow graphs, DAG Representation of Basic Block, , Issues in the design of code generation, Register allocation and assignment	10
Total Hours		**42**

Suggested List of Experiments:

Contents : Unit	Topics	Contact Hours
1	**Practical 1** Write a C Program to remove Left Recursion from grammar	2
2	**Practical 2** Write a C Program to remove Left Factoring from grammar.	2
3	**Practical 3** Write a C program to implement finite automata and string validation.	2
4	**Practical 4** Prepare report for Lex and install Lex on Linux/Windows	2
5	**Practical 5** (a) WALEx Program to count words, characters, lines, Vowels and consonants from given input, (b) WALEx Program to generate string which is ending with zeros.	2
6	**Practical 6** (a) WALex Program to generate Histogram of words, (b) WALex Program to remove single or multi line comments from C program.	2

MR. HARDIK AJAYBHAI DOSHI
Digitally signed by (Name of HOD)

DR. RAJENDRASINH BAHADURSINH JADEJA
Digitally signed by (Name of Dean/ Principal)

Suggested List of Experiments:

Contents : Unit	Topics	Contact Hours
7	**Practical 7** WALex Program to check weather given statement is compound or simple.	2
8	**Practical 8** WALex Program to extract HTML tags from .html file.	2
9	**Practical 9** 9 Write a C Program to compute FIRST Set of the given grammar	2
10	**Practical 10** Write a C Program to compute FOLLOW Set of the given grammar	2
11	**Practical 11** Write a C Program to implement Operator precedence parser	2
12	**Practical 12** Write a C Program for constructing LL (1) parsing	2
13	**Practical 13** Write a C program to implement SLR parsing.	2
14	**Practical 14** Prepare a report on YACC and generate Calculator Program using YACC.	2
Total Hours		**28**

Textbook :

1 "Compilers: Principles, Techniques, and Tools", Alfred V. Aho, Monica S. Lam, Ravi Sethi, and Jeffrey D. Ullman, Pearson education (Singapore), 2003

References:

1 "Engineering a Compiler", "Engineering a Compiler", Keith Cooper and Linda Torczon, Elsevier Science, 2011

2 Systerm Programming, Systerm Programming, D. M. Dhamdhere, Mc Graw Hill Publication, 1999

3 Langendoen: Modern Compiler Design,, Langendoen: Modern Compiler Design,, Dick Grune, Henri E. Bal, Jacob, Wiley India Publication, 2012

Suggested Theory Distribution:

The suggested theory distribution as per Bloom's taxonomy is as follows. This distribution serves as guidelines for teachers and students to achieve effective teaching-learning process

Distribution of Theory for course delivery					
Remember / Knowledge	**Understand**	**Apply**	**Analyze**	**Evaluate**	**Higher order Thinking / Creative**
10.00	15.00	35.00	20.00	15.00	5.00

MR. HARDIK AJAYBHAI DOSHI
Digitally signed by (Name of HOD)

DR. RAJENDRASINH BAHADURSINH JADEJA
Digitally signed by (Name of Dean/ Principal)

Instructional Method:

1 The course delivery method will depend upon the requirement of content and need of students. The teacher in addition to conventional teaching method by black board, may also use any of tools such as demonstration, role play, Quiz, brainstorming, MOOCs etc.

2 The internal evaluation will be done on the basis of continuous evaluation of students in the laboratory and class-room.

3 Practical examination will be conducted at the end of semester for evaluation of performance of students in laboratory.

4 Students will use supplementary resources such as online videos, NPTEL videos, e-courses, Virtual Laboratory.

Supplementary Resources:

1 https://nptel.ac.in/courses/106108052

2 http://web.stanford.edu/class/cs143/

3 https://courses.cs.washington.edu/courses/csep501/11au/

MR. HARDIK AJAYBHAI DOSHI
Digitally signed by (Name of HOD)

DR. RAJENDRASINH BAHADURSINH JADEJA
Digitally signed by (Name of Dean/ Principal)