
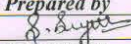
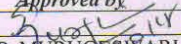
 SRI VIDYA COLLEGE OF ENGINEERING & TECHNOLOGY COURSE PLAN (THEORY)												
ACADEMIC YEAR: 2018-2019												
Subject Code	CS6660	L	P	T	C							
Subject Title	COMPILER DESIGN	3	0	0	3							
Year / Dept / Sem	III/CSE/VI	Regulation Year		2013								
Faculty Name / Desg / Dept	S.SUJITHA /AP/CSE											
Course Prerequisite	KNOW ABOUT THE THEORY OF COMPUTATION.											
Attach the copy of syllabus												
Course Objectives (CO)	CO1: Learn the design principles of a Compiler. CO2: Learn the various parsing techniques and different levels of translation. CO3: Learn the concepts of code generation, Run time storage management & different approaches of Compiler development. CO4: Learn how to optimize and effectively generate machine codes.											
Expected Course Outcomes (ECO)	At the end of the course, the students should be able to: ECO1: Define the phases and types of various translators along with several representations, specification and construction formats. ECO2: Apply parser generator to design syntax tree. ECO3: To learn the concept of regular expression. ECO4: To Define process of Intermediate Code generation. ECO5: Learn the concepts of code generation, Run time storage management. ECO6: To explain the concepts of Code Optimization and about various code improving transformations.											
Mapping of CO & PO (Specify the PO's)												
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	B,2	-	C,3		-	-	-	-	-	-	-	-
CO2	-	-	-	B,2	A,2	-	G,2	-	G,1	-	-	-
CO3	-	B,2	E,2	C,1	-	-	-	-	-	-	-	-
CO4	-	-	-	-	-	-	-	-	-	-	-	-
CO5	-	-	-	-	-	-	-	-	-	-	-	H,1
CO6	-	-	-	-	-	F,1	-	-	-	-	-	-
Bridging the Curriculum Gap (Additional Topics beyond syllabus/Seminars/Assignments)	BCG1: Assignments related to LEX code generation BCG2: To Master in compiler techniques LEX and YACC programming. BCG3: The main reason is that a tree expresses the 'component' relationship commonly used to describe semantics											
Related Website URLs	W1: www.tutorialspoint.com/compiler_design/ W2: www.tutorialspoint.com/compiler_design/compiler_design_tutorial.pdf W3: www.tutorialspoint.com/compiler_design/compiler_design_pdf_version W4: www.diku.dk/~torbenm/Basics/basics_lulu2.pdf											
Related Video Course Materials (min. 3 no.s)	V1: www.youtube.com/watch?v=e73sb5pyriQ V2: www.youtube.com/watch?v=Ti8mGLqwLBw V3: https://www.youtube.com/playlist?list=PL3690D679B876DE6A V4: www.youtube.com/watch?v=Qkwj65l_96I											

S.No	Topic Name	Book Page no ^{T1}	Mode of teaching	No of hrs	Cumulative hrs
UNIT I - INTRODUCTION TO COMPILERS					
1	Introduction about compiler	-	BB	1	1
2	Translators-Compilation and Interpretation-Language processors	1	BB	1	2
3	The Phases of Compiler	4-11	BB	1	3
4	Errors Encountered in Different Phases	8	BB	1	4
5	Grouping of Phases Compiler construction tools	11,12	BB	1	5
6	Programming Language basics	25-35	BB	1	6
7	Review of Unit 1	-	-	1	7
UNIT II - LEXICAL ANALYSIS					
1	Need and Role of Lexical Analyzer	109-113	BB	1	8
2	Lexical Errors	113	BB	1	9
3	Expressing Tokens by Regular Expressions	116-123	BB	1	10
4	Converting Regular Expression to DFA	152-156	BB	1	11
5	Minimization of DFA	180-184	BB	1	12
6	Language for Specifying Lexical Analyzers	117-119	BB	1	13
7	LEX	140-145	BB	1	14
8	Design of Lexical Analyzer for a sample Language.	166-171	BB	1	15
9	Review of Unit 2	-	-	1	16
UNIT III - SYNTAX ANALYSIS					
1	Need and Role of the Parser	192	BB	1	17
2	Context Free Grammars	197-205	BB	1	18
3	Top Down Parsing	217	BB	1	19
4	General Strategies- Recursive Descent Parser Predictive Parser	219,226	BB	1	20
5	LL(1) Parser- Shift Reduce Parser	222-226, 236-238	BB	1	21

6	LR Parser - LR (0)Item	241-242	BB	1	22
7	Construction of SLR Parsing Table	252-256	BB	1	23
8	Introduction to LALR Parser	266-270	BB	1	24
9	Error Handling and Recovery in Syntax Analyzer	283	BB	1	25
10	YACC- Design of a syntax Analyzer for a Sample Language.	287-294	BB	1	26
11	Review of Unit 3	-	-	1	27
UNIT IV - SYNTAX DIRECTED TRANSLATION & RUN TIME ENVIRONMENT					
1	Syntax directed Definitions- Construction of Syntax Tree	303,318	BB	1	28
2	Bottom-up Evaluation of S-Attribute Definitions	312	BB	1	29
3	Design of predictive translator	248	BB	1	30
4	Type Systems	278-281	BB	1	31
5	Specification of a simple type checker	387	BB	1	32
6	Equivalence of Type Expressions	371-372	BB	1	33
7	Type Conversions	388-390	BB	1	34
8	RUN-TIME ENVIRONMENT: Source Language Issues	297-303	BB	1	35
9	Storage Organization	427-429	BB	1	36
10	Storage Allocation	430-438	BB	1	37
11	Parameter Passing	323-327	BB	1	38
12	Symbol Tables	49-51	BB	1	39
13	Dynamic Storage Allocation	308-313	BB	1	40
14	Review of Unit 4	-	-	1	41
UNIT V - CODE OPTIMIZATION AND CODE GENERATION					
1	Principal Sources of Optimization	584-592	BB	2	43
2	DAG	533	BB	1	44
3	Optimization of Basic Blocks	533-539	BB	1	45
4	Global Data Flow Analysis	597-599	BB	1	46
5	Efficient Data Flow Algorithms	618	BB	1	47
6	Issues in Design of a Code Generator	506-512	BB	1	48

7	A Simple Code Generator Algorithm	542-547	BB	1	49
8	Review of Unit 5	-	-	1	50

	<i>Prepared by</i>	<i>Approved by</i>
Signature		
Name	S.SUJITHA	Dr. P. MURUGESWARI
Designation	Assistant Professor / CSE	Professor & HOD (CSE/IT)
Signed date	23/10/18	

LEGEND:

METHODOLOGY TO MAP OBJECTIVE WITH OUTCOME

Course outcomes are achieved through

- a. Suitable Analogies.
- b. Class room teaching.
- c. Assignments.
- d. Tutorials.
- e. Weekly, monthly and model exams.
- f. Brain storming.
- g. Group discussion and role play.
- h. Seminars.

Endorsed / Approved (P.O.)