

Course code	Course Name	L-T-P Credits	Year of Introduction
CS304	COMPILER DESIGN	3-0-0-3	2016
<b>Prerequisite: Nil</b>			
<b>Course Objectives</b>			
<ul style="list-style-type: none"> <li>To provide a thorough understanding of the internals of Compiler Design.</li> </ul>			
<b>Syllabus</b>			
Phases of compilation, Lexical analysis, Token Recognition, Syntax analysis, Bottom Up and Top Down Parsers, Syntax directed translation schemes, Intermediate Code Generation, Triples and Quadruples, Code Optimization, Code Generation.			
<b>Expected Outcome</b>			
The students will be able to			
<ol style="list-style-type: none"> <li>Explain the concepts and different phases of compilation with compile time error handling.</li> <li>Represent language tokens using regular expressions, context free grammar and finite automata and design lexical analyzer for a language.</li> <li>Compare top down with bottom up parsers, and develop appropriate parser to produce parse tree representation of the input.</li> <li>Generate intermediate code for statements in high level language.</li> <li>Design syntax directed translation schemes for a given context free grammar.</li> <li>Apply optimization techniques to intermediate code and generate machine code for high level language program.</li> </ol>			
<b>Text Books</b>			
<ol style="list-style-type: none"> <li>Aho A. Ravi Sethi and D Ullman. Compilers – Principles Techniques and Tools, Addison Wesley, 2006.</li> <li>D. M.Dhamdhare, System Programming and Operating Systems, Tata McGraw Hill &amp; Company, 1996.</li> </ol>			
<b>References</b>			
<ol style="list-style-type: none"> <li>Kenneth C. Loudon, Compiler Construction – Principles and Practice, Cengage Learning Indian Edition, 2006.</li> <li>Tremblay and Sorenson, The Theory and Practice of Compiler Writing, Tata McGraw Hill &amp; Company, 1984.</li> </ol>			
<b>Course Plan</b>			
Module	Contents	Hours	End Sem. Exam Marks
<b>I</b>	Introduction to compilers – Analysis of the source program, Phases of a compiler, Grouping of phases, compiler writing tools – bootstrapping <b>Lexical Analysis:</b> The role of Lexical Analyzer, Input Buffering, Specification of Tokens using Regular Expressions, Review of Finite Automata, Recognition of Tokens.	<b>07</b>	<b>15%</b>
<b>II</b>	<b>Syntax Analysis:</b> Review of Context-Free Grammars – Derivation trees and Parse Trees, Ambiguity. <b>Top-Down Parsing:</b> Recursive Descent parsing, Predictive parsing, LL(1) Grammars.	<b>06</b>	<b>15%</b>

<b>FIRST INTERNAL EXAM</b>			
<b>III</b>	<b>Bottom-Up Parsing:</b> Shift Reduce parsing – Operator precedence parsing (Concepts only) LR parsing – Constructing SLR parsing tables, Constructing, Canonical LR parsing tables and Constructing LALR parsing tables.	<b>07</b>	<b>15%</b>
<b>IV</b>	<b>Syntax directed translation:</b> Syntax directed definitions, Bottom- up evaluation of S-attributed definitions, L- attributed definitions, Top-down translation, Bottom-up evaluation of inherited attributes. <b>Type Checking :</b> Type systems, Specification of a simple type checker.	<b>08</b>	<b>15%</b>
<b>SECOND INTERNAL EXAM</b>			
<b>V</b>	<b>Run-Time Environments:</b> Source Language issues, Storage organization, Storage-allocation strategies. <b>Intermediate Code Generation (ICG):</b> Intermediate languages – Graphical representations, Three-Address code, Quadruples, Triples. Assignment statements, Boolean expressions.	<b>07</b>	<b>20%</b>
<b>VI</b>	<b>Code Optimization:</b> Principal sources of optimization, Optimization of Basic blocks <b>Code generation:</b> Issues in the design of a code generator. The target machine, A simple code generator.	<b>07</b>	<b>20%</b>
<b>END SEMESTER EXAM</b>			

### Question Paper Pattern

1. There will be *five* parts in the question paper – A, B, C, D, E
2. Part A
  - a. Total marks : 12      b.. Four questions each having 3 marks, uniformly covering modules I and II; All four questions have to be answered.
3. Part B
  - a. Total marks : 18      b. Three questionseach having 9 marks, uniformly covering modules I and II; Two questions have to be answered. Each question can have a maximum of three subparts.
4. Part C
  - a. Total marks : 12      b. Four questions each having 3 marks, uniformly covering modules III and IV; All four questions have to be answered.
5. Part D
  - a. Total marks : 18      b. Three questions each having 9 marks, uniformly covering modules III and IV; Two questions have to be answered. Each question can have a maximum of three subparts
6. Part E
  - b. Total Marks: 40      b. Six questions each carrying 10 marks, uniformly covering modules V and VI; four questions have to be answered.
  - c. A question can have a maximum of three sub-parts.
7. There should be at least 60% analytical/numerical questions.