<table>
<tr><th rowspan="2">Course Code</th><th rowspan="2">Course Name</th><th colspan="3">Teaching Scheme (Hrs/week)</th><th colspan="4">Credits Assigned</th></tr>
<tr><th>L</th><th>T</th><th>P</th><th>L</th><th>T</th><th>P</th><th>Total</th></tr>
<tr><td rowspan="4">CPC601</td><td rowspan="4"><b>System Programming and Compiler Construction</b></td><td><b>4</b></td><td><b>-</b></td><td><b>--</b></td><td><b>4</b></td><td><b>-</b></td><td><b>--</b></td><td><b>4</b></td></tr>
<tr><td colspan="7"><b>Examination Scheme</b></td></tr>
<tr><td colspan="2"><b>ISE</b></td><td colspan="2"><b>MSE</b></td><td colspan="3"><b>ESE</b></td></tr>
<tr><td colspan="2"><b>10</b></td><td colspan="2"><b>30</b></td><td colspan="3"><b>100 (60% Weightage)</b></td></tr>
</table>

<table>
<tr><td><b>Pre-requisite Course Codes</b></td><td colspan="2">CPC502 (Operating System)</td></tr>
<tr><td colspan="3">At end of successful completion of this course, student will be able to</td></tr>
<tr><td rowspan="5"><b>Course Outcomes</b></td><td>CO1</td><td>To understand the basics of system programs like editors, compiler, assembler, linker, loader, interpreter and debugger.</td></tr>
<tr><td>CO2</td><td>To understand concepts of assemblers and write macros as and when required to increase readability and productivity.</td></tr>
<tr><td>CO3</td><td>To understand the various phases of compiler and working of parsers.</td></tr>
<tr><td>CO4</td><td>To understand role of Intermediate Code Generation in connection with language designing and apply optimization principles on given code.</td></tr>
<tr><td>CO5</td><td>To analyze the role of Operating System functions such as memory management aspertaining to run time storage management.</td></tr>
</table>

Module No.	Topics	Ref.	Hrs.
1	**System Software** Concept, introduction to various system programs such as assemblers, loaders , linkers ,macro processors, compilers, interpreters, operating systems, device drivers	1-5	**01**
2	**Assemblers** General Design Procedure , Design of Assembler (Single Pass –Assembler IBM PC , multi pass Assembler - IBM 360/370 Processor),Statement of Problem , Data Structure , format of Databases , Algorithm, Look for modularity	1-5	**06**
3	**Macros & Macro processors** Macro instructions, Features of Macro facility, Design of 2 passmacroprocessor	1-5	**04**
4	**Loaders and Linkers** Loader schemes, Design of Absolute loader , Design of Direct linking Loader	1-5	**04**
5	**Software Tools** Software Tools for Program development, Editors: Types of	1-5	**02**

	Editors,Design of Editor ,Debug Monitors		
6	**Compilers** Introduction to Compilers, Phases of a compiler, comparison ofcompilers and interpreters.	1-5	**02**
7	Lexical Analysis Role of a Lexical analyzer, inputbuffering, specification andrecognition of tokens, Designing a lexical analyzer generator,Patternmatching based on NFA’s.	1-5	**02**
8	**Syntax Analysis** Role of Parser, Top down parsing, Recursive descent and predictiveparsers (LL), Bottom Up parsing, Operator precedence parsing, LR,SLR and LALR parsers.	1-5	**08**
9	**Syntax Directed Translation** Syntax directed definitions, Inherited and Synthesized attributes,Evaluation order for SDDs , S attributed Definitions , L attributedDefinitions	1-5	**03**
10	**Intermediate Code Generation** Intermediate languages: declarations, Assignment statements,Boolean expression, case statements, back patching, procedure calls.	1-5	**04**
11	**Code Generation** Issues in the design of Code Generator , Basic Blocks and Flow graphs, code generation algorithm , DAG representation of Basic Block	1-5	**04**
12	**Code Optimization** Principal sources of Optimization, Optimization of Basic Blocks, Loops in Flow graph ,Peephole Optimization	1-5	**03**
13	**Run Time storage** Storage Organization , storage allocation strategies, parameterpassing , Symbol table , introduction to garbage collection andcompaction	1-5	**04**
14	**Compiler-compilers** JAVA compiler environment, YACC compiler-compiler	1-5	**01**
		Total	**48**

References:

[1] J. J Donovan,“Systems Programming”, Tata McGraw Hill Publishing Company

[2] A. V. Aho, R. Shethi and J.D. Ulman, “Compilers - Principles, Techniques and Tools” , Pearson Education

[3] A. V. Aho, R. Shethi, Monica Lam , J.D. Ulman “Compilers - Principles, Techniques and Tools” , Pearson Education , Second Edition.

[4] D. M Dhamdhere,“Systems programming”, Tata McGraw Hill

[5] John R. Levine, Tony Mason & Doug Brown,”lex&yacc”, O’Reilly, 2nd Edition