

**NIRMA UNIVERSITY**  
**Institute of Technology**  
**B.Tech. Computer Science and Engineering**  
**Semester-VI**  
**Department Elective-II**

L	T	P	C
3	0	2	4

<b>Course Code</b>	2CSDE65
<b>Course Title</b>	System Programming

**Course Outcomes:**

At the end of the course student will be able to –

1. explain the concepts and principles of system programming and understand the roles and scope of a system programmer
2. interpret both theoretical and practical aspects of system programming, and understand techniques for designing and implementing system-level programs
3. apply knowledge of different phases and steps to mimic a simple language translator
4. analyze the working of various system software like assembler, loader, linker, editor and device driver.

**Syllabus**

**Teaching  
Hours:45  
08**

**Unit I**

**Overview of System Software:** Software Hierarchy, Systems Programming, Machine Structure, Interfaces, Address Space, Computer Languages  
**Overview of Language Processors:** Programming Languages and Language Processors, Language Processing Activities, Program Execution, Fundamental of Language Processing, Symbol Tables, Data Structures for Language Processing: Search Data structures, Allocation Data Structures

**Unit II**

**Assemblers:** Elements of Assembly Language Programming, Design of the Assembler, Assembler Design Criteria, Types of Assemblers, Two-Pass Assemblers, One-Pass Assemblers, Single pass Assembler for Intel x86 , Algorithm of Single Pass Assembler, Multi-Pass Assemblers, Advanced Assembly Process, Variants of Assemblers Design of two pass assembler

**Unit III**

**Macro and Macro Processors:** Introduction, Macro Definition and Call, Macro Expansion, Nested Macro Calls, Advanced Macro Facilities, Design Of a Macro Pre-processor, Design of a Macro Assembler, Functions of a Macro Processor, Basic Tasks of a Macro Processor, Design Issues of Macro Processors, Features, Macro Processor Design Options, Two-Pass Macro Processors, One-Pass Macro Processors



<b>Unit IV</b>	<b>08</b>
<b>Linkers and Loaders:</b> Introduction, Relocation of Linking Concept, Design of a Linker, Self-Relocating Programs, Linking in MSDOS, Linking of Overlay Structured Programs, Dynamic Linking, Loaders, Different Loading Schemes, Sequential and Direct Loaders, Compile-and-Go Loaders, General Loader Schemes, Absolute Loaders, Relocating Loaders, Practical Relocating Loaders, Linking Loaders, Relocating Linking Loaders, Linkers v/s Loaders	
<b>Unit V</b>	<b>08</b>
<b>Scanning and Parsing:</b> Programming Language Grammars, Classification of Grammar, Ambiguity in Grammatical Specification, Scanning, Parsing, Top Down Parsing, Bottom up Parsing, Language Processor Development Tools, LEX, YACC	
<b>Compilers:</b> Causes of Large Semantic Gap, Binding and Binding Times, Data Structure used in Compiling, Scope Rules, Memory Allocation, Compilation of Expression, Compilation of Control Structure, Code Optimization	
<b>Unit VI</b>	<b>05</b>
<b>Interpreters &amp; Debuggers:</b> Benefits of Interpretation, Overview of Interpretation, The Java Language Environment, Java Virtual Machine, Types of Errors, Debugging Procedures, Classification of Debuggers, Dynamic/Interactive Debugger	

### Self-Study:

The self-study contents will be declared at the commencement of semester. Around 10% of the questions will be asked from self-study contents.

### Laboratory Work:

Laboratory work will be based on the above syllabus with minimum 10 experiments to be incorporated.

### Suggested Reading<sup>^</sup>:

1. D M Dhamdhere, System Programming, McGraw Hill
2. Srimanta Pal, System Programming, OXFORD
3. R.K. Maurya & A. Godbole., System Programming, Wiley
4. Leland L. Beck, System Software –An Introduction to Systems Programming , Pearson Education Asia
5. Santanu Chattopadhyay, System Software, Prentice-Hall

L=Lecture, T=Tutorial, P=Practical, C=Credit

---

<sup>^</sup>this is not an exhaustive list