**SYSTEM PROGRAMMING**

**UNIT 1:** IBM 360/370 & Assembler – Introduction to System Programming & its components, M/c Architecture, Instruction Formats, Data Formats & Register Formats, Concept of assembler, design of single pass and two pass assembler.

**UNIT 2:** Macroprocessor – Concept of macro, macro call within macro, macro definition within macro, recursive macro calls, design of macro processor.

**UNIT 3:** Linker and Loader – Concept of static and dynamic relocation, external symbols, design of linker, design of object file for different loading schemes.

**UNIT 4:** Common Object files format & System Utilities – Structure of object file and executable file, section or segment headers, symbol table, concept of storage class, string, various data types, line insert, character, arrays structures. Source code control system), make, link editor, symbolic debugger, GNU debugger.

**UNIT 5:** Unix Device Drivers – Definition, Anatomy and Types, Device programming, Installation, Incorporation of driver routines, Basic device operation, Implementation with Line Printer, Comparative study between device drivers for UNIX & Windows.

**UNIT 6:** Compiler – Phases of Compilers, Overview of Databases and Algorithms required for all phases. Role of lexical analyzer, recognition of tokens, Study of LEX & YACC.

**Text Books:** 1. System Programming- J. J. Donovan, Tata McGraw-Hill Education. 2. UNIX Device Drivers- George Pajari, Pearson Education. 3. UNIX system Utilities manual. 4. UNIX programming Tools LEX and YACC –Levine, Mason and Brown, O’Reilly.

**Reference Books:** 1. System Programming and Operating systems- D. M. Dhamdhere, Tata McGraw-Hill Education. 2. UNIX programming Environment- Keringham and Pike, PHI. 3. System Software: An introduction to systems programming- Leland L. Beck, Pearson Education. 4. Principles of Compiler Design-Aho and Ullman, Pearson Education.