ITM(SLS) Baroda University - Faculty of Engineering

Department of Computer Science & Engineering
SYLLABUS FOR 3 Semester BTech PROGRAMME
Systems Software (C2310C3)

Type of Course: BTech

Prerequisite:

Rationale: -

Teaching and Examination Scheme:

Teaching Scheme			Credit	Examination Scheme					Total
Lecture Hrs/ Week	Tutorial Hrs/ Week	Practical Hrs/ Week		External		Internal			
				T	P	T	CE	P	
3	-	2	4	100	-	60	-	50	210

SEE - Semester End Examination, **CIA** - Continuous Internal Assessment (It consists of Assignments/Seminars/Presentations/MCQ Tests, etc.)

Contents:

Sr.	Topic	Weightage	Teaching Hrs.
1	**Overview of System Software and Text Editors**: Introduction, Software, Software Hierarchy, Systems Programming Tools, LifeCycle Source Program, Levels of SystemSoftware. Text Editors: Overview of Editing Process, User Interface, Editor Structure, Text Editors - line - by - line (exampled) ,file-oriented WYSIWYG (example)	%	6
2	**Overview of Language Processors::** Programming Languages and LanguageProcessors, Language Processing Activities, Program Execution, Fundamental of Language Processing, Symbol Tables Data Structures for Language Processing: Search Data structures, Allocation Data Structures.	%	6
3	**Assemblers**: Elements of Assembly Language Programming ,Design of the Assembler, Assembler Design Criteria, Types of Assemblers, Two - Pass Assemblers, One Pass Assemblers. Macro and Macro Processors Introduction, Macro Definition and Call, Macro Expansion, Nested Macro Calls, Advanced Macro Facilities, Design of a MacroPre-processor, Design of a Macro Assembler, Functions of a Macro Processor, Basic Tasks of MacroProcessor, Design Issues of Macro Processors	%	10
4	**Overview of the Compiler::** A Simple Compiler, Difference between interpreter, assembler and compiler. Types of Compiler, Analysis of the Source Program, The Phases of a Compiler, parsing techniques,The Grouping of Phases. Frontend and backend of compiler, Parsing Techniques,Code optimization Techniques.	%	10

5	**Interpreter & Debuggers:**: Benefits Interpretation, Overview of Interpretation, The Java Language Environment, JavaVirtualMachine, Types of Errors, Debugging Procedures, Classification of Debuggers, Dynamic/Interactive Debugger.	%	5
6	**Linker & Loader:**: Introduction, Relocation of Linking Concept, Design of Linker, Self-RelocatingPrograms, linking in Linux, Linking Overlay Structured Programs, Dynamic Linking, Loaders, Different Loading Schemes, Sequential and Direct Loaders,Compile-and-Go Loaders.	%	5

*Continuous Evaluation:

It consists of Assignments/Seminars/Presentations/Quizzes/Surprise Tests (Summative/MCQ) etc.

Reference Books:

1. "System Programming and Operating System"
   By D.M.Dhamdhere, Tata McGraw Hill. | 2nd Ed
2. Compilers-Principles and Practice
   By P.H.Dave and H.B.Dave | Pearson Education
3. System Programming
   By Srimanta Pal | OXFORD Publication
4. System Programming and Compiler Construction
   By R.K.Maurya & A.Godbole
5. System Software– An Introduction to Systems Programming (TextBook)
   By bril Lel and Beck | Pearson Education Asia, 2000 | 3rd Edition
6. System Software (TextBook)
   By Santanu Chattopadhyay | Prentice-Hall India

List of Practical:

1. Write a C program to identify whether a given line is a comment or not.
2. Write a C program to test whether a given identifier is valid or not.
3. Write a C program to simulate lexical analyzer for validating operators.
4. Implement following programs using Lex.
5. Use Macro Features Of C language
6. Write A Program To Left factor the given grammar
7. Lex program to Print out all numbers from given file.
8. Write a program to remove the Left Recursion from a given grammar
9. Write a program which generates Quadruple Table for the given post fix String.
10. Write a C program to parse a given string using Predictive parsing for given grammar.Type→ simple |↑id |array[simple] of type simple→integer |char | num dot dot num
11. Write a program to remove the Left Recursion from a given grammar.
12. Write a Program identify all the tokens from the source code.
13. Write a Program for first pass and second pass Assembler.
14. Write A program to show linker.
15. Write a C program to generate in triple intermediate code for assignment statement.