GUJARAT TECHNOLOGICAL UNIVERSITY

SYSTEM PROGRAMMING
SUBJECT CODE: 2150708
B.E. 5thSEMESTER

Type of course: System Programming

Prerequisite: Data Structures and Operating Systems

Rationale: NA

Teaching and Examination Scheme:

Teaching Scheme			Credits	Examination Marks						Total Marks
L	T	P	C	Theory Marks			Practical Marks			
				ESE (E)	PA (M)		ESE (V)		PA (I)	
					PA	ALA	ESE	OEP		
4	0	2	6	70	20	10	20	10	20	150

Content:

Sr. No.	Content	Total Hrs	% Weightage
1	Overview of System Software Introduction, Software, Software Hierarchy, Systems Programming, Machine Structure, Interfaces, Address Space, Computer Languages, Tools, Life Cycle of a Source Program, Different Views on the Meaning of a Program, System Software Development, Recent Trends in Software Development, Levels of System Software	06	10%
2	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.	06	15%
3	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,	06	15%
4	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	08	20%
5	Linkers and Loaders	06	20%

	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		
6	Scanning and Parsing Programming Language Grammars, Classification of Grammar, Ambiguity in Grammatic Specification, Scanning, Parsing, Top Down Parsing, Bottom up Parsing, Language Processor Development Tools, LEX, YACC	06	10%
7	Compilers 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	04	5%
8	Interpreters & Debuggers Benefits of Interpretation, Overview of Interpretation, The Java Language Environment, Java Virtual Machine, Types of Errors, Debugging Procedures, Classification of Debuggers, Dynamic/Interactive Debugger	04	5%

Suggested Specification table with Marks (Theory):

Distribution of Theory Marks					
R Level	U Level	A Level	N Level	E Level	C Level
7	30	33	-	-	-

Legends: R: Remembrance; U: Understanding; A: Application, N: Analyze and E: Evaluate C: Create and above Levels (Revised Bloom's Taxonomy)

Note: This specification table shall be treated as a general guideline for students and teachers. The actual distribution of marks in the question paper may vary slightly from above table.

Reference Books:

1) System Programming by D M Dhamdhere McGraw Hill Publication
2) System Programming by Srimanta Pal OXFORD Publication
3) System Programming and Compiler Construction by R.K. Maurya & A. Godbole.
4) System Software – An Introduction to Systems Programming by Leland L. Beck, 3rd Edition, Pearson Education Asia, 2000
5) System Software by Santanu Chattopadhyay, Prentice-Hall India,2007

Course Outcome:

After learning the course the students should be able to:

1. To understand the execution process of HLL programs.
2. To understand the working of scanners and parsers.

3. To understand the basic design of various system software.
4. To implement various system software.

List of Experiments and Design based Problems (DP)/Open Ended Problem:

(Pl. Note: List of Experiments should be as per theory covered in the class, below mentioned practical are just for the reference purpose)

1. Write a program to implement the lexical analyzer.
2. Write a Lexical Analyzer (using lex utility for UNIX).
3. Write a program to left factor the given grammar.
4. Write a program to remove the Left Recursion from a given grammar.
5. Aim: Implement Recursive Descendent Parsing for the given Grammar.
   E -> T + E / T
   T -> F * T / F
   F -> (E) / i
6. Implement Predictive Parser for the given grammar.
   E -> T + E / T
   T -> F * T / F
   F -> (E) / i
7. Write a SAL program in text file and generate SYMTAB and LITTAB
8. Use macro features of C language
9. Write a program which generates Quadruple Table for the given postfix 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 dotdot num

List of Open Source Software/learning website:

- www.cs.jhu.edu/~scott/pl/lectures/parsing.html
- www.en.wikipedia.org/wiki/System_programming

ACTIVE LEARNING ASSIGNMENTS: Preparation of power-point slides, which include videos, animations, pictures, graphics for better understanding theory and practical work – The faculty will allocate chapters/ parts of chapters to groups of students so that the entire syllabus to be covered. The power-point slides should be put up on the web-site of the College/ Institute, along with the names of the students of the group, the name of the faculty, Department and College on the first slide. The best three works should submit to GTU.