

**DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING**

**SCHEME OF TEACHING AND EXAMINATION**

**B.E. (ISE) V SEMESTER  
(ACADEMIC YEAR 2011-12)**

Sl No	Subject Code	Subject	Credits	Hours/Week			Examination Marks		
				Lecture	Tutorial	Practical	CIE	SEE	Total
1	UIS501C	System Software	04	4	0	0	50	50	100
2	UIS512C	Software Engineering	04	4	0	0	50	50	100
3	UIS503C	Database Management Systems	04	4	0	0	50	50	100
4	UIS504C	Data Communication	03	3	0	0	50	50	100
5		Elective – I	03	3	0	0	50	50	100
6		Open Elective – I	03	3	0	0	50	50	100
7	UIS505L	System Software Laboratory	1.5	0	1	2	50	50	100
8	UIS506L	Microprocessor Laboratory	1.5	0	1	2	50	50	100
9	UIS509L	Operating System Lab	1.5	0	1	2	50	50	100
		<b>Total</b>	<b>25.5</b>	<b>21</b>	<b>3</b>	<b>6</b>	<b>450</b>	<b>450</b>	<b>900</b>

DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING

SCHEME OF TEACHING AND EXAMINATION

B.E. (ISE) VI SEMESTER  
(ACADEMIC YEAR 2011-12)

Sl. No.	Subject Code	Subject	Credits	Hours/Week			Examination Marks		
				Lecture	Tutorial	Practical	CIE	SEE	Total
1	UIS611H	Management and Entrepreneurship	03	3	0	0	50	50	100
2	UIS612C	File Information Structures	04	4	0	0	50	50	100
3	UIS603C	Computer Networks.	04	4	0	0	50	50	100
4	UIS604C	Unix System Programming.	04	4	0	0	50	50	100
5		Elective-II	03	3	0	0	50	50	100
6		Open Elective – II	03	3	0	0	50	50	100
7	UIS608L	Unix System Programming Laboratory	1.5	0	1	2	50	50	100
8	UIS609L	Database Application Laboratory	1.5	0	1	2	50	50	100
9	UIS610L	File Information Structures Laboratory	1.5	0	1	2	50	50	100
		<b>Total</b>	<b>25.5</b>	<b>21</b>	<b>3</b>	<b>6</b>	<b>450</b>	<b>450</b>	<b>900</b>

## SCHEME OF TEACHING AND EXAMINATION

### B.E. (ISE) VII SEMESTER (ACADEMIC YEAR 2011 - 12)

Sl No	Subject Code	Subject	Credits	Hours/Week			Examination Marks		
				Lecture	Tutorial	Practical	CIE	SEE	Total
1	UIS701C	Web Technologies	3	3	0	0	50	50	100
2	UIS702C	Object Oriented Modeling and Design	4	4	0	0	50	50	100
3	UIS713C	Management Information Systems	3	3	0	0	50	50	100
4		Elective- III	3	3	0	0	50	50	100
5		Elective- IV	3	3	0	0	50	50	100
6		Open Elective – III	3	3	0	0	50	50	100
7	UIS707L	Object Oriented System Design Lab	1	0	0	2	50	50	100
8	UIS708L	Web Programming Lab	1.5	0	1	2	50	50	100
9	UIS709L	Computer Network Lab	1.5	0	1	2	50	50	100
10	UIS710P	Project Phase – I	1	0	--	--	50	50	100
		<b>Total</b>	<b>24</b>	<b>19</b>	<b>3</b>	<b>6</b>	<b>450</b>	<b>450</b>	<b>900</b>

## SCHEME OF TEACHING AND EXAMINATION

### B.E. (ISE) VIII SEMESTER (ACADEMIC YEAR 2011 - 12)

Sl No	Subject Code	Subject	Credits	Hours/Week			Examination Marks		
				Lecture	Tutorial	Practical	CIE	SEE	Total
1		Elective – V	03	3	0	0	50	50	100
2		Elective – VI	03	3	0	0	50	50	100
3	UIS803P	Project Phase – II	18	0	-	-	50	50	100
4	UIS804P	Seminar	01	0	-	-	50	50	50
		<b>Total</b>	<b>25</b>	<b>6</b>	<b>0</b>	<b>6</b>	<b>200</b>	<b>150</b>	<b>350</b>

## V SEMESTER

### UIS501C: SYSTEMS SOFTWARE

4-credits (4-0- 0)

#### UNIT - 1

**MACHINE ARCHITECTURE:** Introduction, System Software and Machine Architecture, simplified Instructional Computer (SIC) – SIC Machine Architecture, SIC/XE Machine Architecture, SIC Programming Examples. RISC processor – Ultra SPARK Architecture, CISC processor – Pentium Pro Architecture 05 Hours

**ASSEMBLERS:** Basic Assembler Function - A Simple SIC Assembler, Assembler Algorithm and Data Structures, Machine Dependent Assembler Features - Instruction Formats & addressing Modes, Program Relocation.

Machine Independent Assembler Features – Literals, Symbol-Definition Statements, Expression, Program Blocks, Control Sections and Programming Linking, Assembler Design Operations - One- Pass Assembler, Multi-Pass Assembler. 08 Hours

#### UNIT - 2

**LOADERS AND LINKERS:** Basic Loader Functions - Design of an Absolute Loader, A Simple Bootstrap Loader, Machine-Dependent Loader Features – Relocation, Program Linking, Algorithm and Data Structures for a Linking Loader; Machine-Independent Loader Features - Automatic Library Search, Loader Options, Loader Design Options - Linkage Editor, Dynamic Linkage, Bootstrap Loaders. 08 Hours

**MACRO PROCESSOR:** Basic Macro Processor Functions – Macro Definitions and Expansion, Macro Processor Algorithm and Data Structures, Machine-Independent Macro Processor Features - Concatenation of Macro Parameters, Generation of Unique Labels, Conditional Macro Expansion, Keyword Macro Parameters, Macro Processor Design Options – Recursive Macro Expansion, General-Purpose Macro Processors, Macro Processing Within Language Translators. 05 Hours

#### UNIT - 3

**COMPILERS:** Basic Compiler Function – Grammars, Lexical Analysis, Syntactic Analysis, Code Generation, Machine Dependent Compiler Features: Intermediate Form of the Program, Machine-Dependent Code Optimization

Machine Independent Compiler Features: Structured Variables, Machine-Independent Code Optimization, Storage Allocation, Block-Structured Languages, Compiler Design Options: Division into Passes, Interpreters, P-Code Compilers, Compiler-Compilers. 13 Hours

#### UNIT - 4

**LEX AND YACC:** Lex and Yacc - The Simplest Lex Program, Recognizing Words With LEX, Symbol Tables, Grammars, Parser-Lexer Communication, The Parts of Speech Lexer, A YACC Parser, The Rules Section, Running LEX and YACC, LEX and Hand- Written Lexers, Using LEX - Regular Expression, Examples of Regular Expressions, A Word Counting Program, Parsing a Command Line.

Using YACC - Grammars, Recursive Rules, Shift/Reduce Parsing, What YACC Cannot Parse, A YACC Parser – The Definition Section, The Rules Section, Symbol Values and Actions, The LEXER, Compiling and Running a Simple Parser, Arithmetic Expressions and Ambiguity, Variables and Typed Tokens. 13 Hours

**TEXT BOOK:**

1. **System Software** - Leland. L. Beck, 3rd Edition, Addison-Wesley,1997.
2. **Lex and Yacc** - John. R. Levine, Mason and Doug Brown, O'Reilly,SPD, 1998.

**REFERENCE BOOK:**

1. **System Programming and Operating Systems** – D.M.Dhamdhere,2nd Edition, Tata McGraw - Hill, 1999.

**UIS512C: SOFTWARE ENGINEERING**  
**4-credits (4-0-0)**

**UNIT – I**

**Introduction:** Concepts of Software Engineering, Professional and ethical responsibilities.  
1 hour

**Socio-Technical systems:** Emergent system properties, systems engineering, organizations, people and computer systems, legacy systems.  
3 hours

**Critical systems:** A simple safety critical system, system dependability, Availability and reliability, safety, security.  
2 hours

**Software Processes:** software process models, process iteration, process activities, rational unifies process, computer aided software engineering.  
7 hours

**UNIT – II**

**Software requirements:** Functional and non functional requirements, user requirements, system requirements, the software requirements document.  
4 hours

**Requirements engineering process:** Feasibility studies, requirements elicitation and analysis, requirements validation, requirements management.  
5 hours

**System Models:** Context based models, behavioral models, data models, object models, structured methods.  
4 hours

**UNIT – III**

**Architectural design:** Architectural design decisions, system organization, modular decomposition styles, control styles, reference architectures.  
4 hours

**Object oriented design:** objects and object classes, an object oriented design process, design evolution.  
3 hours

**User interface design:** Design issues, UI design process, user analysis, user interface prototyping, interface evaluation.  
3 hours

**Rapid software development:** agile methods, extreme programming, rapid application development, software prototyping.  
3 hours

**UNIT – IV**

**Verification and validation:** Planning verification and validation, software inspections, automated static analysis, verification and formal methods.  
4 hours.

**Software testing:** system testing, component testing, test case design, test automation.  
3 hours.

**Managing people:** selecting staff, motivating people, managing groups, People capability maturity model.  
2 hours

**Software cost estimation:** software productivity, estimation techniques, algorithmic cost modeling, project duration and staffing.

2 hours

**Quality Management:** Process and product quality, quality assurance and standards, CMMI process framework.

2 hours

Text Book:

1. Ian Somerville, Software Engineering, 7<sup>th</sup> edition, Person Education.

Reference Books:

2. Pressman R.S, "Software Engineering A Practitioners Approach", MGH New Delhi.
- Jalote P, "An Intergrated Approach to Software Engineering", Narosa New Delhi

**UIS503C: DATABASE MANAGEMENT SYSTEMS**  
**4-Credits (4-0-0)**

**UNIT – 1**

**INTRODUCTION:** Introduction; An example; Characteristics of database approach; Advantages of using DBMS approach; when not to use a DBMS. Data models, schemas and instances; Three-schema architecture and data independence; Database languages and interfaces; The database system environment; Centralized and client-server architectures; Classification of Database Management systems. **6 hours**

**ENTITY-RELATIONSHIP MODEL:** Using High-Level Conceptual Data Models for Database Design; An example database application; Entity types, Entity sets, Attributes and Keys; Relationship types, Relationship sets, Roles and Structural constraints; Weak entity types; Refining the ER Design; ER Diagrams, Naming conventions and design issues; Relationship types of degree higher than two. **6 hours**

**UNIT – 2**

**RELATIONAL MODEL AND RELATIONAL DATABASE CONSTRAINTS:** Relational model concepts; Relational model constraints and Relational database schemas; Update operations, Transaction and dealing with constraint violations. **3 hours**

**RELATIONAL ALGEBRA:** Unary relational operations: SELECT and PROJECT; Relational algebra operations from set theory; Binary relational operations: JOIN and DIVISION; Additional relational operations; Examples of queries in relational algebra; Relational database design using ER- to-Relational mapping. **6 hours**

**SQL:** data definition and data types; Specifying basic constraints in SQL; Schema change statements in SQL; Basic queries in SQL; More complex SQL queries. Insert, Delete and Update statements in SQL; Specifying constraints as Assertion and Trigger; Views (Virtual Tables) in SQL; **5 hours**

**UNIT – 3**

**DATABASE DESIGN:** Informal design guidelines for relation schemas; Functional dependencies; Normal forms based on primary keys; General definitions of second and third normal forms; Boyce-Codd Normal Form **6 hours**

Properties of relational decompositions; Algorithms for relational database Schema design; Multivalued dependencies and Fourth Normal Form; Join Dependencies and Fifth Normal Form; Inclusion Dependencies; Other Dependencies and Normal forms. **7 hours**

**UNIT – 4**

**TRANSACTION MANAGEMENT :** Introduction to transaction processing; Transaction & system concepts; Desirable properties of transactions; Characterizing schedules based on recoverability; Characterizing schedules based on serializability; Transaction support in SQL; **CONCURRENCY CONTROL:** Two-phase locking techniques for concurrency control; **8 hours**

**CRASH RECOVERY:** Recovery concepts; Recovery techniques based on deferred update; recovery techniques based on immediate update; shadow paging; The ARIES recovery algorithm; **5 hours**

Text book :

1. Ramez Elmasri & Shamkant B. Navathe “Fundamentals of Database Systems”, 5<sup>th</sup> Edition, Pearson Education;

Reference book:

1. Ramakrishnan Gehrke “ Database Management Systems”, 3<sup>rd</sup> edition, McGraw-Hill Higher Education;
2. C. J. Date, “An Introduction to Data base systems”, Addison Wesley, 4<sup>th</sup> edition.

**UIS504C: DATA COMMUNICATIONS**  
**3-Credits (3-0-0)**

**UNIT-I**

**Introduction:**

**Data Communications:** Components, Data representations, Data flow, Networks: Distributed processing, Network Criteria, physical structures, network models, categories of Networks, internetwork, The Internet: Brief history, the internet today, Protocols and Standards: protocols, standards, standard organizations, internet standards, Layered tasks: sender, receiver and carrier The OSI Model: layered architecture, peer to peer processes, encapsulation, Layers in the OSI model, TCP / IP Protocol Suite: physical, data link, transport and application layer, Addressing: physical, logical and port addresses.

**Digital Transmissions:**

Digital-to-Digital conversion: line coding, line coding schemes, Analog-to-Digital conversion: PCM, Transmission modes: parallel and serial transmissions.

**10 hours**

**UNIT-II**

**Analog Transmissions:**

Digital - to - Analog conversion; Aspects, ASK, FSK, PSK, Analog - to - Analog conversion: AM, FM, PM, Multiplexing and Demultiplexing: Frequency Division, Wavelength division, and time division (Synchronous and Statistical) multiplexing

**TRANSMISSION MEDIA:** Twisted pair cable, Coaxial cable, Fiber-Optic cable, Radio waves, Microwaves, Infrared. **ERROR DETECTION AND CORRECTION:** Introduction to error detection / correction, Block coding, linear block codes, cyclic codes, Checksum

**SWITCHING:** Circuit switched networks, Data gram Networks, Virtual circuit networks, structure of a switch

**11 hours**

**UNIT-III**

**DATA LINK CONTROL:** Framing: fixed and variable sized framing, Flow and Error control, Protocols, Noiseless channels, Noisy channels, HDLC, Point-to-point Protocol: framing, transition phases.

**MULTIPLE ACCESSES:** Random Access: CSMA, CSMA/CD, CSMA/CA, Controlled Access: reservation, polling, token poling, Canalizations: FDMA, TDMA, CDMA

**11 hours**

**UNIT-IV**

**Wired LANs: Ethernet**

**IEEE standards:** Data link layer, physical layer, Standard Ethernet: MAC Sub layer, physical layer, changes in the standards: bridged Ethernet, switched Ethernet, full duplex Ethernet, Fast and Gigabit Ethernet

**WIRELESS LANS AND CONNECTION OF LANS**

IEEE 802.11: Architecture, MAC sub layer, Addressing mechanism, physical layer. Bluetooth. Connecting devices, Cellular telephony

**10 hours**

**TEXT BOOK:**

1. **Data Communications and Networking** – Behrouz A. Forouzan, 4th Edition, Tata McGraw-Hill, 2006.

**REFERENCE BOOKS:**

1. **Communication Networks: Fundamental Concepts and Key Architectures** - Alberto Leon, Garcia and Indra Widjaja, 3<sup>rd</sup> Edition, Tata McGraw- Hill, 2004.
2. **Data and Computer Communication**, William Stallings, 8<sup>th</sup> Edition, Pearson Education, 2007.
3. **Computer Networks: A Systems Approach** - Larry L. Peterson and Bruce S. David, 4th Edition, Elsevier, 2007.
4. **Introduction to Data Communications and Networking** – Wayne Tomasi, Pearson Education, 2005.
5. **Computer and Communication Networks** – Nader F. Mir, Pearson Education, 2007.

## UIS505L: SYSTEMS SOFTWARE LABORATORY

1.5 CREDITS (0-1-2)

### PART A

#### *Execution of the following programs using C:*

- 1) Write a C program to extract label, opcode and operands given any SIC/XE instruction belonging to any one of the formats discussed below;  
F1: label opcode  
F2: label opcode r1 r2  
F3 : label opcode displacement
- 2) Write a C program translate the given arithmetic statement into SIC/XE assembly code.  
Example:  $C = x+y$  must be translated to;  
LDA x  
ADD y  
STA C
- 3) Write a C program to generate symbol table of assembler.
- 4) Write a C program to machine code an instruction belonging to any one of the formats discussed below;  
F1: label opcode  
F2: label opcode r1 r2  
F3 : label opcode displacement
- 5) Write a c-program to implement lexical analysis phase of a compiler.

### PART B

#### *Execution of the following programs using LEX:*

- 1) Program to count the number special symbols of C program.
- 2) Program to count number of
  - a) Positive and negative integers
  - b) Positive negative fractions
- 3) Program to count the numbers of comment lines in a given C program. Also eliminate them and copy that program into separate file.
- 4) Program to count the number of 'scanf' and 'printf' statements in a C program. Replace them with 'readf' and 'writef' statements respectively.
- 5) Program to recognize a valid arithmetic expression and identify the identifiers and operators present. Print them separately.

***Execution of the following programs using YACC:***

- 1) Program to recognize nested IF control statements and display the number of levels of nesting.
- 2) Program to recognize a valid arithmetic expression that uses +, -, \* and /.
- 3) Program to recognize a valid variable, which starts with a letter, followed by any number of letters or digits.
- 4) Program to evaluate an arithmetic expression involving operators +, -, \* and / for integers and real data
- 5) Program to recognize the given the words such as '0' is bounded by equal No. of '1's towards left and right.

Ex : 101, 11011, 1110111 valid  
110, 01, 1101. 1011 are invalid.

**Instructions:**

In the examination, a combination of one question from Part A for 20 marks and one LEX and one YACC problem from Part B for 20 Marks has to be given and Viva must be conducted for 10 Marks.

**General Remarks:**

- 1) Lab schedule: 3hrs/week for each student (1 hr tutorial, 2 hrs- program execution).
- 2) Student should complete all the lab assignments.
- 3) Evaluation CIE 50 marks:  
Lab assignments : 30 marks  
Lab CIE : 10 marks

**UIS506L: MICROPROCESSOR LAB**  
**1.5 CREDITS (0-1-2)**

**Part A: Assembly Language Assignments.**

1. Write an assembly language program to simulate arithmetic calculator to support addition, subtraction, multiplication and division on given two 8-bit numbers.
2. Write an assembly language program to add two 3x3 matrices, where matrices are stored in array (row wise).
3. Write an assembly language program to read an alphanumeric character and display its equivalent ASCII code at specified location(x and y coordinate values are given) .
4. Write an assembly language program using macros as follows:
  - i) To read a character from the keyboard in the module (1)
  - ii) To display a character in module (2)
  - iii) Use the above two modules to read a string of characters from the keyboard terminated by the carriage return and print the string on the display in the next line.
5. Write an assembly language program to search a key element in a list of 'n' 16-bit numbers using Binary search algorithm.
6. Write an assembly language program to reverse a given string and check whether it is palindrome or not.
7. Write an assembly language program to generate the first N Fibonacci numbers using procedures.
8. Write an assembly language program to create a file (Input File) and to delete an existing file.

**Part B: Interfacing Assignments.**

1. Write an assembly language program to read the status of eight input bits from the logic controller interface and display 'FF' if it is even parity bits otherwise display '00'. Also display number of 1's in the input data.
2. Write an assembly language program to perform the following functions using logic controller interface.
  - i) BCD up-down counter
  - ii) Ring counter
3. Write an assembly language program to display FIRE and HELP alternatively with flickering effects on a 7-segment display interface for a suitable period of time. Ensure a flashing rate that makes it easy to read both the messages.
4. Write an assembly language program to drive a stepper motor interface to rotate the motor by N steps left direction and N steps right direction. Introduce suitable delay between successive steps.
5. Write an assembly language program to scan a 8\*3 keypad for key closure and to store code of key pressed in memory location or display on screen. Also display row and column numbers of key pressed.
6. Write an assembly language program to generate SINE WAVE using DAC(Digital to Analog Converter) interface (the output of the DAC is to be displayed on the CRO).
7. Write an assembly language program to drive an elevator interface in the following way:
  - i) Initially the elevator should be in the ground floor, with all requests in OFF state.
  - ii) When a request is made from a floor, the elevator should move to that floor, wait there for a couple of seconds, and then come down to ground floor and stop. If any requests arrive while elevator is moving up or coming down they should be ignored.

**General Remarks:**

- 1) Lab schedule: 3hrs/week for each student (1 hr tutorial, 2 hrs- program execution).
- 2) Student should complete all the lab assignments.
- 3) Evaluation CIE 50 marks:
  - Lab assignments : 30 marks
  - Lab CIE : 20 marks

**UIS509L: OPERATING SYSTEM LABORATORY**  
**1.5-CREDITS (0-1-2)**

- Note:
- i. Assignments are to be implemented in Linux Environment
  - ii Instructor is expected to supply needed case instances parameters for Implementations.
  - iii. Marks: 50 Marks CIE + 50 SEE

- 1: Write a C/C++ program to implement inter process communication using pipe mechanism.
- 2: Write a C/C++ program to compute average waiting time and average turnaround time for FCFS algorithm. The program should accept arrival time and burst time as input.
- 3: Write a C/C++ program to compute average waiting time and average turnaround time for SJF algorithm. The program should accept arrival time and burst time as input.
- 4: Write a C/C++ program to compute average waiting time and average turnaround time for priority scheduling algorithm. The program should accept arrival time, burst time and priority algorithm as input.
- 5 : Write a C/C++ program to compute average waiting time and average turnaround time for Round robin scheduling algorithm. The program should accept arrival time and burst time as input. Assume suitable time quantum as input.
- 6: Write a C/C++ program to illustrate multithreading concept.
- 7: Write a program to implement Petersons solution to solve critical section problem..
- 8: Write a solution to the bounded buffer problem in C/C++ using POSIX thread library.
- 9: Write a program to detect whether the system is in safe state, the program should accept allocation, max and available matrices. Generate the need matrix.
- 10: Write a program to implement Contiguous memory allocation scheme for a system consisting of several processes.
- 11: Write a C/C++ program that implements FIFO page replacement algorithm.
- 12: Write a C/C++ program to implement LRU page replacement algorithm

**General Remarks:**

- 1) Lab schedule: 3hrs/week for each student (1 hr tutorial, 2 hrs- program execution).
- 2) Student should complete all the lab assignments.
- 3) Evaluation CIE 50 marks:

Lab assignments	: 30 marks
Lab CIE	: 20 marks

## VI SEMESTER

### UIS611H: Management and Entrepreneurship 3 Credits (3-0-0)

#### UNIT I

**INTRODUCTION:** (03 Hours)

Management: Science, Theory and Practice, Managing: Science or Art, The Functions of Managers, The Systems Model of Management, Management and Society, Social Responsibility and Ethics

**PLANNING:** (04 Hours)

The Nature and Purpose of Planning, Types of Plans, Steps in Planning, The Planning Process, Objectives: Management by Objectives, Strategies, Policies and Planning Premises, The strategic Planning Process, Effective Implementation of Strategies, Premising and Forecasting, Decision Making, Importance of Rational Decision making, Limitations of Rational Decision making, Types of Decision Making, Case Studies

**ORGANIZING:** (04 Hours)

The Nature and Purpose of Organizing, Formal and Informal Organization, Organizational Division, The Department, Organization Levels and span of management, The structure and process of Organizing, Effective Organizing, The Departmentation, Matrix Organization, Strategic Business Units, Line Staff Authority and Decentralization, Authority and Power, Line and Staff Concepts, Functional Authority, Decentralization of Authority, Delegation of Authority, Promoting an appropriate Organization Culture, Case Studies

#### UNIT II

**STAFFING:** (04 Hours)

The Systems Approach to HRM, An Overview of the staffing Function, Situational Factors affecting Staffing, Selection Process, Techniques and Instruments, Orienting and Socializing New Employees, Performance Appraisal and Career Strategy, Formulating the Career Strategy, Manager and Organization Development, anager Development Process and Training, Case Studies

**LEADING:** (05 Hours)

Human Factors in Managing, Motivation and Motivators, Motivation Content and Process, Theories, Motivational Techniques, A systems and Contingency Approach to Motivation, Leadership, Ingredients of Leadership, Trait Approaches to Leadership, Leadership Behavior and styles, Contingency Approaches to Leadership

#### UNIT III

**COMMUNICATION:** (05 Hours)

Communication: Introduction, The Communication function in Organization, The Communication Process, Communication in the Enterprise, Barriers and Breakdown to Communication, Towards Effective Communication, Case Studies.

**CONTROLLING:** (05 Hours)

The System and Process of Controlling, Control as a feedback system, Feed Forward Control, Requirements for Effective Controls, Control Techniques and Information Technology, The Budget, Traditional Non-budgetary Control, Information Technology, Productivity and Operations, Direct Control Vs Preventive Control, Case Studies

#### UNIT IV

**ENTREPRENEUR:**

(03 Hours)

Meaning of an Entrepreneur, Evolution of the Concept, Functions of an Entrepreneur, Types of Entrepreneurs, Intrapreneur – an emerging class, Concept of Entrepreneurship, Steps in Entrepreneurial process, Role of Entrepreneurs in Economic Development, Entrepreneurship in India, Entrepreneurship: Barriers

**PREPARATION OF PROJECT:**

(02 Hours)

Meaning of Project and, Project Identification / Project Selection, Project Report: Contents and Formulation, Identification of Business Opportunities, Project Appraisal, Market Feasibility Studies, Technical Feasibility Studies, Financial Feasibility Studies, Social Feasibility Studies.

**INSTITUTIONAL SUPPORT:**

(02 Hours)

Different Schemes: TECSOK, KIADB, KSSIDC, KSIMC, DIC, Single window Agency, MSME, NSIC, SIDBI, KSFC.

**MICRO, SMALL & MEDIUM ENTERPRISES (MSME):**

(03 Hours)

Definition and Characteristics, Need and Rationale, Objectives and Scope, Role of MSME in Economic Development, Advantage of MSME, Steps to start an MSME Government Policy towards MSME, Impact of Liberalisation, Privatisation & Globalization on MSME, Effect of WTO, GATT

**Text Books:**

1. Harold Koontz and Heinz wehrich, Essentials of Management, TMH, 7th Edition.
2. Tripathi and Reddy, Principles of Management, TMH, 4th Edition,
3. Poornima M Charantimath, Entrepreneurship Development – Small Business Enterprises

## **UIS612C: FILE AND INFORMATION STRUCTURES**

### **4CREDITS (4-0-0)**

#### **UNIT I**

INTRODUCTION TO THE DESIGN OF FILE STRUCTURE: The Heart of the file structures Design, A Short History of File Structures Design; Fundamental File Processing Operations: Physical Files and Logical Files, Opening Files, Closing Files, Reading and Writing, Seeking; Secondary Storage: Disks, the organization of disks, estimating capacities space needs, organizing tracks by sector, organizing tracks by blocks, non-data overhead, the cost of a disk access; *CD-ROM*: Physical Organization of CD-ROM, CD-ROM Strengths and Weaknesses. FUNDAMENTAL FILE STRUCTURE CONCEPTS: Field and Record Organization, Buffer Management. Using Classes to Manipulate Buffers, Using Inheritance for Record Buffer Classes, Record access, Header Records.

**13 hours**

#### **UNIT II**

ORGANIZATION OF FILES FOR PERFORMANCE: Reclaiming Space in files, Internal Sorting and Binary Searching, Key sorting . INDEXING: Introduction, A Simple Index for Entry Sequenced File, Object-Oriented support for Indexed, Entry-Sequenced Files of Data Objects, operation required to maintain an indexed file, Class text index file, Indexes that are too large to hold in Memory, Indexing to provide access by Multiple keys, Retrieval Using Combinations of Secondary Keys, Improving the Secondary Index structure, Inverted List.

**13 hours**

#### **UNIT III**

COSEQUENTIAL PROCESSING & THE SORTING OF LARGE FILES: An object oriented model for implementing consequential processes: Matching Names in two lists, Merging two lists, Summary of the consequential processing model, Extension of the model to include multiway merging : A K-way merge algorithm, a selective tree for merging large numbers of lists, A second look at sorting in memory: overlapping processing and I/O: heapsort, Building the heap while reading the file, sorting while writing to the file, Merging as a way of sorting large files on disk: Time for merge sort , Sorting a file that is ten times larger , the cost of in creasing the file size, Hardware based improvements, Decreasing the number of seeks using multiple step merges, Increasing run length s using replacement selection, replacement selection plus multistep merging, Using two disk drives with replacement selection , More drives, More processors, Effects of multiprogramming, A conceptual toolkit for external sorting.

MULTI-LEVEL INDEXING AND B-TREES: Statement of the problem, Indexing with BST, AVL tree, paged binary tree, problem with paged binary trees. Multilevel indexing: A better approach to tree indexes, Working up from the bottom, Example of Creating a B-Tree, An Object-Oriented Representation of B-Trees, B-Tree Methods search, insert & others, Formal Definition of B-Tree Properties, Worst-case Search Depth, Deletion, Merging and Redistribution during insertion.

**13 hours**

#### **UNIT IV**

INDEXED SEQUENTIAL FILE ACCESS AND PREFIX B +TREES :Indexed Sequential Access, Maintaining a Sequence Set, Adding a Simple Index to the Sequence Set, The Content of the Index: Separators Instead of Keys, The Simple Prefix B+ Tree and its maintenance, Index Set Block Size, Internal Structure of Index Set Blocks: A Variable-order

B- Tree, Loading a Simple Prefix B+ Trees. HASHING: Introduction, A Simple Hashing Algorithm, Hashing Functions and Record Distribution, Extra Memory used, Collision resolution by progressive overflow, Sorting more than one record per address Buckets, Making deletions, Other collision resolution techniques. **13 hours**

**TEXT BOOK:**

1. Michael J. Folk, et al: File Structures-An Object Oriented Approach with C++ (Third Edition) Pearson education 2004 (Chapters 1.1, 1.2, Chapter2.1to2.5, Chapter 3.1, 3.4, 3.5, 3.6, Chapter 4.1, 3.9, 4.2, 4.3, 5.1, 5.2.2, Chapter 6.2 to 6.4, Chapter 7.1 to 7.8, Chapter 8.1,8.3 to 8.5,Chapter 9.1 to 9.8, 9.10 to 9.13, Chapter 10.1 to 10.9, Chapter 11.1 to 11.8)

**REFERENCE BOOKS:**

1. Scot Robert Ladd: C++ Components and Algorithms, BPB Publications, 1993.
2. File Structures: Conceptual Toolkit with C++, Venkatesh N.Baitipuli, 1<sup>st</sup> Edition, 2004
3. Introduction Algorithms: Thomas H. Cormen, et.al, 2nd edition, PHI

**UIS603C: COMPUTER NETWORKS**  
**4 CREDITS (4-0-0)**

**UNIT-I**

**ATM Networks:**

Design Goals, Problems, Architecture, Switching, ATM Layers, ATM LANs: ATM LAN Architecture, LAN Emulation.

**7 Hrs**

**Network Layer: Logical Addressing:**

IPv4 Addresses: Address Space, Notation, Classful Addressing, Classless Addressing, IPv6 Addresses: Structure, Address Space

**5 Hrs**

**UNIT-II**

**Network Layer: Internet Protocol:**

Internetworking, IPv4, IPv6, Transition from IPv4 to IPv6. Address Mapping, Error Reporting & Multicasting: Address Mapping, ICMP, IGMP. Delivery, Forwarding & Routing: Delivery, Forwarding, Unicast Routing Protocols: Optimization, Intra- and Interdomain Routing, Distance vector routing, Link state routing, Multicast routing protocols: unicast, multicast and broadcast, multicast routing, routing protocols: MOSPF, DVMRP, Flooding, RPF, RPB, CBT, PIM.

**14 Hrs**

**UNIT-III**

**Transport Layer:**

Process-to-Process Delivery: UDP, TCP, SCTP. Congestion Control

**6 Hrs**

**Quality of Service:** Flow characteristics, Flow classes, Techniques to improve QoS, Integrated Services, Differentiated Services, QoS in Switched Networks.

**6 Hrs**

**UNIT-IV**

**Application Layer:**

**Domain Name System:** Name Space, Domain Name Space, Distribution of Name Space, DNS In The Internet, Resolution, DNS Messages, Types of records, Registrars, Dynamic Domain Name System, Encapsulation. Remote Logging, Electronic Mail, File Transfer. WWW and HTTP: Architecture, Web Documents, HTTP

**Network Management:** Network Management System, SNMP.

**14 Hrs**

**TEXT BOOK:**

1. **Data Communications and Networking** - Behrouz A. Forouzan, 4th Edition, Tata McGraw-Hill, 2006.

**REFERENCES BOOKS:**

1. **Communication Networks –Fundamental Concepts and Key Architectures** - Alberto Leon-Garcia and Indra Widjaja, 2<sup>nd</sup> Edition, Tata McGraw-Hill, 2004.

2. **Computer and Communication Networks** - Nader F. Mir, Pearson Education, 2007.
3. **Data and Computer Communication** - William Stallings, 8<sup>th</sup> Edition, Pearson Education, 2007.
4. **Computer Networks – A Systems Approach** - Larry L. Peterson and Bruce S. David, 4th Edition, Elsevier, 2007.
5. **Introduction to Data Communications and Networking** – Wayne Tomasi, Pearson Education, 2005.

**UIS604C: UNIX SYSTEM PROGRAMMING**  
**4 CREDITS (4-0-0)**

**UNIT I**

**1. INTRODUCTION** **4 hrs**

UNIX and ANSI Standards: The ANSI C Standard, The ANSI/ISO C++ Standards, Differences between ANSI C and C++, The POSIX Standard, The POSIX.1 FIPS Standard, The X/Open Standards.

UNIX and POSIX APIs: The POSIX APIs, The UNIX and POSIX Development Environment, API Common Characteristics

File Types, The UNIX and POSIX File system, The UNIX and POSIX File Attributes, Inodes in UNIX System v, UNIX Kernel Support for Files, Hard and Symbolic links

**2. UNIX and POSIX API'S** **9 hrs**

General File API's, File and record locking, Directory file API's.

**UNIT II**

**3. UNIX PROCESSES** **4 hrs**

Unix Kernel Support for Processes, Process API's, Process Attributes, Changing Process Attributes,

**4. UNIX SIGNALS** **9 hrs**

The Unix Kernel Support for Signals, signal, Signal mask, sigaction, the SIGCHLD Signal and the waitpid function, The sigsetjmp and siglongjmp Functions, kill alarm, Interval Timers and POSIX.1 Timers

**UNIT III**

**5. INTER-PROCESS COMMUNICATION** **13 hrs**

**POSIX.1b IPC** methods, The UNIX system V IPC methods, Unix system V messages, POSIX.1b messages, Unix system V semaphores, Unix system V shared memory, Memory mapped I/O.

**UNIT IV**

**6. SOCKETS and TL1** **13 hrs**

Sockets, A stream socket example, Client –server message handling example, TL1, TL1 class, Client-server message example, Datagram example

**TEXT BOOK:**

1. Terrence Chan: Unix System Programming Using C++, Printice Hall India, 2003

**REFERENCE BOOK:**

1. W. Richard Stevens; Advanced Programming in the Unix Environment, Addison Wesley/PHI

**UIS608L: UNIX SYSTEM PROGRAMMING LAB**  
**1.5 CREDITS (0-1-2)**

- 1 Write a C program that creates a child process to read commands from the standard input and execute them (a minimal implementation of a shell like program). You can assume that no arguments will be passed to the commands to be executed.
- 2 . Write C program filecopy that asks for an input path and output path and then copies the input to output. Input and output are ordinary files. Make copy operation faster.
3. Write a C program print out a directory tree. Prompt the user and accept name of the starting directory, print the name of the starting directory, read the directory and ignore everything except directories, print the names of any directories encountered along with any directories that they contain, indent each level of directory two spaces.
4. Write a C program to implement stat() function
5. Write a C program sv, that copies file to diir/file1, dir/file2.....etc except that when a target file is newer than its source file, no copy is made and a warning is printed. To avoid making multiple copies of linked files, sv does not allow /'s in any of the source files names
6. Write a C program to do the following Using fork() create a child process. The child process prints its own process id and id of its parent and then exits. The parent process waits for its child to finish (by executing wait()) and prints its own process id and the id of its child process and then exits.
7. Write a C program will do the following: The parent process calls pipe to allocate a pipe device file. It then calls fork to create a child process. Both the parent and the child can access the pipe. The child process is designated as the sender of the message to the parent.
8. Write a C program to implement popen() and pclose() system calls.
- 9 Write a C program that will do the following: The process signal mask is set to SIGTERM signal. The process then defines signal handler for the SIGINT signal and also specifies that the SIGSEGV signal is to be blocked when the process is handling the SIGINT signal. The process then suspends its execution via the pause API. Use sigaction() API to handle the signal.
- 10 Write a C program to implement kill command using kill API
11. Program to illustrate IPC
12. Program to illustrate Sockets.

**UIS609L: DATABASE APPLICATIONS LABORATORY**  
**1.5 CREDITS (0-1-2)**

1. Create the tables for the following with suitable assumptions:

**Employee and Salary**

- a. Create the above tables by properly specifying the primary keys and the foreign keys
- b. Generate the queries to do the following:
  - (i) To sum the salary of each employee and sort it in descending order on the sum of basic.
  - (ii) To list the employees who earn less than the average salary.
  - (iii) To list the names of employees and salary details, whose basic is less than the average salary.

2. Consider the Insurance database given below. The primary keys are underlined and the data types are specified:

PERSON (Driver – id #: string, Name: string, Address: string)

CAR (Regno: string, Model: string, Year: int)

ACCIDENT (Report-number: int, Accd-Date: date, Location: string)

**OWNS (Driver-id #: string, Regno:string)**

PARTICIPATED (Driver-id: string, Regno:string, Report-Number: int, Damage Amount: int)

- (i) Create the above tables by properly specifying the primary keys and the foreign keys.
- (ii) Demonstrate how you
  - a. Update the damage amount to 25000 for the car with a specific Regno in the ACCIDENT table with report number 12.
  - b. Add a new accident to the database.
- (iii) Find the total number of people who owned cars that were involved in accidents in 2008.
- (iv) Find the number of accidents in which cars belonging to a specific model were involved.

3. Consider the following database of student enrollment in courses & books adopted for each course:

STUDENT (Regno: string, Name: string, Major: string, Bdate:date)

COURSE (Course #:int, Cname:string, Dept:string)

**ENROLL ( Regno:string, Course#:int, Sem:int, Marks:int)**

BOOK \_ ADOPTION (Course# :int, Sem:int, Book-ISBN:int)

TEXT (Book-ISBN:int, Book-Title:string, Publisher:string, Author:string)

- (i) Create the above tables by properly specifying the primary keys and the foreign keys.
- (ii) Demonstrate how you add a new textbook to the database and make this book be adopted by some department.**
- (iii) Produce a list of textbooks (include Course #, Book-ISBN, Book-Title) in the alphabetical order for courses offered by the 'CS' department that use more than two books.
- (iv) List any department that has *all* its adopted books published by a specific publisher.

4. A database management system is to be created for a study centre to keep track of the students and the assignments records of the students. The database records the date of submission of assignments, the date of evolution, the date of viva, the date of declaration of results, who evaluated the assignment, and the mark list description in which the assignment marks were sent to regional office. The database also maintains the details of the evaluators. Perform the following activities for the description as given above.

- a) Design the database with suitable integrity constraints and create the database
- b) Write the following queries using SQL:
  - (i) Find the list of the students who have not submitted even a single assignment
  - (ii) Find the details of the evaluators by whom average marks awarded are more than 70%
  - (iii) Find the students who have passed more than 5 assignments
  - (iv) Find the list of students who have not appeared in VIVA. Make and state suitable assumptions, if any.

5. Design a database for maintaining the details of shows and ticketing for the shows of multiples. Now perform the following activities for the system:

- a) Create the database
- b) Write the following queries using SQL
  - (i) Find the details of the movies whose shows are not yet full
  - (ii) Find the details of the movies that had been screened at least one year earlier to
  - (iii) Find the names of those that have an overall state of 60% of capacity on all days of screening

**6. Consider the following database for a banking enterprise:**

BRANCH(Branch-name:string, Branch-City:string, Assets:real)

ACCOUNT(Accno:int, Branch-Name:string, Balance:real)

DEPOSITOR(Customer-Name:string, Accno:int)

CUSTOMER(Customer-Name:string, Customer-Street:string, Customercity:string)

LOAN(Loan-Number:int, Branch-Name:string, Amount:real)

BORROWER(Customer-Name:string, Loan-Number:int)

- (i) Create the above tables by properly specifying the primary keys and the foreign keys
- (ii) Find all the customers who have at least two accounts at the Main branch.
- (iii) Find all the customers who have an account at all the branches located in a specific city.
- (iv) Demonstrate how you delete tuples in ACCOUNT relation at every branch located in a specific city.
- (v) Find all loan numbers for loans made at the specific branch with loan amounts greater than Rs1200.
- (vi) Find all loan numbers for loans with loan amounts between Rs 90,000 and Rs100000.

**Instructions:**

1. The exercises are to be solved in an RDBMS environment like Oracle.
2. Enter at least five tuples for each relation.
3. Create suitable front end for querying and displaying the results.
4. Suitable tuples have to be entered so that queries are executed correctly.
5. Front end may be created using either VB or any other similar tool.
6. Generate suitable reports.
7. The student need not create the front end in the examination. The results of the queries may be displayed directly.
8. Relevant queries other than the ones listed along with the exercises may also be asked in the examination.
9. Questions must be asked based on lots.
10. Each group should be of maximum 4 members.

11. Group tasks will be assigned based on lots
12. 10(CIE) marks will be given to group task implementation based on demonstration.
13. In the SEE examinations the viva marks are based on knowledge of the subject and demonstration of group task.

**Group Tasks (not limited to this list):**

1. Library Management
2. Hotel Management
3. Attendance Report
4. Bus/Rail/Air Ticket Reservation Management
5. Employee Record System
6. Insurance Database
7. Order processing Database
8. Bank Database
9. Application for gold selling scheme.
10. Billing system for general stores.
11. Information management system.

**General Remarks:**

- 1) Lab schedule: 3hrs/week for each student (1 hr tutorial, 2 hrs- program execution).
- 2) Student should complete all the lab assignments.
- 3) One group task adequate per student batch (2 students/batch).
- 4) Each student should demonstrate his/her group work individually.
- 5) Evaluation CIE 50 marks:

Lab assignments	: 30 marks
Group work	: 10 marks
Lab CIE	: 10 marks

## UIS610L: FILE AND INFORMATION STRUCTURES LAB

### 1.5 CREDITS (0-1-2)

1. Write a C++ program to write & read (display) STUDENT objects where each object has data members USN, Name, Address, Sem and Branch, with delimited records & delimited field structures. Implement pack ( ), unpack ( ) methods.
2. Write a C++ program to read and write EMPLOYEE objects where each object has data members employee – ID, Name & Designation, with variable length records (using length indicator) & each field begin with length indicator. Implement pack ( ) & unpack ( ) methods.
3. Write a C++ program to read and write CUSTOMER objects where each object has data members Customer – ID, Name & Address, with fixed length records & the fields delimited by ' | '. Implement pack ( ) , unpack ( ) methods.
4. Write a C++ program to write TEACHER objects where each object has teacher Name, Address and Designation, with fixed length records using any suitable field structure & to read from this file a teacher record using RRN.
5. Write a C++ program to write ITEM objects where each object has item – Code, Name & Type, with variable length record using any suitable record structure. Sort the records using item code as key applying key sorting technique.
6. Write a C++ program to write PATIENT objects where each object has Patient Number, Name & Disease, with variable length record using any suitable record structure. Search a desired patient using binary search technique by patient number as key.
7. Write a C++ program to read two lists of names & then match the names in the two lists using co sequential match operation. Output the names common to both the lists.
8. Write a C++ program to read K-lists of names & merge them using K- way merge algorithm with  $K = 4$ .
9. Write a C++ program to read a list of name & sort them using Heap Sort method.
10. Write a C++ program to write ACCOUNT objects where each objects has date number, Account Number., Account holder Name, Account Type, with variable length record structure & to delete a record by account number.

# MINI PROJECTS

## **PART I: The following specifications are common to all projects.**

Design a class called BOOK. Each object of this class represents information about a single book. Members should be included for book Number, Title, Author and Publication. Provide methods to store objects as records in file using variable length record structure (delimited record, delimited field structure). Add methods to read the member values to the output stream. Add pack ( ) & unpack ( ) methods to class book.

## **PART II: One of the following mini projects is to be completed by each student batch.**

**PROJECT 1:** Develop an index for the book record file with Book Number as the key & Title as the secondary key. Write driver program to create an index for the file created in PART I & prompt for the book number to display book record. Write the program so that if a secondary key value is entered by the user & if there are multiple matching records, then all the records are displayed.

**PROJECT 2:** Develop secondary index using Book Title as key for the file created in PART I. Select another secondary key (i.e. Author) and demonstrate the retrieval using combination of secondary keys. Use inverted list structures to rearrange the secondary key index when new insertion is to be done.

**PROJECT 3:** Develop an application that produces books transcripts using cosequential process & master transaction process. For each book record (Master) print the book information & list of departments (transaction) can refer the book & implement two-way merge.

**PROJECT 4 :** Create a B – tree index to the data file created in PART I, with Book Number as key. Demonstrate the retrieval information based on book number & prints all objects that match it. Invoke a method to indicate the level of collapse & level of splitting.

**PROJRCT 5:** Develop a hashed index of book record file with Book Number as the key. Write a driver program to prompt for book number to retrieve book record. Use any collision resolution techniques.

## **GENERAL REMARKS:**

1. One mini project adequate per student batch (Two students per batch).
2. Each student should demonstrate his/her project individually.
3. Evaluation CIE 50 marks:
  - a. Demonstration of the project and related viva : 10
  - b. Lab Assignment execution and related viva : 30
  - c. Lab CIE : 10

## VII Semester

### UIS701C: Web Technologies

3 Credits: (0-0-3)

#### UNIT – 1

**FUNDAMENTALS OF WEB, XHTML – 1:** Internet, WWW, Web Browsers, and Web Servers; URLs; Security; The Web Programmers Toolbox XHTML: Origins and evolution of HTML and XHTML; Basic syntax; Standard XHTML document structure; Basic text markup. **XHTML – 2:** Images; Hypertext Links; Lists; Tables; Forms; Frames; Syntactic differences between HTML and XHTML.

**CSS:** Introduction; Levels of style sheets; Style specification formats; Selector forms; Property value forms; Font properties; List properties; Color; Alignment of text; The Box model; Background images; The <span> and <div> tags; Conflict resolution. **10 Hours**

#### UNIT - 2

**JAVASCRIPT:** Overview of JavaScript; Object orientation and JavaScript; General syntactic characteristics; Primitives, operations, and expressions; Screen output and keyboard input; Control statements; Object creation and modification; Arrays; Functions; Constructor; Pattern matching using regular expressions; Errors in scripts;

**JAVASCRIPT AND HTML DOCUMENTS:** The Javascript execution environment; The Document Object Model; Element access in Javascript Events and event handling; Handling events from the Body elements, Button elements, Text box and Password elements; The DOM 2 event model; The navigator object; DOM tree traversal and modification

**10 Hours**

#### UNIT-3

**DYNAMIC DOCUMENTS WITH JAVASCRIPT:** Introduction to dynamic documents; Positioning elements; Moving elements; Element visibility; Changing colors and fonts; Dynamic content; Stacking elements; Locating the mouse cursor; Reacting to a mouse click; Slow movement of elements; Dragging and dropping elements.

**XML:** Introduction; Syntax; Document structure; Document Type definitions; Namespaces; XML schemas; Displaying raw XML documents; Displaying XML documents with CSS; XSLT style sheets; XML processors; Web services.

**10 Hours**

#### UNIT - 4

**PERL, CGI PROGRAMMING:** Origins and uses of Perl; Scalars and their operations; Assignment statements and simple input and output; Control statements; Fundamentals of arrays; Hashes; References; Functions; Pattern matching; File input and output; Examples. The Common Gateway interface; CGI linkage; Query string format; CGI.pm module; A survey example; Cookies

**10 Hours**

#### TEXT BOOK:

1. **Programming the World Wide Web** - Robert W. Sebesta, 4<sup>th</sup> Edition, Pearson Education, 2008.

#### REFERENCE BOOKS:

1. **Internet & World Wide Web How to program** - M. Deitel, P.J.Deitel, A. B. Goldberg, 3rd Edition, Pearson Education / PHI, 2004.

2. **Web Programming Building Internet Applications** - Chris Bates, 3rd Edition, Wiley India, 2006.

3. **The Web Warrior Guide to Web Programming** - Xue Bai et al, Thomson, 2003.

# UIS702C: Object Oriented Modeling and Design

## UNIT – 1

**INTRODUCTION, MODELING CONCEPTS, CLASS MODELING:** What is Object Orientation? What is OO development? OO themes; Evidence for usefulness of OO development; OO modeling history.

**Modeling as Design Technique:** Modeling; abstraction; the three models. Class Modeling: Object and class concepts; Link and associations concepts; Generalization and inheritance; A sample class model; Navigation of class models; Practical tips.

**Advanced Class Modeling:** Advanced object and class concepts; Association ends; N-ary associations; Aggregation; Abstract classes; Multiple inheritance; Metadata; Reification; Constraints; Derived data; Packages; Practical tips.

**12 Hours**

## UNIT - II

**STATE MODELING, ADVANCED STATE MODELING, INTERACTION MODELING, PROCESS OVERVIEW:**

**State Modeling:** Events, States, Transitions and Conditions; State diagrams; State diagram behavior; Practical tips.

**Advanced State Modeling:** Nested state diagrams; Nested states; Signal generalization; Concurrency; A sample state model; Relation of class and state models; Practical tips.

**Interaction Modeling:** Use case models; Sequence models; Activity models. Use case relationships; Procedural sequence models; Special constructs for activity models.

**Process Overview:** Development stages; Development life cycle.

**13 Hours**

## UNIT – III

**SYSTEM CONCEPTION, DOMAIN ANALYSIS, APPLICATION ANALYSIS, SYSTEM DESIGN - 1:**

**System Conception:** Devising a system concept; Elaborating a concept; Preparing a problem statement.

**Domain Analysis:** Overview of analysis; Domain class model; Domain state model; Domain interaction model; Iterating the analysis.

**Application Analysis:** Application interaction model; Application class model; Application state model; Adding operations.

**System Design -1:** Overview of system design; Estimating performance; Making a reuse plan; Breaking a system in to sub-systems; Identifying concurrency; Allocation of sub-systems; Management of data storage; Handling global resources; Choosing a software control strategy.

**13 Hours**

## UNIT IV

### **SYSTEM DESIGN - 2, CLASS DESIGN, IMPLEMENTATION MODELING AND DESIGN PATTERNS:**

**System Design -2:** Handling boundary conditions; Setting the trade-off priorities; Common architectural styles; Architecture of the ATM system as the example.

**Class Design:** Overview of class design; Bridging the gap; Realizing use cases; Designing algorithms; Recursing downwards, Refactoring; Design optimization; Reification of behavior; Adjustment of inheritance; Organizing a class design; ATM example.

**Implementation Modeling:** Overview of implementation; Fine-tuning classes; Fine-tuning generalizations; Realizing associations; Testing.

**Design patterns:** What is a pattern and what makes a pattern? Pattern categories; Relationships between patterns; Pattern description.

Communication Patterns: Forwarder-Receiver; Client-Dispatcher-Server; Publisher-Subscriber.

**14 Hours**

### **TEXT BOOKS:**

1. **Object-Oriented Modeling and Design with UML** - Michael Blaha, James Rumbaugh, 2<sup>nd</sup> Edition, Pearson Education, 2005.
2. **Pattern-Oriented Software Architecture A System of Patterns, Volume 1** - Frank Buschmann, Regine Meunier, Hans Rohnert, Peter Sommerlad, Michael Stal John Wiley and Sons, 2006.

### **REFERENCE BOOKS:**

1. **Object Oriented Systems Development** - Ali Bahrani, McGraw-Hill, 1999.
2. **Object-Oriented Analysis and Design with Applications** - Grady Booch et al, 3<sup>rd</sup> Edition, Pearson, 2007.
3. **Practical Object-Oriented Design with UML** - Mark Priestley, 2<sup>nd</sup> Edition, Tata McGraw-Hill, 2003.
4. **Object-Oriented Design with UML and JAVA** - K. Barclay, J. Savage, Elsevier, 2008.
5. **The Unified Modeling Language User Guide** - Booch, G., Rumbaugh, J., and Jacobson, I, 2<sup>nd</sup> Edition, Pearson, 2005.
6. **Design Patterns- Elements of Reusable Object-Oriented Software** - E. Gamma, R. Helm, R. Johnson, J. Vlissides, Addison-Wesley, 1995.
7. **Object-Oriented Systems Analysis and Design Using UML** - Simon Bennett, Steve McRobb and Ray Farmer, 2<sup>nd</sup> Edition, Tata McGraw-Hill, 2002

**UIS713C: MANAGEMENT INFORMATION SYSTEMS**  
**3 CREDITS (3-0-0)**

**UNIT 1**

**10 Hrs**

**Foundations of Information System in Business**

**Foundation Concepts:** Information System in business, Introduction, IT, What you need to know, an information system frame work for business professionals, Trends in Information System, the role of e- business in business.

**Types of Information System:** Operation support system, management support system, other classifications.

**Components of Information System:** what is system, Feedback & Control, other systems characteristics, Information System resources, Information System activities.

**Competing with Information technology:** using Information technology for strategic advantage, strategic uses of Information technology, Reengineering business process, through knowledge management, becoming an agile company, creating a virtual company, building a knowledge creating company , knowledge management system.

**UNIT 2**

**10 Hrs**

**Electronic business system:** Enterprises business system, Introduction, cross functional enterprise application, enterprise application architecture, enterprise application integration.

**Transaction processing system:** the transaction processing cycle, enterprise collaboration system, tools for enterprise collaboration.

**Functional business system:** Introduction, marketing system, interactive marketing, targeting marketing; application integration , sales force automation, manufacturing system, human resource system, accounting system, financial management.

**Enterprise business system:** Enterprise resource planning, introduction, what is ERP, benefits 7 challenges of ERP.

**UNIT 3**

**10 Hrs**

**Supply chain management:** Introduction, what is Supply chain management, EDI, the role of Supply chain management, benefits & challenges of Supply chain management.

**Decision support system:** Decision supporting business, introduction, Decision support system, management Information System, online analysis processing, using DSS, executive Information system, knowledge management system.

**AI technology in business:** business & AI, ES, developing ES, neural network, fuzzy logic system, generic algorithms, virtual reality.

**UNIT 4**

**10 Hrs**

**Development process:** Developing business or IT solutions, developing business system, Information system development, System approach, system thinking,

**System development cycle:** prototyping, starting the system development process, system analysis, system design, end user development, management challenges.

**Security & ethical challenges:** Introduction, Ethical responsibility of business processions, computer crime, privacy issues, other challenges.

**Security management of IT:** Tools of security management, inter network security defenses: encryption, firewalls, denial of service defenses, email monitoring, virus defenses, other security measures.

**TEXT BOOK:**

1. **James A. O'Brien:** Management Information Systems (7<sup>th</sup> edition) Tata McGraw Hill

**REFERENCE BOOKS:**

1. **W.S. Jawadekar:** Management Information Systems, Tata McGraw Hill 1998
2. **Laudon and Laudon:** Management Information Systems: Organization and Technology. (4<sup>th</sup> edition), Pearson Education/Prentice Hall India 1999.

**UIS707L: Object Oriented Modeling & Design Laboratory**  
**1 CREDITS (0-0-2)**

1. Design the Library system: Identify the use cases of the system. (Suggestive use cases: borrow book, return books, read newspapers, reference, and digital library). Develop the use case diagram, Packages and documentation for the same. Preferable use of uses & Extends relationships expected.
2. Design the Examination system: Identify the use cases. (Suggestive use cases:– Form filling, Get Hall Ticket, Write exam, get result Verify Hall Ticket) Develop the use case diagram, Packages and documentation for the same. Preferable use of uses & Extends relationships expected.
3. Analyze and design the system for ATM Transaction: Identify the use cases. (Suggestive use cases: Transaction, Approval process, Invalid PIN, Deposit Amount, Deposit savings, Deposit checking, withdraw Amount, withdraw checking, saving, withdraw saving denied, checking Transaction History, saving Transaction History). Package, documentation Develop the use case diagram, Packages and documentation for the same. Draw the essential class diagrams.
4. Analyze and design the system for Electronics voting system (The actors are presiding officer, 1<sup>st</sup> polling officer, 2<sup>nd</sup> polling officer, voters list, Election officer, voter candidate, EVM ID; Processes: Vote counting, and announcement of results). Develop the use case diagram, Packages and documentation for the same. Draw the essential sequence diagrams, activity diagram and state chart diagrams.
5. Analyze and design the system for Employee reference. (The Process HR Manager contacts Employees of his company and HR manager of other company to publicize about the vacancy. The person, who has referred the right candidate, will be given bonus. Interview, Short-listing, selection list announcement, Bonus for referred employees are all parts of the process.). Develop the use case diagram, sequence diagrams and state chart diagrams.
6. Analyze and design the system for Vehicle Purchase, registration and licensing Systems. Develop the use case diagram, sequence diagrams, activity diagrams and packages.
7. Develop State transition diagrams for
  - a. Telephone line System
  - b. Nested State diagram for vehicle transmission states
- 8 Analyze and design the system for autonomous education system. (Classes: students Teacher, courses, subjects, core, Electives, Labs). Develop the class diagrams, sequence diagrams and packages.
9. Analyze and design the system for Results section of autonomy, mainly responsible for CGPA, SGPA Calculation, and Grade card generation. Develop the class diagrams, use case diagram and packages.
10. Analyze and design the system for Railway reservation, which includes the details of Boarding station, classes of reservation, fare, TimeTable, concessions, No. of Seats, Seat No, State of reservation (confirmed, waiting, RAC). Develop the class diagrams, packages, and use case diagrams.
11. Analyze and design the system for Income Tax assessment. Develop the class diagrams, use case and state chart diagrams.

**General Remarks:**

- 1) Lab schedule: 3hrs/week for each student (1 hr tutorial, 2 hrs- program execution).
- 2) Student should complete all the lab assignments.
- 3) Evaluation CIE 50 marks:
  - Lab assignments : 30 marks
  - Lab CIE : 20 marks

## UIS708L: WEB PROGRAMMING LABORATORY

### 1.5 CREDITS (0-0-3)

1. Develop and demonstrate a XHTML document that illustrates the use external style sheet, ordered list, table, borders, padding, color, and the <span> tag.
2. Develop and demonstrate a XHTML file that includes Javascript script for the following problems:
  - a) Input: A number n obtained using prompt Output: The first n Fibonacci numbers
  - b) Input: A number n obtained using prompt Output: A table of numbers from 1 to n and their squares using alert
3. Develop and demonstrate a XHTML file that includes Javascript script that uses functions for the following problems:
  - a) Parameter: A string Output: The position in the string of the left-most vowel
  - b) Parameter: A number Output: The number with its digits in the reverse order
4. a) Develop and demonstrate, using Javascript script, a XHTML document that collects the USN ( the valid format is: A digit from 1 to 4 followed by two upper-case characters followed by two digits followed by two upper-case characters followed by three digits; no embedded spaces allowed) of the user. Event handler must be included for the form element that collects this information to validate the input. Messages in the alert windows must be produced when errors are detected.  
b) Modify the above program to get the current semester also (restricted to be a number from 1 to 8)
5. a) Develop and demonstrate, using Javascript script, a XHTML document that contains three short paragraphs of text, stacked on top of each other, with only enough of each showing so that the mouse cursor can be placed over some part of them. When the cursor is placed over the exposed part of any paragraph, it should rise to the top to become completely visible.  
b) Modify the above document so that when a paragraph is moved from the top stacking position, it returns to its original position rather than to the bottom.
6. a) Design an XML document to store information about a student in an engineering college affiliated to VTU. The information must include USN, Name, Name of the College, Branch, Year of Joining, and e-mail id. Make up sample data for 3 students. Create a CSS style sheet and use it to display the document.  
b) Create an XSLT style sheet for one student element of the above document and use it to create a display of that element.
7. a) Write a Perl program to display various Server Information like Server Name, Server Software, Server protocol, CGI Revision etc.  
b) Write a Perl program to accept UNIX command from a HTML form and to display the output of the command executed.
8. a) Write a Perl program to accept the User Name and display a greeting message randomly chosen from a list of 4 greeting messages.  
b) Write a Perl program to keep track of the number of visitors visiting the web page and to display this count of visitors, with proper headings.

9. Write a Perl program to display a digital clock which displays the current time of the server.

10. Write a Perl program to insert name and age information entered by the user into a table created using MySQL and to display the current contents of this table.

11.a) Write a perl program to store current date-time in a COOKIE and display the 'Last visited on' date-time on the web page upon reopening of the same page.

b) Write a perl program to store page views count in SESSION, to increment the count on each refresh, and to show the count on web page.

12. Create a XHTML form with Name, Address Line 1, Address Line 2, and E-mail text fields. On submitting, store the values in MySQL table. Retrieve and display the data based on Name.

**General Remarks:**

1. Lab schedule: 3hrs/week for each student (1 hr tutorial, 2 hrs- program execution).

2. Student should complete all the lab assignments.

3. Evaluation CIE 50 marks:

Lab assignments : 30 marks

Lab CIE : 20 marks

**UIS709L: NETWORKS LABORATORY**  
**1.5 CREDITS (0-0-3)**

**PART - A**  
**SIMULATION EXERCISES**

**The following experiments shall be conducted using either NCTUns or any other suitable simulator.**

1. Simulate a three nodes point – to – point network with duplex links between them. Set the queue size and vary the bandwidth and find the number of packets dropped.
2. Simulate a four node point-to-point network with the links connected as follows:  
n0 – n2, n1 – n2 and n2 – n3. Apply TCP agent between n0-n3 and UDP between n1-n3. Apply relevant applications over TCP and UDP agents changing the parameter and determine the number of packets sent by TCP / UDP.
3. Simulate the different types of Internet traffic such as FTP and TELNET over a network and analyze the throughput.
4. Simulate the transmission of ping messages over a network topology consisting of 6 nodes and find the number of packets dropped due to congestion.
5. Simulate an Ethernet LAN using n nodes (6-10), change error rate and data rate and compare throughput.
6. Simulate an Ethernet LAN using n nodes and set multiple traffic nodes and determine collision across different nodes.
7. Simulate an Ethernet LAN using n nodes and set multiple traffic nodes and plot congestion window for different source / destination.
8. Simulate simple ESS and with transmitting nodes in wire-less LAN by simulation and determine the performance with respect to transmission of packets.

**PART - B**

**Implement the following in C/C++:**

1. Write a program for error detecting code using CRC-CCITT (16- bits).
2. Write a program for frame sorting technique used in buffers.
3. Write a program for distance vector algorithm to find suitable path for transmission.
4. Using TCP/IP sockets, write a client – server program to make the client send the file name and to make the server send back the contents of the requested file if present.
5. Implement the above program using as message queues or FIFOs as IPC channels.
6. Write a program for simple RSA algorithm to encrypt and decrypt the data.
7. Write a program for Hamming code generation for error diction and correction.
8. Write a program for congestion control using leaky bucket algorithm

**General Remarks:**

1. Lab schedule: 3hrs/week for each student (1 hr tutorial, 2 hrs- program execution).
2. Student should complete all the lab assignments.
3. Evaluation CIE 50 marks:
  - Lab assignments : 30 marks
  - Lab CIE : 20 marks

# **Project Work**

## **Tasks to be carried out:**

1. Identify the team members and the Guide
2. Identify the Problem
3. Literature survey
  - a. Domain knowledge
  - b. Identify the issues/challenges in the domain area
4. Define the problem statement
5. For application oriented projects (non-research projects), students will study and give presentation on technology (tool/language/simulator etc) related to their project work.
6. For research oriented projects,
  - a. identify the relevant papers (one or two) w.r.t. their problem definition.
  - b. Presentation on the critical analysis of the selected paper(s).
7. Presentation of Pre-project demo .
8. Mid – term demo.
9. Final demo.
10. Preparation of paper on project work (optional).
11. Report preparation and submission.

# UIS710P: Project Phase – I

1 Credit (0-0-0)

## Evaluation

- 1) Internal : 50  
Guide will allocate the marks.
- 2) External : 50 (Evaluation will be done by Project Evaluation Committee)  
Presentation on works done:
  - a. Identify the Problem -
  - b. Literature survey -
    - I. Domain knowledge
    - II. Identify the issues/challenges in the domain area
  - c. Define the problem statement -
  - d. For application/technology oriented projects (non-research projects), students will study and give presentation on technology (tool/language/simulator etc) related to their project work -

**Or**

For research oriented projects,  
Identify the relevant papers (one or two) w.r.t. their problem definition  
Presentation on the critical analysis of the selected paper (s)

## VIII Semester

### UIS803P: Project Phase – II

18 Credits (0-0-0)

#### **Evaluation Criteria:**

I. Internal (CIE) : 50

Each project batch has to give three demonstration of their project work done before the Project Evaluation Committee. Guide along with committee will award marks.

II. External (SEE) : 50

**SEE Marks** – The Common Project Evaluation Committee (PEC) will conduct the examination and award marks.

**UIS804P: Seminar**  
**1 Credit (0-0-0)**

Student has to select current topics for seminar in consultation with guide.

**Evaluation Criteria:**

**CIE: 50 Marks** - CIE shall be based on a presentation by the student before the guide. The guide shall award marks.

**SEE 50 Marks-** SEE shall be based on a presentation before the seminar evaluation committee at the end of the semester. The committee shall award marks.

## LIST OF ELECTIVES

<b>Sl. No</b>	<b>Subject Code</b>	<b>Subject Name</b>
01	UIS001E	ADVANCED COMPUTER ARCHITECTURES
02	UIS002E	ADVANCED GRAPH THEORY
03	UIS003E	ARTIFICIAL INTELLIGENCE
04	UIS004E	CRYPTOGRAPHY AND NETWORK SECURITY
05	UIS005E	DIGITAL IMAGE PROCESSING
06	UIS006E	DISTRIBUTED OPERATING SYSTEMS
07	UIS007E	FUZZY LOGIC
08	UIS008E	GENETIC ALGORITHMS
09	UIS009E	NETWORK MANAGEMENT SYSTEMS
10	UIS010E	PATTERN RECOGNITION
11	UIS011E	REAL TIME OPERATING SYSTEMS
12	UIS012E	SOFTWARE ARCHITECTURE
13	UIS013E	WEB SERVICES AND SERVICE ORIENTED ARCHITECTURE
14	UIS014E	SYSTEM MODELING AND SIMULATION
15	UIS015E	JAVA AND J2EE
16	UIS016E	ADVANCED DBMS.
17	UIS017E	ARTIFICIAL INTELLIGENCE AND EXPERT SYSTEMS
18	UIS018E	CLIENT SERVER COMPUTING
19	UIS019E	DATA MINING
20	UIS020E	DISTRIBUTED DATA BASES
21	UIS021E	EMBEDDED SYSTEMS
22	UIS022E	LINUX KERNEL PROGRAMMING
23	UIS023E	NEURAL NETWORKS
24	UIS024E	PRINCIPLES OF PROGRAMMING LANGAUAGES
25	UIS025E	SOFT COMPUTING
26	UIS026E	WEB PROGRAMMING
27	UIS027E	MOBILE COMPUTING
28	UIS028E	COMPILER DESIGN
29	UIS029E	C# PROGRAMMING AND .NET
30	UIS030E	SOFTWARE TESTING
31	UIS031E	STORAGE TECHNOLOGY

**UIS001E: ADVANCED COMPUTER ARCHITECTURES**  
**3 CREDITS (3-0-0)**

**UNIT - 1**

**FUNDAMENTALS OF COMPUTER DESIGN:** Introduction; Classes of computers; Defining computer architecture; Trends in Technology, power in Integrated Circuits and cost; Dependability; Measuring, reporting and summarizing Performance; Quantitative Principles of computer design; **PIPELINING:** Introduction; Pipeline hazards.

**10 hours**

**UNIT - 2**

**IMPLEMENTATION OF PIPELINE;** Issues pipelining implementation.

**INSTRUCTION –LEVEL PARALLELISM – 1:** ILP: Concepts and challenges; Basic Compiler Techniques for exposing ILP; Reducing Branch costs with prediction; Overcoming Data hazards with Dynamic scheduling; Hardware-based speculation.

**10 Hours**

**UNIT - 3**

**INSTRUCTION –LEVEL PARALLELISM – 2:** Exploiting ILP using multiple issue and static scheduling; Exploiting ILP using dynamic scheduling, multiple issue and speculation; Advanced Techniques for instruction delivery and Speculation; The Intel Pentium 4 as example.

**MULTIPROCESSORS AND THREAD –LEVEL PARALLELISM:**

Introduction; Symmetric shared-memory architectures; Performance of symmetric shared-memory multiprocessors;

**10 Hours**

**UNIT - 4**

**DISTRIBUTED SHARED MEMORY AND DIRECTORY-BASED COHERENCE;** Basics of synchronization; Models of Memory Consistency.

**REVIEW OF MEMORY HIERARCHY:** Introduction; Cache performance; Cache Optimizations, Virtual memory.

**10 Hours**

**TEXT BOOK:**

1. **Computer Architecture, A Quantitative Approach** – John L. Hennessey and David A. Patterson., 4th Edition, Elsevier, 2007.

**REFERENCE BOOKS:**

1. **Advanced Computer Architecture Parallelism, Scalability** – Kai Hwang., Programability, Tata Mc Grawhill, 2003.

2. **Parallel Computer Architecture, A Hardware / Software Approach** – David E. Culler, Jaswinder Pal Singh, Anoop Gupta: Morgan Kaufman, 1999.

**UIS002E: ADVANCED GRAPH THEORY.  
3 CREDITS (3-0-0)**

**UNIT – I**

**10 hrs.**

Introduction: What is a graph?, Applications of graphs, Finite and Infinite Graphs, Incidence and degree, Isolated and pendent vertices, null graphs.

Paths and Circuits: Isomorphism, Sub graphs, Walks, Paths, and Circuits, Connected graphs, Disconnected graphs, and components. Euler graphs, Hamiltonian paths and circuits. Traveling salesman problem.

Trees and Fundamental Circuits: Trees, Properties of trees, Pendent vertices in trees, Distance and centers in trees, Rooted and binary trees, Spanning trees, Fundamental Circuits, All spanning trees, spanning trees in a weighted graph.

**UNIT – II**

**10 hrs**

Cuts and vertices: Cut-sets, Properties of cut-sets, All cut-sets in a graph, connectivity and seperability, network flows, I & II isomorphism.

Planar and dual graphs: Combinatorial v/s geometric graphs, planar graphs, representations of a planar graphs, detection of planarity, geometric dual, combinatorial dual, criteria of planarity.

Matrix representations of graphs: Incident matrix, submatrices of  $A(G)$ , Circuit matrix, Fundamental circuit matrix and rank of  $B$ , Cut-set matrix, Relationships among  $A_f$ ,  $B_f$ , and  $C_f$ . Path matrix, adjacency matrix

**UNIT – III**

**10 hrs**

Coloring, Covering and partitioning: Chromatic Number, Chromatic partitioning, Chromatic Polynomial, Matchings, Coverings, four color problem.

Directed Graphs: What is Directed graphs, types of directed graphs, digraphs and binary relations, Euler digraphs, trees with directed edges, Fundamental circuits in directed graphs, Matrices  $A$ ,  $B$  and  $C$  of digraphs, Adjacency matrix of digraphs, paired comparison and tournament, acyclic digraphs and decylization.

**UNIT – IV**

**12 hrs**

Graph theoretic algorithma and Computer programs: Algorithms, representation of a graphs, some basic algorithms, connectedness and components, a spanning tree, a set of fundamental circuits, cut-vertices and seperability, directed circuits, shortest paths algorithms, depth first search on a graph, isomorphism.

Graphs in computer programming.

**Text Book:**

1. Narsingh Deo, Graph Theory: With applications to Engineering and Computer Science, Eastern Economy Edition ( Ch.1, 2, 3, 4, 5, 7, 8, 9, 11, 15.3)

**Reference Book:**

1. Ralph P. Grimaldi, B V. Ramana, ‘Discrete and Combinatorial Mathematics’, 5<sup>th</sup> edition, Pearson Education.

**UIS003E: ARTIFICIAL INTELLIGENCE  
3 CREDITS (3-0-0)**

**UNIT - 1**

**OVERVIEW OF ARTIFICIAL INTELLIGENCE:**

Overview of AI; The importance of AI; Early work in AI; AI and the related fields Knowledge.

**KNOWLEDGE: THE GENERAL CONCEPTS:**

Introduction; Definition and importance of Knowledge; Knowledge-Based Systems; Representation of knowledge; Knowledge Organization; knowledge manipulation; Acquisition of Knowledge.

**10 Hours**

**UNIT - 2**

**FORMALIZED SYMBOLIC LOGICS:**

Introduction; Syntax and Semantics for propositional Logic; Syntax and Semantics for FOPL; Properties of Wffs; Conversion to Clausal form; Inference rules; The Resolution principle; Non deductive inference methods; Representation using rules.

**DEALING WITH INCONSISTENCIES AND UNCERTAINTIES:**

Introduction; Truth Maintenance systems; Default reasoning and the closed world assumption; Predicate completion and circumscription; Modal and temporal logics; Fuzzy logic and natural language computations

**10 Hours**

**UNIT - 3**

**PROBABILISTIC REASONING**

Introduction; Bayesian probabilistic inference; Possible world representations; Dempster-Shafter theory; Ad-Hoc methods; heuristic reasoning methods;

**STRUCTURED KNOWLEDGE; GRAPHS, FRAMES, AND RELATED STRUCTURES**

Introduction; Associative networks; Frame structures; Conceptual dependencies and scripts.

**10 Hours**

**UNIT - 4**

**KNOWLEDGE ORGANIZATION AND MANIPULATIONS:**

**SEARCH and CONTROL STRATEGIES:**

Introduction; preliminary concepts; Examples of Search problems; Uniformed or Blind search; Informed search; Searching And-Or graphs.

**EXPERT SYSTEM ARCHITECTURES:**

Introduction; Rule based system architectures; Non production system architectures; Dealing with uncertainty; Knowledge acquisition and validation; Knowledge system building tools.

**10 Hours**

**TEXT BOOK:**

1. **Introduction to Artificial Intelligence and Expert Systems-** Dan W. Patterson, PHI, 2003

**REFERENCE BOOKS:**

1. **Artificial Intelligence** - Elaine Rich, Kevin Knight, 2nd Edition, Tata McGraw Hill, 1991.
2. **Principles of Artificial Intelligence** – Nils J. Nilsson, Elsevier, 1980.
3. **Artificial Intelligence: A Modern Approach** – Stuart Russel, Peter Norvig, 2nd Edition, Pearson Education, 2003.

**UIS004E: CRYPTOGRAPHY AND NETWORK SECURITY**  
**3 CREDITS (3-0-0)**

**UNIT I**

**INTRODUCTION TO NETWORK SECURITY:** OSI security architecture, security attacks, security services, Security Mechanisms, a model of Network Security.

**SYMMETRIC CIPHERS**

**10 Hrs**

Classical Encryption Techniques, Block Ciphers and the Data Encryption Standard, Introduction to Finite Fields, Confidentiality using Symmetric Encryption.

**UNIT II**

**PUBLIC - KEY ENCRYPTION AND HASH FUNCTIONS**

**10 Hrs**

Introduction to Number Theory, Public-Key Cryptography and RSA, Key Management : Diffie-Hellman Key Exchange, Message Authentication and Hash Functions, secure Hash Algorithm, Digital Signatures and Authentication Protocols.

**UNIT III**

**NETWORK SECURITY PRACTICE**

**10 Hrs**

Authentication Applications: Kerberos, X.509 Authentication Service, Electronic mail Security: Pretty Good Privacy, S/MIME, IP Security: Overview, Architecture, Authentication header, ESP, Key management.

**UNIT IV**

**SYSTEM SECURITY**

**10 Hrs**

Malicious Software: Viruses and Related Threats, Viruses Countermeasures. Distributed Denial of Service Attacks, Firewalls: Firewall Design Principles, Trusted Systems.

**REFERENCES:**

1. William Stallings, "Cryptography And Network Security – Principles and Practices", Pearson Education, Fourth Edition, 2006.  
(Chapters: 1, 2, 3.1, 3.2, 3.3, 4, 7, 8, 9, 10.1, 10.2, 11, 12.1, 13, 14, 15, 16, 19, 20)
2. Atul Kahate, "Cryptography and Network Security", Tata McGraw Hill, 2003.
3. Behrouz A. Forouzan, Introduction to Cryptography and Network Security, 2008, McGraw-Hill
4. Charles B. Pfleeger, Shari Lawrence Pfleeger, "Security in Computing", Fourth Edition, Pearson Education, 2007.

**UIS005E: DIGITAL IMAGE PROCESSING**  
**3 CREDITS (3-0-0)**

**UNIT-I**

**Fundamentals**

Introduction to Digital Image Processing, Elements of Visual Perception, Image Sensing and Acquisition, Image Sampling and Quantization, Basic relationships between pixels

**Image Enhancement in Digital Spatial Domain**

Background, Some basic gray level transformations, Histogram Processing, Enhancement using Arithmetic/Logic Operations, Basics of spatial filtering, Smoothing spatial filters, sharpening spatial filters

**Image Enhancement in the Frequency Domain**

Background, Introduction to the Fourier transform and the frequency domain, Smoothing frequency domain filters, Sharpening frequency domain filters, Homomorphic filtering, Implementation

**10Hrs**

**UNIT-II**

**Image Restoration**

A model of the image degradation/restoration process, Noise models, Restoration in the presence of noise only-spatial filtering, Periodic noise reduction by frequency domain filtering, Linear, position-invariant degradations, Estimating the degradation function, Inverse filtering, Minimum mean square error filtering, Constrained least squares filtering, Geometric mean filter, Geometric transformations

**10 Hrs**

**UNIT-III**

**Image Compression**

Fundamentals, Image compression models, Elements of information theory, Error-free compression, Lossy compression and image compression standards.

**10Hrs**

**UNIT-IV**

**Image Segmentation**

Detection of discontinuities, Edge linking and boundary detection, Thresholding, Region-based segmentation, Use of motion in segmentation.

**Object Recognition**

Pattern and pattern classes, Recognition based on Decision-Theoretic Methods

**10Hrs**

**TEXT BOOK:**

1. Rafael C. Gonzalez, Richard E. Woods: “**Digital Image Processing**”, 2nd Edition, Pearson Education, 2002.

**REFERENCE BOOKS:**

1. Anil K. Jain: “**Fundamentals of Digital Image Processing**”, Prentice-Hall of India Pvt. Ltd., 1997.

2. B. Chanda, Dutta Majumdeer: “**Digital Image Processing and Analysis**”, Prentice-Hall of India Pvt. Ltd., 2002.

**UIS006E: DISTRIBUTED OPERATING SYSTEMS**  
**3 CREDITS (3-0-0)**

**UNIT I**

**Fundamentals**

**4 hrs**

What is Distributed Computing Systems?, Distributed Computing System Models, What is Distributed Operating System?, Issues in Designing a Distributed Operating system, Introduction to Distributed Computing Environment(DCE).

**Message Passing**

**6 hrs**

Desirable Issues of Good Message Passing, Issues in IPC by Message Passing Synchronization, Buffering, Multidatagram Messages, Encoding and Decoding of message Data, Process Addressing, Failure Handling, Group Communication, Case Study: 4.3 BSD UNIX IPC Mechanism.

**UNIT II**

**Remote Procedure Calls**

**4 hrs**

The RPC Model, Transparency of RPC, Implementing RPC Mechanism, Stub Generation, RPC Messages, Marshaling Arguments and Results, Server Management, Parameter – Passing Semantics Call semantics, Communication Protocols for RPCs, Complicated RPCs, Client-Server Binding, Exception Handling, Security, Case Studies: Sun RPC.

**Distributed Shared Memory**

**6 hrs**

General Architecture of DSM Systems, Design and Implementation Issues of DSM, Granularity, Structure of Shared Memory Space, Consistency Models, Replacement Strategy, Thrashing.

**UNIT III**

**Synchronization**

**10 hrs**

Clock Synchronization, Event Ordering, Mutual Exclusion, Dead Lock, Election Algorithms.

**UNIT IV**

**Resource Management**

**5 hrs**

Desirable Features of a Good Global Scheduling Algorithm, Task Assignment Approach, Load – Balancing Approach, Load – Sharing Approach.

**Process Management**

**3 hrs**

Process Migration, Threads.

**Distributed File Systems**

**2 hrs**

Desirable Features of a Good Distributed File System, File models, File – Accessing Models, File – Sharing Semantics, File – Caching Schemes.

**TEXT BOOK:**

- 1) Distributed Operating System: Concepts and Design, Pradeep .K. Sinha, 1997, PHI  
[Chapters: 1: 1.1, 1.3, 1.5-1.7, Chapter 3: 3.2-3.11, Chapter 4: 4.2-4.15, 4.20, Chapter 5: 5.2-5.8, Chapter 6: 6.2-6.6, Chapter 7: 7.2-7.4, Chapter 8: 8.2-8.3, Chapter 9: 9.3-9.6]

**REFERENCE BOOK:**

Distributed Operating System, Andrew .S. Tanenbaum, Pearson Education, 2002.

**UIS007E: FUZZY LOGIC**  
**3 CREDITS (3-0-0)**

**UNIT – 1**

**CRISP SETS AND FUZZY SETS:**

Introduction; crisp sets: An overview; The notion of Fuzzy Sets; Basic concepts of Fuzzy Sets; Classical logic : An overview; Fuzzy logic.

**OPERATIONS ON FUZZY SETS:**

General discussion; Fuzzy Complement Fuzzy Union; Fuzzy intersection; Combination of operations; General aggregation operations.

**CLASSICAL RELATIONS AND FUZZY RELATIONS:** Cartesian Product, Crisp Relations - Cardinality of Crisp Relations, Operations on Crisp Relations, Properties of Crisp Relations, Composition. Fuzzy Relations - Cardinality of Fuzzy Relations, Operations on Fuzzy Relations, Properties of Fuzzy Relations, Fuzzy Cartesian Product and Composition, Non interactive Fuzzy sets. Tolerance and Equivalence Relations – Crisp Equivalence relation, Crisp Tolerance Relation, Fuzzy Tolerance and Equivalence relations. Value Assignments - Cosine Amplitude, Max-min Method, other Similarity methods.

**10 Hours**

**UNIT - 2**

**MEMBERSHIP FUNCTIONS:** Features of the Membership Function, Standard Forms and Boundaries, Fuzzification, Membership Value Assignments – Intuition, Inference, Rank Ordering, Angular Fuzzy Sets, Neural Networks, Genetic Algorithms, Inductive Reasoning.

**FUZZY-TO-CRISP CONVERSIONS, FUZZY ARITHMETIC:**

Lambda-Cuts for Fuzzy Sets, Lambda-Cuts for Fuzzy Relations, Defuzzification Methods. Extension Principle - Crisp Functions, Mapping and Relations.

**10Hours**

**UNIT – 3**

**FUNCTIONS OF FUZZY SETS** – Extension Principle, Fuzzy Transform (Mapping), Practical Considerations. Fuzzy Numbers Interval Analysis in Arithmetic, Approximate Methods of Extension - Vertex method, DSW Algorithm, Restricted DSW Algorithm, Comparisons. Fuzzy Vectors.

**CLASSICAL LOGIC AND FUZZY LOGIC:** Classical Predicate Logic – Tautologies, Contradictions, Equivalence, Exclusive Or and Exclusive Nor, Logical Proofs, Deductive Inferences. Fuzzy Logic, Approximate Reasoning, Fuzzy Tautologies, Contradictions, Equivalence and Logical Proofs, Other forms of the Implication Operation, Other forms of the Composition Operation.

**10Hours**

**UNIT – 4**

**FUZZY RULE- BASED SYSTEMS:** Natural Language, Linguistic Hedges, Rule-Based Systems - Canonical Rule Forms, Decomposition of Compound Rules, Likelihood and Truth Qualification, Aggregation of Fuzzy Rules.

**FUZZY DECISION MAKING:** Fuzzy Synthetic Evaluation, Fuzzy Ordering, Preference and consensus, Multiobjective Decision Making, FuzzyBayesian Decision Method, Decision Making under Fuzzy States and Fuzzy actions.

**10 Hours**

**TEXT BOOK:**

1. **Fuzzy Logic with Engineering Applications** – Timothy J. Ross, McGraw-Hill, 1997.
2. **Fuzzy Sets Uncertainty and Information** – George J. Klir and Tina A. Folger ,Prentice Hall of India,2006

**REFERENCE BOOK:**

1. **Neural Networks and Fuzzy systems: A Dynamical System Approach** – B Kosko, Prentice Hall, 1991.

**UIS008E: GENETIC ALGORITHMS**  
**3 CREDITS (3-0-0)**

**UNIT I**

**INTRODUCTION TO GENETIC ALGORITHMS** **10 Hrs**

**A Gentle Introduction to genetic algorithms**

What Are Genetic Algorithms?, Robustness of Traditional Optimization and Search Methods, The Goals of Optimization, How Are Genetic Algorithms Different from Traditional Methods?, A Simple Genetic Algorithm, Genetic Algorithms at Work-a Simulation by hand, Grist for the Search Mill-Important Similarities, Similarity Templates (Schemata), learning the Lingo.

**Genetic Algorithms Revisited: Mathematical Foundations**

Who Shall Live and Who Shall Die? The Fundamental Theorem, Schema Processing at Work: An Example by Hand Revisited, The Two-armed and k-armed Bandit Problem, How Many Schemata Are Processed Usefully, The Building Block Hypothesis, Another Perspective: The Minimal Deceptive Problem

**UNIT II**

**GA OPERATORS** **11 Hrs**

**Computer Implementation of a Genetic Algorithm**

Data Structures, Reproduction, Roulette-wheel Selection, Boltzman Selection, Tournament Selection-Rank Selection, Steady state selection, Crossover, and Mutation , A Time to Reproduce, a Time to Cross , Get with the Main Program, How Well Does it Work?, Mapping Objective Functions to Fitness Form, Fitness Scaling, Codings, A Multiparameter, Mapped, Fixed-Point Coding, Discretization , Constraints.

**UNIT III**

**APPLICATIONS OF GA** **11 Hrs**

**Some Applications of Genetic Algorithms**

The Rise of Genetic Algorithms, Genetic Algorithm Applications of Historical Interest, De Jong and Function Optimization, Improvements in Basic Technique, Current Applications of Genetic Algorithms.

**Advanced Operators and Techniques in Genetic Search**

Dominance, Diploidy, and Abeyance, Inversion and Other Reordering Operators, Other Micro-operators, Niche and Speciation, Multiobjective Optimization, Knowledge-Based Techniques.

**UNIT IV**

**INTRODUCTION TO GENETICS-BASED MACHINE LEARNING** **10 Hrs**

**Introduction to Genetics Based Machine learning**

Genetics-Based Machine Learning: Whence It Came, What is a Classifier System? , Rule and Message System, Apportionment of Credit: The Bucket Brigade, Genetic Algorithm, A Simple Classifier System in Pascal, Results Using the Simple Classifier System.

**TEXT BOOKS**

1. David E. Gold Berg, "Genetic Algorithms in Search, Optimization & Machine Learning", Pearson Education, 2001.
2. S.Rajasekaran, G.A.Vijayalakshmi Pai, "Neural Networks, Fuzzy Logic and Genetic Algorithms", PHI , 2003 ( Chapters 8 and 9 ).

## REFERENCE BOOK

1. Kalyanmoy Deb, "Optimization for Engineering Design, algorithms and examples", PHI, 1995.

**UIS009E: NETWORK MANAGEMENT SYSTEMS**  
**3 CREDITS (3-0-0)**

**UNIT – 1**

**INTRODUCTION, N/W MANAGEMENT STANDARDS, MODELS:**

Introduction: Analogy of Telephone Network Management, Data and Telecommunication Network Distributed computing Environments, TCP/IPBased Networks: The Internet and Intranets, Communications Protocols and Standards- Communication Architectures, Protocol Layers and Services; Case Histories of Networking and Management – The Importance of topology, Filtering Does Not Reduce Load on Node, Some Common Network Problems; Challenges of Information Technology Managers, Network Management: Goals, Organization, and Functions- Goal of Network Management, Network Provisioning, Network Operations and the NOC, Network Installation and Maintenance; Network and System Management, Network Management System platform, Current Status and Future of Network Management. Network Management Standards, Network Management Model, Organization Model, Information Model – Management Information Trees.

**10 Hours**

**UNIT – 2**

**N/W MANAGEMENT LANGUAGE, SNMPV1 NETWORK MANAGEMENT – 1**

Communication Model; ASN.1- Terminology, Symbols, and Conventions, Objects and Data Types, Object Names, An Example of ASN.1 from ISO 8824; Encoding Structure; Macros, Functional Model. Snmpv1 network management – 1: Managed Network: The History of SNMP Management, Internet Organizations and standards, Internet Documents, The SNMP Model, The Organization Model, System Overview.

**10 Hours**

**UNIT - 3**

**SNMPV1 NETWORK MANAGEMENT – 2 SNMP MANAGEMENT – RMON:** The Information Model – Introduction, The Structure of Management Information, Managed Objects, Management Information Base. The SNMP Communication Model – The SNMP Architecture, Administrative Model, SNMP Specifications, SNMP Operations, SNMP MIB Group, Functional Model. Snmp management – RMON: Remote Monitoring, RMON SMI and MIB, RMON1- RMON1 Textual Conventions, RMON1 Groups and Functions, Relationship Between Control and Data Tables, RMON1 Common and Ethernet Groups, RMON Token Ring Extension Groups.

**10 Hours**

**UNIT – 4**

**RMON2, BROADBAND N/W MANAGEMENT, N/W MANAGEMENT APPLICATIONS:**

RMON2 – The RMON2 Management Information Base, RMON2 Conformance Specifications; ATM Remote Monitoring, A Case Study of Internet Traffic Using RMON. Broadband Network Management: ATM Network: Broadband Networks and Services, ATM Technology – Virtual Path-Virtual Circuit, TM Packet Size, Integrated Service, SONET, ATM LAN Emulation, Virtual LAN; ATM Network Management – The ATM Network Reference Model, The Integrated Local Management Interface, The ATM Management Information Base, The Role of SNMP and ILMI in ATM Management, M1 Interface: Management of ATM Network Element, M2 Interface: Management of Private Networks, M3 Interface: Customer Network Management of Public Networks, M4 Interface: Public Network Management, Management of LAN Emulation, ATM Digital Exchange Interface Management. Network Management Applications: Configuration Management- Fault Management, Performance Management, Event Correlation, Security Management,

Accounting Management applications overview, Policy- Based Management, Service Level Management.

**10 Hours**

**TEXT BOOK:**

1. **Network Management- Principles and Practice** – Mani Subramanian, Pearson Education, 2003.

**REFERENCE BOOK:**

1. **Network Management Concepts and Practices A Hands-On Approach** - J. Richard Burke, PHI, 2008.

**UIS010E: PATTERN RECOGNITION**  
**3 CREDITS (3-0-0)**

**UNIT I**

**(10 hrs)**

**Pattern Recognition Overview:** Pattern recognition, classification and description, Patterns and feature extraction with example, Training and learning PR systems, Pattern Recognition Approaches.

**Bayes Decision theory:** Introduction, Bayesian Decision Theory - continuous features, minimum error rate classification, classifiers, discriminant functions, and decision surfaces. Error probabilities and integrals, normal density, discriminant functions for normal density, Bayes Decision theory - Discrete features.

**UNIT II**

**(10 hrs)**

**Supervised learning(parametric technique):** Parameter estimation and supervised learning, Maximum likelihood estimation, The Bayesian Parameter estimation approach..

**Nonparametric technique:** Density estimation, Parzen windows,  $k_n$  - Nearest Neighbor estimation, estimation of posterior probabilities, nearest- neighbor rule.

**UNIT III**

**(10 hrs)**

**Linear discriminant functions:** Linear discriminant functions and decision surfaces, generalized linear discriminant functions, 2-category linearly separable case, non-separable behavior, linear programming algorithms.

**Multiplayer neural networks:** Feed forward operation and classification, Back propagation algorithm, error surfaces, back propagation as feature mapping, practical techniques for improving back propagation.

**UNIT IV**

**(10 hrs)**

**Unsupervised learning and clustering:** Mixture densities and identifiably, maximum likelihood estimates, application to normal mixtures, unsupervised Bayesian learning, data description and clustering, hierarchical clustering.

**Text Books**

1. R. O. Duda, P. E. Hart, and D. G. Stork, “*Pattern Classification*”, 2nd ed., John Wiley & Sons, New York, 2001 (Chapter 2,4,5,6,10)
2. Robert J. Schalkoff, “*Pattern Recognition: Statical, Structural and Neural Approaches*” John Wiley & Sons, New York, 2005 (Chapter 1,3, )

**Reference Books**

1. K. S. Fu, “*Syntactic Pattern Recognition and Applications*”, Prentice Hall, Eaglewood cliffs, N.J., 1982
2. C. M. Bishop, “*Neural Network for Pattern Recognition*”, Oxford University Press, New York, 1998
3. E. Gose, R. Johnsonbaugh, and S. Jost, “*Pattern Recognition and Image Analysis*”, Prentice Hall of India, New Delhi, 1999

**UIS011E: REAL TIME OPERATING SYSTEMS**  
**3 CREDITS (3-0-0)**

**UNIT – 1**

**Basic Real-Time Concepts:**

Basic computer architecture, Some terminology, Real Time Design Issues, Example Real Time Systems, Basic History , **Computer Hardware:** CPU, Memories, Input and Output, Other Devices

**08 Hours**

**UNIT – II**

**REAL TIME Specification and Design Techniques:**

Natural Languages, Mathematical Specification, Flowcharts, structure charts, Pseudo code and Programming Design Languages, Finite state Automata, Data Flow Diagrams, Petri Nets, Warmier- Orr Notation, and State Charts.

**Real Time Kernels:** Polled loop systems, phase/state driven code, co routines, Interrupt driven systems, foreground/background systems, full featured real time operating systems, POSIX

**10Hours**

**UNIT – III**

**Intertask Communication and Synchronization:** Buffering data, Mailboxes, Critical Regions, Semaphores, Event Flags and Signals, Deadlock.

**Real Time Memory Management:** Process Stack Management, Dynamic Allocation

**10 Hours**

**UNIT – IV**

**FAULT TOLERANCE TECHNIQUES:**

Introduction, Errors and Failures, Fault types, Detection and Containment, Redundancy, Integrated Failure Handling.

**RELIABILITY EVALUATION:** Introduction, Parameters, Reliability Models for Hardware, Software Error Models.

**12Hours**

**TEXT BOOKS:**

1. **Real-Time Systems Design and Analysis**, Phillip. A. Laplante, second edition, PHI, 2005.
2. **Real Time Systems**, C. M. Krishna, Kang. G. Shin, Mc Graw Hill, India, 1997.

**REFERENCE BOOKS:**

1. **Embedded Systems**, Raj Kamal, Tata Mc Graw Hill, India, 2008.
2. **Real Time Systems**, Jane. W. S. Liu, Pearson education, 2005.

**UIS012E: SOFTWARE ARCHITECTURE  
3 CREDITS (3-0-0)**

**UNIT – 1**

**10 Hrs**

**INTRODUCTION, ARCHITECTURE STYLES**

The Architecture Business Cycle: Software processes and the architecture business cycle; Qualities of a “good” architecture; Other points of view; Architectural patterns, reference models and reference architectures; Importance of software architecture; Architectural structures and views.

Architectural styles; Pipes and filters; Data abstraction and object-oriented organization; Event-based, implicit invocation, Layered systems, Repositories, Interpreters, Process control; Other familiar architectures.

**UNIT – 2**

**ARCHITECTURE STYLES: CASE STUDIES, QUALITY**

**10 Hrs**

Case Studies: Keyword in Context; Instrumentation software; Mobile robotics; Cruise control; Heterogeneous architectures;

Quality : Functionality and architecture; Architecture and quality attributes; System quality attributes; Quality attribute scenarios in practice; Other system quality attributes; Business qualities; Architecture qualities. Achieving Quality: Introducing tactics;

**UNIT – 3**

**10 Hrs**

**ARCHITECTURAL PATTERNS**

Architectural patterns: Introduction, classification-1) From mud to structure: Basic concepts of Layers, 2) Distributed Systems: Basic concepts of Broker; 3) Interactive Systems: Basic concepts of MVC, 4): Adaptable Systems: Basic concepts of Microkernel, reflection

**UNIT -4**

**10 Hrs**

**DESIGN PATTERNS, DESIGNING AND DOCUMENTATION**

Some design patterns: Structural decomposition: Basic concepts of Whole – Part; Organization of work: Basic concepts of Master – Slave; Access Control: Basic concepts of Proxy.

Designing and documenting software: Architecture in the life cycle; Designing the architecture; Forming the team structure; Creating a skeletal system. Uses of architectural documentation; Views; Choosing the relevant views; Documenting a view; Documentation across views.

**TEXT BOOKS:**

1. Software Architecture in Practice - Len Bass, Paul Clements, Rick Kazman, 2nd Edition, Pearson Education, 2003.
2. Pattern-Oriented Software Architecture A System of Patterns, Volume 1 - Frank Buschmann, Regine Meunier, Hans Rohnert, Peter Sommerlad, Michael Stal, John Wiley and Sons, 2006
3. Software Architecture- Perspectives on an Emerging Discipline - Mary Shaw and David Garlan, Prentice-Hall of India, 2007

**REFERENCE BOOK:**

Design Patterns- Elements of Reusable Object-Oriented Software - E. Gamma, R. Helm, R. Johnson, J. Vlissides, Addison-Wesley, 1995

**UIS013E: WEB SERVICES AND SERVICE ORIENTED ARCHITECTURE  
3 CREDITS (3-0-0)**

**UNIT I**

**INTRODUCTION TO WEB SERVICES AND XML**

**10 Hrs.**

**Introduction**

The basics of Web Services: An Example; Next Generation of the Web; Interacting with Web Services; The Technology of Web Services; XML for business collaboration; ebXML; Web Services versus Other Technologies; Additional Technologies;

**XML**

An Example; Instance and Schema; Processing XML Documents; Namespaces; Transformation; XML Specifications and information.

**UNIT II**

**WSDL AND SOAP**

**11 Hrs.**

**WSDL**

Basics; WSDL elements; The Extensible WSDL framework; Importing WSDL elements; WSDL-Related Namespaces; Extensions for binding to SOAP.

**SOAP**

Example; The SOAP Specifications; SOAP Message Processing; SOAP Use of Namespaces; Changes in the V1.2 draft; SOAP Multipart MIME Attachments; SOAP In the Context of Existing Systems; Future directions.

**UNIT III**

**INTRODUCTION TO SOA & EVOLUTION OF SOA**

**11 Hrs.**

**INTRODUCTION TO SOA, EVOLUTION OF SOA:** Fundamental SOA; Common Characteristics of contemporary SOA; Common tangible benefits of SOA; An SOA timeline (from XML to Web services to SOA);

**WEB SERVICES AND PRIMITIVE SOA:** The Web services framework; Services (as Web services); Service descriptions (with WSDL); Messaging (with SOAP).

## UNIT IV

### WEB SERVICES AND CONTEMPORARY SOA

10 Hrs.

**WEB SERVICES AND CONTEMPORARY SOA – 1:** Message exchange patterns; Service activity; Coordination; Atomic Transactions; Business Activities; Orchestration; Choreography.

**WEB SERVICES AND CONTEMPORARY SOA – 2:** Addressing; Reliable messaging; Correlation; Policies

### TEXT BOOKS

1. Eric Newcomer, "Understanding Web Services XML, WSDL, SOAP, and UDDI", Pearson Education, 2002.
2. James McGovern et al, "Java Web Services Architecture", Elsevier, 2003.
3. Thomas Erl "Service-Oriented Architecture Concepts, Technology, and Design ", Pearson Education, 2005.

### REFERENCE BOOK

Eric Newcomer, Greg Lomow, "Understanding SOA with Web Services", Pearson education, 2005.

**UIS014E: SYSTEM MODELING AND SIMULATION**  
**3 CREDITS (3-0-0)**

**UNIT I**

**10 Hours**

**Introduction** When simulation is the appropriate tool and when it is not appropriate; Advantages and disadvantages of Simulation; Areas of application; Systems and system environment; Components of a system; Discrete and continuous systems; Model of a system; System modeling , principles used in modeling, Types of Models; Monte Carlo simulation method , Discrete-Event System Simulation; Steps in a Simulation Study. Simulation examples: Simulation of queuing systems;

**UNIT II**

**10 Hours**

**General Principles:** Concepts in Discrete-Event Simulation: The Event-Scheduling/ Time-Advance Algorithm, World Views, Manual simulation Using Event Scheduling. **Random-Number Generation:** Properties of random numbers; Generation of pseudo-random numbers; Techniques for generating random numbers; Tests for Random Numbers.

**UNIT III**

**10 Hours**

**Random-Variate Generation:** Inverse transforms technique-Exponential distribution, uniform distribution, discrete distributions, Acceptance-Rejection Technique-Poisson Distribution.**Verification and Validation of Simulation Models:** Model building, verification and validation; Verification of simulation models; Calibration and validation of models.

**UNIT IV**

**10 Hours**

**Input Modeling:** Data Collection; Identifying the distribution with data; Parameter estimation; Goodness of Fit Tests; Fitting a non-stationary Poisson process; Selecting input models without data. **Output Analysis for a Single Model:** Types of simulations with respect to output analysis; Stochastic nature of output data. Measures of performance and their estimation; Output analysis for terminating simulations. **Simulation of computer system:** Introduction Simulation tools- Process orientation, Event Orientation, CPU simulation.

**Text Books:**

1. Jerry Banks, John S. Carson II, Barry L. Nelson, David M. Nicol: Discrete-Event System Simulation, 4<sup>th</sup> Edition, Pearson Education, 2007.
2. Geoffrey Gordon “System Simulation”, PHI, Second Edition.

**Reference Books:**

1. Lawrence M. Leemis, Stephen K. Park: Discrete – Event Simulation: A First Course, Pearson / Prentice-Hall, 2006.
2. Averill M. Law: Simulation Modeling and Analysis, 4th Edition, Tata McGraw-Hill, 2007

## UIS015E: JAVA AND J2EE

**3 CREDITS (3-0-0)**

### UNIT 1

**(9 hours)**

**INTRODUCTION TO JAVA:** Java and Java applications; Java Development Kit(JDK); Java is interpreted, Byte, JVM; Object oriented programming; Simple Java programs. Data types and other tokens: Boolean variables, int, long, char, operators, arrays, white spaces, literals, assigning values; Creating and destroying objects; Access specifiers. Operators and Expressions: Arithmetic operators, Bitwise operators, Relational operators, The Assignment Operator, The ? Operator; Operator Precedence; Logical expression; Type casting; Strings. Control Statements: Selection statements, iteration statements, Jump statements.

**CLASSES, INHERITANCE, EXCEPTIONS:** **Classes:** classes in Java; Declaring a class; Class name; Super classes; Constructors; Creating instances of class; Inner classes.  
**Inheritance:** Simple, multiple and multilevel inheritance; Overriding, overloading.  
Exception handling: Exception handling in Java.

### UNIT 2

**(11 hours)**

#### **APPLETS, MULTI THREADED PROGRAMMING:**

**The Applet Class:** Two types of Applets; Applet basics; Applet Architecture; An Applet skeleton; Simple Applet display methods; Requesting repainting; Using the Status Window; THE HTML APPLLET tag; Passing parameters to Applets; getDocumentbase() and getCodebase(); ApletContext and showDocument(); The AudioClip Interface; The AppletStub Interface; Output to the Console.

**Multi Threaded Programming :** What are threads? How to make the classes threadable; Extending threads; Implementing runnable; Synchronization; Changing state of the thread; Bounded buffer problems, read-write problem, producer-consumer problems.

### UNIT 3

**(10 hours)**

**EVENT HANDLING:** Two event handling mechanisms; The delegation event model; Event classes; Sources of events; Event listener interfaces; Using the delegation event model; Adapter classes; Inner classes.

#### **JAVA 2 ENTERPRISE EDITION OVERVIEW, DATABASE ACCESS:**

Overview of J2EE and J2SE.

The Concept of JDBC; JDBC Driver Types; JDBC Packages; A Brief Overview of the JDBC process; Database Connection; Associating the JDBC/ODBC Bridge with the Database; Statement objects; ResultSet; Transaction Processing; Metadata, Data types; Exceptions.

## UNIT 4

(10 hours)

**SERVLETS:** Background; The Life Cycle of a Servlet; Using Tomcat for Servlet Development; A Simple Servlet; The Servlet API; The javax.servlet Package; Reading Servlet Parameter; The javax.servlet.http package; Handling HTTP Requests and Responses; Using Cookies; Session Tracking.

**JSP, RMI:** Java Server Pages (JSP): JSP, JSP Tags, Tomcat, Request String, User Sessions, Cookies, Session Objects.

Java Remote Method Invocation: Remote Method Invocation concept; Server side, Client side.

### TEXT BOOKS:

1. **Java The Complete Reference** – Herbert Schildt, 7th Edition, Tata McGraw Hill, 2007.
2. **J2EE The Complete Reference** – Jim Keogh, Tata McGraw Hill, 2007.

### REFERENCE BOOKS:

1. **Introduction to JAVA Programming-** Y. Deniel Liang, 6<sup>th</sup> Edition, Pearson Education, 2007.
2. The J2EE Tutorial- Stephanie Bodoff et al, 2<sup>nd</sup> Edition, Pearson Education, 2004.

**UIS016E: ADVANCED DBMS.**  
**3 CREDITS (3-0-0)**

**UNIT – I**

**10 hrs.**

Overview of normalization techniques.

Disk Storage, Basic File Structures, and Hashing: Introduction, Secondary storage structures, Buffering of blocks, Placing file records on disk, Operations on files, Files of unordered records (heap files), Files of ordered records (sorted files), Hashing Techniques.

Indexing structures for files: Types of single level ordered indexes, Multiple indexes, Dynamic Multilevel indexes using B-tress and B<sup>+</sup>-trees, Indexes on multiple keys, Other types of indexes.

**UNIT – II**

**10 Hrs**

Algorithms for query processing and optimization: Translating SQL queries into relational algebra, Algorithms for external sorting, Algorithms for SELECT, JOIN, PROJECT and Set operations, Implementing aggregate operations and OUTER JOINS Combining operations using pipelines, Using heuristics in query optimization, Using selectivity and cost estimates in query optimization.

Physical Database design and Tuning: Physical database design in relational database, Database tuning in relational systems.

**UNIT – III**

**10 Hrs**

Web Database Programming Using PHP: Structured, semistructured and semi structured data, A simple PHP example, Overview of basic features of PHP, Overview of PHP Database programming.

XML: XML hierarchical (tree) data model, XML documents, DTD and XML schema, XML querying.

**UNIT – IV**

**10 Hrs**

Data Mining Concepts: Overview of data mining technology, Association rules, Classification, Clustering, Approaches to other data mining problems, Applications of data mining, commercial data mining tools.

Overview of Data Warehousing and OLAP: Introduction definition and terminology, characteristics of data warehousing, data modeling for data warehouses, Building a data warehouse, typical functionality of data warehouse, data warehouse versus views, problems and open issues in data warehouses.

**Text Book:**

1. Rameez Elmashri, Shamakant B Navathe, 'Fundamentals of Database Systems', Fith Edition, Pearson Education.

**Reference Books:**

1. C J. Date, 'An Introduction to Database Systems', Sixth Edition, Adidson-Wesley.
2. Raghu Ramakrishna, Gehrke, 'Database Management Systems', Mc. Graw-Hill, Fifth edition.
3. Peter Rob, Carlos Coronel, 'Database Systems: Design, Implementation & Management, Fourth Edition, Thomson Publications.
4. Bipin C. desai, 'Introduction to Database Systems', Galgotia, New Delhi.

# **UIS017E: Artificial Intelligence and Expert System**

**3 Credits (3-0-0)**

## **UNIT – I**

### **Introduction and Problems and Search**

#### **Introduction**

Concepts and definition of AI, AI Problems, The Underlying assumption, What is an AI technique?, AI characteristics, Artificial Intelligence versus Natural Intelligence, Applications of AI. **6 Hrs.**

#### **Problems, Problem Spaces, and Search**

Defining the Problem as State Space Search, Production Systems, Problem Characteristics, Production Systems Characteristics, Issues in the Design of Search Programs, Advantages and Disadvantages of DFS & BFS Techniques. **6 hrs**

## **UNIT - II**

### **Heuristic Search Techniques and Knowledge Representation**

Generate – and – Test, Hill Climbing, Best-First Search, Problem reduction – AND – OR Graphs, Means-Ends Analysis. **6 Hrs**

**Knowledge Representation** - Representations and Mappings, Approaches to Knowledge Representation **4 Hrs.**

## **UNIT - III**

### **Knowledge Representation**

Issues in Knowledge Representation. Predicate Logic, Representing Simple Facts in Logic, Representing Instance and Isa Relationships, Computable Functions and Predicates. **8 Hrs**

Resolution – Conversion to Clause form, the basis of resolution, resolution in propositional logic. **2 Hrs.**

## **UNIT - I V**

### **Expert Systems**

Basic Concepts of Expert System, Structure of Expert Systems, The Human Element in Expert Systems, How Expert Systems Work, Example of an Expert System Consultation, Problem Areas Addressed by Expert Systems, Benefits of Expert Systems, Problems and Limitations of Expert Systems, Expert System Success Factors, Type of Expert Systems, Expert Systems and the Internet / Intranets / Web. **10 Hrs**

**Text Books:**

1. Artificial Intelligence, Elaine Rich, Kevin Knight, Second Edition, Tata McGraw Hill.
2. Decision Support Systems and Intelligent Systems, Efraim Turban and Jay E. Aronson, Sixth Edition 2002, Pearson Education Asia.

**Reference Book:**

1. Artificial Intelligence & Expert Systems, S- Dan W. Patterson, Prentice Hall of India.

**UIS018E: CLIENT SERVER COMPUTING**  
**3 CREDITS (3-0-0)**

**UNIT I**

**10 Hrs**

**Introduction:** to client/server computing, advantages of client/server computing.  
**Architecture:** Data access architecture, Execution architecture Vertical slice-two-tiered client/server, stored procedure, three-tiered architecture. **Role of the client,** client services, Remote procedure call, print. Services, Remote services, Utility services, Message services, Network services, Application services, Database services,

**UNIT II**

**10 Hrs**

**Server functionality,** Request processing, File services, Database services, Communication services, Security services, Network operating system, platforms, Server operating system.  
**Connectivity** - Open systems interconnect, communications interface technology, interprocess communication.

**UNIT III**

**10Hrs**

**Application development management issues--** platform and productivity, environment definition, productivity measures, performance, support, organization and management, task allocation server and client side.

**UNIT IV**

**10 Hrs**

Distributed objects and components - CORBA, compound documents, Opendoc component model.

**TEXT BOOKS:**

1. Client/Server computing' by Patrick Smith and Steve Guengerich, II Edition, Prentice Hall.
2. The Essential client/server survival Guide' by Robert Orfali, Dan Harkey, Jeri Edwards, II edition, Galgotia Publications.

**REFERENCE BOOKS:**

1. Client/Server System Design and implementation by Larry T Vaughn, McGraw-Hill international Edition.
2. The CORBA Reference Guide by Alan Pope, Addison Wellesley

**UIS019E: DATA MINING  
(3-0-0)**

**UNIT – I**

**10 Hrs**

Introduction to data mining: Definition of Data Mining, Motivating Challenges of DM, Data Mining Tasks.

Data: Data Attributes, Types of Data, Quality of Data and Data Preprocessing, Measures of Similarity and Dissimilarity.

**UNIT – II**

**10 Hrs**

Association Analysis: Definition of Association Analysis, Frequent Item Set Generation, Rule Generation, Compact Representation of Frequent Item Sets. Alternate Method of Generating Item Sets, FP Growth Algorithms, Evaluation of Association Patterns

**UNIT – III**

**10 Hrs**

Classification: Preliminaries, General Approach To Solving Classification Problem, Decision Tree Based Classifier, Rule Based Classifier, Nearest Neighbor Classifier.

Cluster Analysis: Overview, K-means, DBSCAN

**UNIT – IV**

**10 Hrs**

Applications: Data Mining Applications, Web Mining, Search Engines

**Text Books:**

1. "Introduction to Data Mining", Pang-Ning Tan, Michael Steinbach, Vipin Kumar, Pearson Education. (Chapter 1, 2, 4.1, 4.2, 4.3, 5.1, 5.2, 6.1, 6.2, 6.3, 6.4, 6.5, 6.7, 6.8, 8.1, 8.2, 8.4)
2. "Data Mining – Concepts and Techniques", Jiawei Han and Micheline Kamber, Morgan Kaufman, 2006, 2<sup>nd</sup> Edition. (Chapter 10)
3. "Introduction to Data Mining with Case Studies", G K Gupta, PHI. (Chapter 5, 6)

**UIS020E: DISTRIBUTED DATA BASES**  
**3 CREDITS (3-0-0)**

**UNIT – I**

**10 Hrs**

Distributed Databases an Overview.

Levels of distribution transparency: Reference architecture, Types of data fragmentation, distribution transparency for read only queries, distribution transparency update applications, Distributed Database access primitives, Integrity constraints in distributed Databases.

Distributed Database Design: A frame work for Distributed Database design, Design of database fragmentation, allocation of fragments.

**UNIT – II**

**10 Hrs**

Translation of global queries into fragment queries: Equivalence transformation for queries, Transforming global queries into fragment queries, Distributed grouping and aggregate function evaluation, Parametric queries.

Optimization of Access strategies: A frame work for query optimization, Join queries, General queries.

**UNIT – III**

**10 Hrs**

Management of Distributed Transactions: A frame work for transaction management, supporting atomicity of distributed transactions, Concurrency control for distributed transactions, Architectural aspects of distributed transactions.

Concurrency control: Foundations of distributed concurrency control, distributed deadlocks, concurrency control based on time stamps.

**UNIT – IV**

**10 Hrs**

Reliability: Basic concepts, Nonblocking commitment protocols, reliability and concurrency control,

Distributed database administration: Catalog management, Authorization and protection.

The R\* project: Case study.

**Text Books:**

1. Stefano Cery, Giuseppe Pelagatti, 'Distributed Databases: Principles and Systems', Mc. Graw-Hill International Editions. (ch. 1, 3, 4, 5, 6, 7, 8, 9, 10, 13)

**Reference Books:**

1. Rameez Elmashri, Shamkant B Navathe, 'Fundamentals of Database Systems', Fifth Edition, Pearson Education.
2. C.J Date, 'An Introduction to Database Systems', Sixth Edition, Addison-Wesley.
3. Raghu Ramakrishna, Gehrke, 'Database Management Systems', Mc. Graw-Hill, Fifth edition.

**UIS021E: EMBEDDED SYSTEMS**  
**3 CREDITS (3-0-0)**

**UNIT – 1**

**INTRODUCTION:** Overview of embedded systems, embedded system design challenges, common design metrics and optimizing design metrics. Survey of different embedded system design technologies, trade-offs. Custom Single- Purpose Processors, Design of custom single purpose processors.

**SINGLE-PURPOSE PROCESSORS:** Hardware, Combinational Logic, Sequential Logic, RT level Combinational and Sequential Components, Optimizing single-purpose processors. Single-Purpose Processors: Software, Basic Architecture, Operation, Programmer's View, Development Environment, ASIPS.

**08 Hours**

**UNIT – II**

Standard Single-Purpose Peripherals, Timers, Counters, UART, PWM, LCD Controllers, Keypad controllers, Stepper Motor Controller, A to D Converters.

**MEMORY:** Introduction, Common memory Types, Compulsory memory, Memory Hierarchy and Cache, Advanced RAM.

**08 Hours**

**UNIT – III**

Interfacing, Communication Basics, Microprocessor Interfacing, Arbitration, Advanced Communication Principles, Protocols - Serial, Parallel and Wireless. INTERRUPTS: Basics - Shared Data Problem - Interrupt latency.

**08 Hours**

**UNIT – IV**

Survey of Software Architecture, Round Robin, Round Robin with Interrupts – Function Queues - scheduling - RTOS architecture.

**INTRODUCTION TO RTOS:** Tasks - states - Data - Semaphores and shared data. More operating systems services - Message Queues - Mail Boxes -Timers – Events - Memory Management, Interrupt routines in RTOS environment.

**16 Hours**

**TEXT BOOKS:**

1. **Embedded System Design: A Unified Hardware/Software Introduction** - Frank Vahid, Tony Givargis, John Wiley & Sons, Inc.2002
2. **An Embedded software Primer** - David E. Simon: Pearson Education, 1999

**REFERENCE BOOKS:**

1. **Embedded Systems: Architecture and Programming**, Raj Kamal, TMH. 2008
2. **Embedded Systems Architecture – A Comprehensive Guide for Engineers and Programmers**, Tammy Noergaard, Elsevier Publication, 2005
3. **Embedded C programming**, Barnett, Cox & O'cull, Thomson (2005).

**UIS022E: LINUX KERNEL PROGRAMMING**  
**3 CREDITS (3-0-0)**

**UNIT I**

1. INTRODUCTION TO THE KERNEL **3 hrs**

Important data structures, main algorithms, Implementation of system calls,

2. MEMORY MANAGEMENT **7 hrs**

The architecture independent memory model, The virtual address space of a process, Blocking device caching, Paging under Linux.

**UNIT II**

3.. INTERPROCESS COMMUNICATION **4 hrs**

Synchronization in the kernel, Communication via files, pipes, debugging using ptrace, System V IPC, IPC with sockets

4. THE LINUX FILE SYSTEM **6 hrs**

Basics principles, The representation of file systems in the kernel, The EX2 file system, The proc file system

**UNIT III**

5. DEVICE DRIVERS UNDER LINUX **5 hrs**

Character and block devices, Hardware, Polling, interrupts, and waiting queues, Implementing a driver, Dynamic and static drivers

6. NETWORK IMPLEMENTATION **5 hrs**

Introduction and overview, Important data structures, Network devices under Linux

**UNIT IV**

7. MODULES AND DEBUGGING **5 hrs**

What are modules?, Implementation in the kernel, The meaning of object sections, for modules and kernels, Parameter transfer and examples for modules and kernels, What can be implemented as a module? The kernel daemon, Simple date swapping between modules, An example module, Debugging

8. MULTIPROCESSING **5 hrs**

The Intel multiprocessor specification, Problems with multiprocessor system, Changes to the kernel, Atomic operations, Spin locks

**Text Books:**

1. Michael Beck , Linux Kernel Programming, Pearson Education, Third Edition  
[Chapters:3,4,5,6,7,8,9,10]

**Reference Books:**

1. Daniel P. Bovet, Marco Cesati, Understanding the Linux kernel, O'Reilly Media, 2<sup>nd</sup> Edition, 2002.

**UIS023E: NEURAL NETWORKS**  
**3 CREDITS (3-0-0)**

**UNIT - 1**

**ARTIFICIAL NEURAL SYSTEM:** Neural computation, classifiers, Approximation and Autonomous drivers, Simple memory, Restoration patterns, Optimizing networks. Memory-based learning, Hebbian learning, Competitive learning.

**FUNDAMENTAL CONCEPTS AND MODELS OF ARTIFICIAL NEURAL SYSTEM:**

Biological neuron and their artificial models: biological neuron, Mc-Culloch-Pitts Neuron model, Neuron modeling of artificial neural systems. Models of artificial neural networks; neural processing; learning and Adaptation; neural network learning rules, Overview of neural networks.

**10 Hours**

**UNIT – 2**

**SINGLE-LAYER PERCEPTRON CLASSIFIERS:**

Classification model, Features, and Decision regions; Discriminant functions; Linear machine and Minimum distance classification; Non parametric training concept; Training and classification using the discrete perceptron algorithm and example, classifier AI nature of the learning process, Statistical learning theory, Single Layer continuous perceptron networks for linearly separable classifications; Multicategory Single Layer perceptron networks.

**10 Hours**

**UNIT - 3**

**MULTILAYER FEEDFORWARD NETWORKS – :** Linearly nonseparable pattern classification; Delta learning rule for Multiperceptron layer; Generalized Delta learning rule; Feedforward recall and Error back propagation Training; Learning factors; Classifying and Expert layered networks; Functional link networks.

**10 Hours**

**UNIT – 4**

**SINGLE-LAYER FEEDBACK NETWORKS :**

Basic concepts of Dynamical systems; Mathematical Foundations of Discrete-time Hopfield networks; Mathematical Foundations of Gradient type Hopfield networks; Transient Response of Continuous-time networks; Relaxation modeling in Single Layer feedback networks; Example solutions of Optimization problems.

**10 Hours**

**TEXT BOOK:**

1. **An Introduction to ARTIFICIAL NEURAL SYSTEM** – Jacek M. Zurada, A Jaico Publishing House

**REFERENCE BOOKS:**

- 1) **Understanding Neural Networks and Fuzzy Logic** –Stamatios V. Kartalopoulos, IEEE press
- 2) **An Introduction to Game Theory** – Martin Osborne, Oxford University Press, Indian Edition, 2004.
- 3) **Game Theory: Analysis of Conflict** – Roger B. Myerson, Harvard University Press, 1997.
- 4) **Microeconomic Theory** – Andreu Mas-Colell, Michael D. Whinston, and Jerry R. Green, Oxford University Press, New York, 1995.
- 5) **Game Theory and Strategy** – Philip D. Straffin, Jr., The Mathematical Association of America, January 1993.

**UIS024E: PRINCIPLES OF PROGRAMMING LANGAUAAGES**  
**3 CREDITS (3-0-0)**

**UNIT I**

**Introduction**

**4 Hrs**

Toward higher level languages, Programming paradigms, Language implementation: Bridging the gap, Expression notations, Abstract syntax trees.

**Types : Data Representation**

**6 Hrs**

Elementary data types – Data objects, Variables and Constants, Data types, Declarations, Type checking and type conversion. Numeric data types, Enumerations, Booleans, characters. Structural data types- Structured data objects and data types, Specification of data structure types, Implementation of data structure types, Vectors and arrays, Records, Type equivalence.

**UNIT II**

**Imperative Programming**

**6 Hrs**

Basic statements, Structured sequence control, Handling special cases in loops, Programming with invariants, Proof rules for partial correctness. Procedure activations – simple call-return subprograms, Parameter passing methods, Scope rules for names, Nested scopes in the source text, Activation records, Lexical scope: Procedures as in C.

**Object- Oriented Programming**

**4 Hrs**

Program design with modules, What is an object ?, Object oriented thinking, Inheritance, Object oriented programming in C++, Derived classes and information hiding.

**UNIT III**

**Functional Programming**

**10 Hrs**

Elements of functional programming – A little language of expressions, Types: values and operations, Approaches to expression evaluation, Lexical scope, Type checking. Functional programming in a typed language – Exploring a list, Function declaration by cases, Function as first – class values, ML: Implicit types, Data types, Exception handling in ML. Functional programming with lists – Scheme, a dialect of lisp, The structure of lists, List manipulation.

**UNIT IV**

**Logic Programming**

**10 Hrs**

Computing with relations, Introduction to prolog, Data structures in prolog, Programming techniques, Control in prolog.

**TEXT BOOKS:**

1. Ravi sethi, Programming languages, 2<sup>nd</sup> Edition, Pearson Education.  
[Chapters: 1.1,1.3, 1.4,2.1,2.2,2.3.4,3.5,3.6,5.2 to 5.6, 6.3, 7.1 to 7.4,7.6,8.1,8.2,8.4,8.5,8.6,9.1 to 9.6, 10.1 to 10.3, 11.1 to 11.5]
2. Terrence W.Pratt, Programming languages Design and implementation, 4<sup>th</sup> Edition, Pearson Education/PHI.  
[Chapters: 5.1.1 to 5.1.4,5.2.1 to 5.2.4, 6.1.1 to 6.1.3, 6.1.5, 6.1.6, 6.4.1, 8.3.1, 8.3.2, 9.1.1)

**REFERENCE BOOKS:**

- 1) Allen Tucker, Robert Nonan, Programming languages Tata McGraw-Hill, 2007  
Second Edition

**UIS025E: SOFT COMPUTING**  
**3 CREDITS (3-0-0)**

**UNIT I: (10 hrs)**

**Introduction:** Hard computing,-Features of hard computing, Soft computing - Features of soft computing, Hybrid Computing. **Optimization and Some Traditional Methods:** Introduction to Optimization - A practical example, Classification of optimization problems, principle of optimization, Duality principle, Traditional methods of optimization – Exhaustive search method, Random walk method, Steepest descent method, Drawbacks of traditional optimization methods.

**UNIT II: (10 hrs)**

**Introduction to Genetic Algorithms (GA):** Working cycle of a Genetic Algorithm, Binary coded GA, GA - parameters setting, Constraints handling in GA, Advantages & Disadvantages of GA. **Some Specialized Genetic Algorithms:** Real Coded GA, Micro GA, Visualized interactive GA, Scheduling GA.

**UNIT III: (10 hrs)**

**Introduction to Fuzzy Sets:** Crisp Sets, Fuzzy sets. **Fuzzy Reasoning and clustering:** Introduction, Fuzzy logic controller, Fuzzy clustering.

**UNIT IV: (10 hrs)**

**Fundamentals of Neural Networks:** Introduction, Static vs. Dynamic Neural Networks, Training of Neural Networks. **Some Examples of Neural Networks:** Multi Layer Feed Forward Neural Network (MLFFNN), Radial Basis Function Network (RBFN), Self organizing map(SOM), Recurrent Neural Networks(RNNs).

**Text Book**

- 1) D.K. Pratihari, “Soft Computing”, Narosa Publishing House, New Delhi, (Chapters 1, 2, 3, 4, 5, 6, 7, 8)

**Reference Books**

- 1) R. Rajasekaran and G. A. Vijayalakshmi Pai, “Neural Networks, Fuzzy Logic, and Genetic Algorithms: Synthesis and Applications”, Prentice Hall of India, New Delhi, 2003
- 2) D. E. Goldberg, “Genetic Algorithms in Search, Optimization, and Machine learning”, Addison-Wesley, Reading, MA, 1989
- 3) T. Ross, “Fuzzy Logic with Engineering Applications”, Tata McGraw Hill, New Delhi, 1995
- 4) B. Yegnanarayana, “Artificial Neural Networks”, Prentice Hall of India, New Delhi, 1999

**UIISO26E: WEB PROGRAMMING**  
**3 CREDITS (3-0-0)**

**UNIT – 1**

**FUNDAMENTALS OF WEB, HTML, AND XHTML**

Internet, WWW, Web Browsers and Web Servers, URLs, MIME, HTTP, Security, The web programmer's toolbox.

**XHTML:** Origins and evolution of HTML and XHTML, Basic Syntax, Standard XHTML, document structure, Basic text markup.

**XHTML-** Images; Hypertext links, Lists, Tables, Forms, Frames, Differences between HTML and XHTML.

**11 Hours**

**UNIT – 2**

**JAVASCRIPT**

**JAVASCRIPT:** Overview, Object orientation and JavaScript, General Syntactic characteristics, Primitives, operations and expressions, Screen o/p and keyboard i/p, Control statements, Arrays, Functions, Constructor, Pattern Matching using regular expressions, Errors in Scripts, Examples..

**10 Hours**

**UNIT – 3**

**JAVASCRIPT AND HTML DOCUMENTS, DYNAMIC DOCUMENTS WITH JAVASCRIPT**

**JAVASCRIPT AND HTML DOCUMENTS:** The JavaScript execution environment, The Document Object Model, Element access in JavaScript, Events and Event Handling, Handling events from the Body elements, Button elements, Text Box and Password elements.

**DYNAMIC DOCUMENTS WITH JAVASCRIPT:** Introduction to dynamic documents, Positioning and moving elements, element visibility, changing colors and fonts, Dynamic Content, Stacking elements, Locating the mouse cursor, reacting to a mouse click, slow movement of elements, Dragging and dropping elements.

**10 Hours**

**UNIT – 4**

**SSI, PHP, PERL-CGI PROGRAMMING**

**SSI:** Introduction – How it works, Tutorial

**PHP:** Introduction, Embedding PHP into HTML, Language Syntax – Variables, data types, Web variables, operators, Flow-control constructs.

**PERL-CGI Programming:** Origin and uses of Perl, Scalars and their operations, Assignment statements, simple i/p and o/p, Control statements, Fundamentals of Arrays, Hashes, References.

**The CGI:** CGI Linkage, Query String Format, CGI.pm module.

**11 Hours**

### **TEXT BOOKS**

- 1) **Programming the World Wide Web** – Robert W. Sebesta, 4<sup>th</sup> Edition, Pearson Education, 2008.
- 2) **Open Source Web Development with LAMP** – James Lee and Brent Ware, Addison Wesley/Pearson Education Inc. 2003.

### **REFERENCE BOOKS**

- 1) **Web Programming Building Internet Applications** – Chris Bates, 3<sup>rd</sup> Edition, Wiley India, 2006.
- 2) **Internet and World Wide Web How to Program** – M. Deitel, P.J. Deitel, C.B. Goldberg, 3<sup>rd</sup> Edition, Pearson Education/PHI, 2004.
- 3) **The Web Warrior Guide to Web Programming** – Xue Bai et al, Thomson, 2003.

**UIS027E: MOBILE COMPUTING  
3CREDITS (3-0-0)**

**UNIT I**

**Introduction:** Applications; A short history of wireless communication. **Wireless Transmission:** Frequency for radio transmission, Signals, Antennas, Signal propagation, Multiplexing, Modulation, Spread spectrum, Cellular systems. **Medium Access Control:** Motivation for a specialized MAC: Hidden and Exposed terminals. Near and Far terminals; SOMA, FOMA, TOMA: Fixed TOM, Classical Aloha, Slotted Aloha, Carrier sense multiple access, Demand assigned multiple access, PRMA packet reservation multiple access, PRMA packet reservation multiple access, reservation TOMA, Multiple access with collision avoidance, Polling, Inhibit sense multiple access; CDMA: Spread Aloha multiple access.

**10 Hours**

**UNIT II**

**GSM and Similar Architectures:** GSM – Services and System Architectures, Radio Interfaces, General Packet Radio Services. **Wireless Medium Access Control and CDMA – Based Communication** Medium Access Control, Introduction to CDMA – Based Systems, OFDM. **Mobile IP Network Layer:** IP and Mobile IP Network Layers Packet Delivery and Handover Management, Registration, Route Optimization, Dynamic Host Configuration Protocol. **Mobile Transport Layer:** Indirect TCP, Snooping TCP, Mobile TCP.

**10 Hours.**

**UNIT III**

**Databases:** Database Hoarding Techniques, Data Caching, Client – Server Computing and Adaptation, Transactional Models, Query Processing, **Data Dissemination and Broadcasting Systems:** Communication Asymmetry, Classification of Data – Delivery Mechanisms, Data Dissemination Broadcast Models, Digital Audio Broadcasting and Digital Video Broadcasting. **Data Synchronization in Mobile Computing Systems:** Synchronization, Synchronization Protocols, SyncML – Synchronization Language for Mobile Computing.

**10 Hours**

**UNIT IV**

**Mobile Devices, Server and Management:** Mobile agent, Application Server, Gateways, Portals, Service Discovery, Device Management, Mobile File Systems, Security. **Wireless LAN, Mobile Internet Connectivity and Personal Area Network:** Wireless LAN (WiFi) Architecture and Protocol Layers, WAP 1.1 and WAP 2.0 Architectures, Bluetooth – enabled Devices Network, Zigbee. **Mobile Operating Systems:** Operating System, PalmOS, Windows CE, Symbian OS, Linux for Mobile Devices.

**10 Hours**

**TEXT BOOK:**

- 1) Raj Kamal, “**Mobile Computing**”, Oxford University Press, 2007.

**Reference Books:**

- 1) Asoke Talkukder, Roopa R. Yavagal, “**Mobile Computing – Technology, Applications and Service Creation**”, Tata McGraw Hill, 2007.
- 2) Schiller, “**Mobile Communication**”, Pearson Publication, 2004.

**UIS028: COMPILER DESIGN**  
**3 CREDITS (3-0-0)**

*UNIT -I*

**INTRODUCTION, LEXICAL ANALYSIS:** Language processors; The structure of a Compilers; The evolution of programming languages; The science of building a compiler; Applications of Compiler technology; Programming language basics.

**Lexical analysis:** The Role of Lexical Analyzer; Input Buffering; Specifications of Tokens; Recognition of Tokens.

**10 Hours**

**UNIT –II**

**SYNTAX ANALYSIS:** Introduction; Context-free Grammars; Writing a Grammar; Top-down Parsing, Bottom-up Parsing; Introduction to LR Parsing: Simple LR.

**10 Hours**

**UNIT –III**

**SYNTAX-DIRECTED TRANSLATION:** Syntax-Directed definitions; Evaluation order for SDDs; Applications of Syntax-directed translation.

**INTERMEDIATE CODE GENERATION:** Variants of syntax trees; Three address code; **Types and declarations** – Type Expressions, Type equivalence, Declarations, **Type checking** – Rules for Type Checking, Type conversions; Control flow; Back patching.  
**10 Hours**

**UNIT –IV**

**RUN-TIME ENVIRONMENTS:** Storage Organization; Stack allocation of space; Access to non-local data on the stack; Heap management.

**CODE GENERATION:** Issues in the design of Code Generator; The Target language; Addresses in the target code; Basic blocks and Flow graphs; Optimization of basic blocks.  
**10 Hours**

**TEXT BOOK:**

1. **Compilers- Principles, Techniques and Tools** - Alfred V Aho, Monica S. Lam, Ravi Sethi, Jeffrey D Ullman, 2nd Edition, Addison- Wesley, 2007. (1.1 – 1.6; 3.1 – 3.4, 3.8; 4.1 – 4.6; 5.1 – 5.3; 6.1 – 6.3(6.3.1 - 6.3.3); 6.5 (6.5.1-6.5.2) – 6.7; 7.1 – 7.4; 8.1 – 8.5.

**REFERENCE BOOKS:**

1. **Crafting a Compiler with C** - Charles N. Fischer, Richard J. leBlanc,Jr, Pearson Education, 1991.
2. **Modern Compiler Implementation in C** - Andrew W Apple, Cambridge University Press, 1997.
3. **Compiler Construction Principles & Practice** - Kenneth C Loudon, Thomson Education, 1997.

## UIS029E: C# PROGRAMMING AND .NET

3 CREDITS (3-0-0)

### UNIT I

**THE PHILOSOPHY OF .NET:** Understanding the Previous State of Affairs, The .NET Solution, The Building Block of the .NET Platform (CLR,CTS, and CLS), The Role of the .NET Base Class Libraries, What C# Brings to the Table, An Overview of .NET Binaries ( aka Assemblies ), Intrinsic CTS Data Types, Understanding the Common Language Specification, Understanding the Common Language Runtime A tour of the .NET Namespaces, Increasing Your Namespace Nomenclature, Deploying the .NET Runtime, **BUILDING C# APPLICATIONS:** The Role of the Command Line Compiler (csc.exe), Building C # Application using csc.exe Working with csc.exe Response Files, Generating Bug Reports , Remaining C# Compiler Options, The Command Line Debugger (cordbg.exe) Using the, Visual Studio .NET IDE, Other Key Aspects of the VS.NET IDE, C# “Preprocessor:” Directives, An Interesting Aside: The System .Environment Class.

10 Hours

### UNIT II

**C# LANGUAGE FUNDAMENTALS:** The Anatomy of a Basic C# Class, Creating objects: Constructor Basics, The Composition of a C# Application, Default Assignment and Variable Scope, The C# Member Initialization Syntax, Basic Input and Output with the Console Class, Understanding Value Types and Reference Types, The Master Node: System, Object, The System Data Types (and C# Aliases), Converting Between Value Types and Reference Types: Boxing and Unboxing, Defining Program Constants, C# Iteration Constructs, C# Controls Flow Constructs, The Complete Set of C# Operators, Defining Custom Class Methods, Understating Static Methods, Methods Parameter Modifies, Array Manipulation in C #, String Manipulation in C#, C# Enumerations, Defining Structures in C#, Defining Custom Namespaces.

10 Hours

### UNIT III

**OBJECT- ORIENTED PROGRAMMING WITH C#:** Forms Defining of the C# Class, Definition the “Default Public Interface” of a Type, Recapping the Pillars of OOP, The First Pillars: C#'s Encapsulation Services, Pseudo-Encapsulation: Creating Read-Only Fields, The Second Pillar: C#'s Inheritance Supports, keeping Family Secrets: The “Protected” Keyword, Nested Type Definitions, The Third Pillar: C #'s Polymorphic Support, Casting Between. **EXCEPTIONS AND OBJECT LIFETIME:** Ode to Errors, Bugs, and Exceptions, The Role of .NET Exception Handling, the System.Exception Base Class, Throwing a Generic Exception, Catching Exception, CLR System – Level Exception (System.System Exception), Custom Application-Level Exception (System.System Exception), Handling Multiple Exception, The Family Block, the Last Chance Exception Dynamically Identifying Application – and System Level Exception Debugging System Exception Using VS. NET, Understanding Object Lifetime, the CIT of “new”, The Basics of Garbage Collection,, Finalization a Type, The Finalization Process, Building an Ad Hoc Destruction Method, Garbage Collection Optimizations, The System. GC Type.

10 Hours

### UNIT IV

**INTERFACES AND COLLECTIONS:** Defining Interfaces Using C# Invoking Interface Members at the object Level, Exercising the Shapes Hierarchy, Understanding Explicit Interface Implementation, Interfaces As Polymorphic Agents, Building Interface Hierarchies, Implementing, Implementation, Interfaces Using VS .NET, understanding the IConvertible Interface, Building a Custom Enumerator (IEnumerable and Enumerator), Building Cloneable objects (ICloneable), Building Comparable Objects (I Comparable), Exploring the

system. Collections Namespace, Building a Custom Container (Retrofitting the Cars Type). **CALLBACK INTERFACES, DELEGATES, AND EVENTS:** Understanding Callback Interfaces, Understanding the .NET Delegate Type, Members of System. Multicast Delegate, The Simplest Possible Delegate Example, , Building More a Elaborate Delegate Example, Understanding Asynchronous Delegates, Understanding (and Using)Events.

**10 Hours**

**TEXT BOOKS:**

1. **Pro C# with .NET 3.0** - Andrew Troelsen, Special Edition, Dream Tech Press, India, 2007.
2. **Programming in C#** - E. Balagurusamy, 5th Reprint, Tata McGraw Hill, 2004.

**REFERENCE BOOKS:**

1. **Inside C#** - Tom Archer, WP Publishers, 2001.
2. **The Complete Reference C#** - Herbert Schildt, Tata McGraw Hill, 2004

## UIS030E: SOFTWARE TESTING

3 CREDITS (3-0-0)

### UNIT – 1

10 HRS

**BASICS OF SOFTWARE TESTING:** Human Errors and Testing; Software Quality; Requirements, Behavior and Correctness; Correctness versus Reliability; Testing and Debugging; Test Metrics. Software and Hardware Testing; Testing and Verification; Defect Management; Execution History; Test-generation Strategies, Static Testing. Model-Based Testing and Model Checking; Control-Flow Graph; Types of Testing; The saturation effect.

### UNIT – 2

10 HRS

**TEST GENERATION FROM REQUIREMENTS:** Introduction; The Test-Selection Problem; Equivalence Partitioning; Boundary Value Analysis; Category-Partition Method. Cause-Effect Graphing, Test Generation from Predicates.

### UNIT – 3

11 HRS

**STRUCTURAL TESTING:** Overview; Statement testing; Branch testing; Condition testing, Path testing; Procedure call testing; Comparing structural testing criteria; The infeasibility problem.

**DEPENDENCE, DATA FLOW MODELS, AND DATA FLOW TESTING:** Definition-Use pairs; Data flow analysis; Classic analyses; From execution to conservative flow analysis; Data flow analysis with arrays and pointers; Inter-procedural analysis; Overview of data flow testing; Definition-Use associations; Data flow testing criteria; Data flow coverage with complex structures; The infeasibility problem.

### UNIT – 4

11 HRS.

**TEST CASE SELECTION AND ADEQUACY,:** Overview; Test specification and cases; Adequacy criteria; Comparing criteria;

#### **PROCESS:**

Integration and component-based software testing: Overview; Integration testing strategies; Testing components and assemblies. System, Acceptance and Regression Testing: Overview; System testing; Acceptance testing; Usability; Regression testing; Regression test selection techniques; Test case prioritization and selective execution.

#### **TEXT BOOKS:**

1. **Foundations of Software Testing** - Aditya P Mathur, Pearson Education, 2008. (ch 1, 2)
2. **Software Testing and Analysis Process Principles and Techniques** – Mauro Pezze, Michal Young, Wiley India, 2008. (chapter 6,9,12,13,20,21,22)

#### **REFERENCE BOOKS:**

1. **Software Testing Principles and Practices** - Srinivasan Desikan, Gopaldaswamy Ramesh, 2<sup>nd</sup> Edition, Pearson, 2007.
2. **Software Testing** - Ron Patton, 2<sup>nd</sup> edition, Pearson, 2004.
3. **The Craft of Software Testing** - Brian Marrick, Pearson, 1995. Introduction to Information Storage and Management

**UIS031E: STORAGE TECHNOLOGIES**  
**3 Credits(3-0-0)**

**Unit – I**

**10 Hrs**

1. Information Storage, Evolution of Storage Technology and Architecture, Data center infrastructure,  
Key challenges in Managing information, Information life cycle.
- 1. Storage System Environment**  
Components of a Storage system environment, Disk drive components, Disk drive performance, Application requirements and disk performance.

**Unit -II**

**10 Hrs**

- 2. Storage Systems Architecture**  
Implementation of RAID, RAID Array Components, RAID Levels, RAID Comparison, RAID Impact on Disk performance.
- 3. Networked Storage**  
Types of DAS, Disk drive Interfaces, Storage Area Networks. (SAN) : Fiber channel :Overview, Components of SAN, FC Connectivity, Fibre channel ports, FC Architecture, Zoning, FC Topologies, Network Attached Storage (NAS): Components of NAS, NAS Implementations, NAS File sharing Protocols.

**Unit -III**

**10 Hrs**

- 6. Storage Virtualization**  
Forms of Virtualization, SNIA Storage Virtualization Taxonomy, Storage Virtualization Configurations, Storage Virtualization Challenges, Types of Storage Virtualization.

**Unit -IV**

**10 Hrs**

- 7. Content Addressed Storage**  
Types of Archives, Features and Benefits of CAS, CAS Architecture, Object Storage and Retrieval in CAS.
- 8. Managing the Storage Infrastructure**  
Monitoring the Storage Infrastructure, Storage Management activities, developing an Ideal Solution.

**Text Book:**

1. G.Somasundaram,Alok Shrivastava ,EMC Education Services, Information Storage and Management , Wiley Publishing, Inc 2009.

**Reference Books:**

1. EMC Education services, storage technology Foundations Student guide Vol-1, Vol-2, 2007.
2. Spalding, R., Storage Networks The Complete Reference, Tata McGrawHill , Osborne, 2003.
3. Farely, M, Building Storage Networks, Tata McGraw Hill ,Osborne, 2001
4. Gupta, M., Storage Area Network Fundamentals, Pearson Education Limited, 2002.

**Question Paper Pattern for SEE (for every theory subject):**

Total of eight questions, two questions from each unit carrying twenty marks each, are to be set from each unit for SEE examination. Each FULL question should have a maximum of four sub questions. A student is expected to answer FIVE full questions choosing at least one FULL question from each unit.

**Evaluation scheme for theory subjects:**

A student will be evaluated in the subject through Continuous Internal Evaluation (CIE) and Semester End Examinations (SEE).

The CIE is for 50 marks and consists of 2 tests of 22.5 marks each and an assignment for 5 marks. The CIE tests are conducted for 1.5 hr for 45 marks, and marks obtained are scaled down to 22.5.

The SEE for 50 marks is conducted as 3 hrs exam for 100 marks, then the marks obtained are scaled down to 50.

**Evaluation scheme for laboratory subjects:**

A student will be evaluated in the subject through Continuous Internal Evaluation (CIE) and Semester End Examinations (SEE).

The CIE is for 50 marks. The students are evaluated for 30 marks for the conduct of the laboratory assignments and journal writing. The student is evaluated through an internal lab test for the remaining 20 marks.

The SEE for 50 marks is through a lab exam of 3 hrs duration. In the lab exam student is evaluated as following:

- a. 25% of total marks for initial write-up and overall submission: 12.5 marks.
- b. 50% of total marks for conduct/completion of the task: 25 marks.
- c. 25% of total marks for viva-voce on the lab/subject: 12.5 marks.

The student is awarded 'S' to 'F' grade based on his/her performance in CIE and SEE taken together.