

St. PETER'S UNIVERSITY

St. Peter's Institute of Higher Education and Research
(Declared under section 3 of UGC Act 1956)
Avadi, Chennai – 600 054.



M.Tech. (INFORMATION TECHNOLOGY)

(I to IV SEMESTERS)

REGULATIONS AND SYLLABI

(REGULATIONS – 2013)

**(With a retrospective amendment in the credits from the batch
of students admitted in 2014-15)**

M.TECH. (INFORMATION TECHNOLOGY) PROGRAMME

Regulations and Syllabi

(Effective from the Academic Year 2013-'14)

(With a retrospective amendment in the credits from the batch of students admitted in 2014-15)

- 1. Eligibility:** Candidates who passed B.E. / B.Tech. (CSE / IT) of the University or any other equivalent examination thereto are eligible for admission to Two Year M.Tech.(Information Technology) Programme.
- 2. Duration:** Two Years Comprising 4 Semesters. Each semester has a minimum 90 working days with a minimum of 5 hours a day a minimum of 450 hours per Semester. Candidates who have completed the duration of the programme of study are permitted to appear for the arrear subjects examinations, if any within two years after the duration of the programme.
- 3. Medium:** English is the medium of instruction and examination.
- 4. Weightage for Internal and End Assessment:** The weightage for Continuous Assessment (CA) and End Assessment (EA) is 25:75 unless the ratio is specifically mentioned in the scheme of Examinations. The question paper is to be set for a maximum of 100 Marks.
- 5. Choice Based Credit System:** Choice Based Credit System is followed with one credit equivalent to one hour for a theory paper and two hours for a practical per week in a cycle of 18 weeks (that is, one credit is equal to 18 hours for each theory paper and one credit is equal to 36 hours for a practical in a semester) in the Time Table. The total credit for the programme (4 semesters) is 90.

6. Scheme of Examinations

I Semester

Code No.	Course Title	L	T	P	C
Theory					
113ITPT01	Applied Probability and Statistics	3	1	0	4
113ITPT02	Advanced Data Structures and Algorithms	3	0	0	3
113ITPT03	Multicore Architectures	3	0	0	4
113ITPT04	Internetworking Technologies	3	0	0	4
113ITPT05	Object Oriented Software Engineering	3	0	0	4
113ITPT06	Advanced Databases	3	0	0	3
Practical					
113ITPP01	Advanced Data Structures Laboratory	0	0	3	2
113ITPP02	Internetworking Laboratory	0	0	3	2
113ITPP03	Mini Project	0	0	2	1
Total		18	1	8	27

II Semester

Code No.	Course Title	L	T	P	C
Theory					
213ITPT01	Web Technologies	3	0	0	3
213ITPT02	Cloud Computing	3	0	0	4
213ITPT03	Network and Information Security	3	0	0	4
213ITPT04	Data Warehousing and Data Mining	3	0	0	4
213ITPT08	Elective I: XML and Web Services	3	0	0	4
213ITPT16	Elective II: Big Data Analytics	3	0	0	4
Practical					
213ITPP01	Web Technology Laboratory	0	0	3	2
213ITPP02	Cloud Computing Laboratory	0	0	3	2
213ITPP03	Technical Seminar*	-	-	-	-
Total		18	0	6	27

* Satisfactory report from the HoD

III SEMESTER

Code No.	Course Title	L	T	P	C
THEORY					
313ITPT01	Soft Computing	3	0	0	4
313ITPT04	Elective III: Web Mining	3	0	0	4
313ITPT17	Elective IV: Spatial Database	3	0	0	4
PROJECT					
313ITPP01	Project Phase I*	0	0	12	8
	Viva Voce				
Total		9	0	12	20

* Candidates who have completed Project work (Phase I) successfully are eligible for Project Work (Phase - II) Examination.

IV SEMESTER

Code No.	Course Title	L	T	P	C
413ITPP01	Project Phase II*	0	0	24	16
	Viva Voce				
Total		0	0	24	16

LIST OF ELECTIVES

II Semester

Code No.	Course Title	L	T	P	C
ELECTIVE-I					
213ITPT05	Software Metrics and Reliability	3	0	0	4
213ITPT06	Network Management	3	0	0	4
213ITPT07	Bio Informatics	3	0	0	4
213ITPT08	XML and Web Services	3	0	0	4
213ITPT09	Enterprise Application Integration	3	0	0	4
213ITPT10	Video Analytics	3	0	0	4
ELECTIVE-II					
213ITPT11	Software Project Management	3	0	0	4
213ITPT12	Mobile and Pervasive Computing	3	0	0	4
213ITPT13	Principles of Programming Languages	3	0	0	4
213ITPT14	Multimedia Technologies	3	0	0	4
213ITPT15	Automata Theory and Compiler Design	3	0	0	4
213ITPT16	Big Data Analytics	3	0	0	4

III Semester

Code No.	Course Title	L	T	P	C
ELECTIVE-III					
313ITPT02	Software Quality and Testing	3	0	0	4
313ITPT03	Wireless Adhoc and Sensor Networks	3	0	0	4
313ITPT04	Web Mining	3	0	0	4
313ITPT05	Image Processing and Pattern Analysis	3	0	0	4
313ITPT06	Intelligent Agents	3	0	0	4
313ITPT07	Internet of Things	3	0	0	4
313ITPT08	Web Engineering	3	0	0	4
313ITPT09	Network infrastructure and cyber computing	3	0	0	4
313ITPT10	Web Advertising and Marketing	3	0	0	4
ELECTIVE-IV					
313ITPT11	Parallel Programming Paradigms	3	0	0	4
313ITPT12	Social Network Analysis	3	0	0	4
313ITPT13	Knowledge Engineering	3	0	0	4
313ITPT14	Energy Aware Computing	3	0	0	4
313ITPT15	4G Technologies	3	0	0	4
313ITPT16	Performance Evaluation of Computer Systems	3	0	0	4
313ITPT17	Spatial Database	3	0	0	4
313ITPT18	Management for information Technology	3	0	0	4
313ITPT19	Introduction to Machine Learning	3	0	0	4

7. Passing Requirements: The minimum pass mark (raw score) be 50% in End Assessment (EA) and 50% in Continuous Assessment (CA) and End Assessment (EA) put together. No minimum mark (raw score) in Continuous Assessment (CA) be prescribed unless it is specifically mentioned in the Scheme of Examination.

8. Grading System: Grading System on a 10 Point Scale be followed with 1 mark = 0.1 Grade point to successful candidates as given below.

CONVERSION TABLE

(1 mark = 0.1 Grade Point on a 10 Point Scale)

Range of Marks	Grade Point	Letter Grade	Classification
90 to 100	9.0 to 10.0	O	First Class
80 to 89	8.0 to 8.9	A	First Class
70 to 79	7.0 to 7.9	B	First Class
60 to 69	6.0 to 6.9	C	First Class
50 to 59	5.0 to 5.9	D	Second Class
0 to 49	0 to 4.9	F	Reappearance

Procedure for Calculation

Cumulative Grade Point Average (CGPA)	=	$\frac{\text{Sum of Weighted Grade Points}}{\text{Total Credits}}$
	=	$\frac{\sum (CA+EA) C}{\sum C}$
Where Weighted Grade Points in each Course	=	Grade Points (CA+EA) multiplied by Credits
	=	(CA+EA)C
Weighted Cumulative Percentage of Marks(WCPM)	=	CGPAx10

C- Credit, CA-Continuous Assessment, EA- End Assessment

9. Pattern of the Question Paper: The question paper for End Assessment will be set for three hours and for the maximum of 100 marks with following divisions and details.

Part A: 10 questions (with equal distribution to all units in the syllabus).
Each question carries 2 marks.

Part B: 5 questions with either or type (with equal distribution to all units in the syllabus). Each question carries 16 marks.
The total marks scored by the candidates will be reduced to the maximum prescribed in the Regulations.

10. Effective Period of Operation for the Arrear Candidates : Two Year grace period is provided for the candidates to complete the arrear examination, if any.

Registrar

11. Syllabus

113ITPT01 - APPLIED PROBABILITY AND STATISTICS

OBJECTIVES:

- To develop the ability to apply the concepts of Matrix theory and Linear programming in Engineering problems.
- To familiarize the students in calculus of variations
- To introduce the concepts of one dimensional and two dimensional random variables.

UNIT I CALCULUS OF VARIATION

Introduction – Euler’s equation – several dependent variables language’s equation of dynamics – Integrals involving derivatives higher than the first – problem with constraints – Direct methods and eigen value problems.

UNIT II MATRIX THEORY

Eigen values using QR transformations- generalized eigenvectors – canonical forms – singular value decomposition and applications – pseudo inverse – least square approximations.

UNIT III LINEAR PROGRAMMING PROBLEM

Graphical method – simplex method – Big M technique – Integer programming.

UNIT IV ONE DIMENSIONAL RANDOM VARIABLES

Random variables - Probability function – Moments – Moment generating functions and their properties – Binomial, Poisson, Geometric, Uniform, Exponential, Gamma and Normal distributions – Functions of a Random Variable.

UNIT V TWO DIMENSIONAL RANDOM VARIABLES

Joint distributions – Marginal and Conditional distributions – Functions of two dimensional random variables – Regression Curve – Correlation.

References

1. Gupta.A.S ,Calculus of variations with Applications,Prentice –Hall of India New Delhi,1997.
2. Bronson.R,“Matrix Operation” Schaums Outline Series ,Mc Graw Hill , New York ,1989.
3. K.H.Rosen ,Discrete Mathematics and its Applications,Mc Graw Hill Book Company , 1999.
4. Taha H.A, “Operation Research -An Introduction”, Prentice Hall of India, 2001.
5. Jay L. Devore, “Probability and Statistics For Engineering and the Sciences”, Thomson and Duxbury, 2002.
6. Gupta S.C. and Kapoor V.K.“Fundamentals of Mathematical Statistics”, Sultan and Sons, 2001.

113ITPT02 - ADVANCED DATA STRUCTURES AND ALGORITHMS

OBJECTIVES:

- To understand the principles of iterative and recursive algorithms.
- To learn the graph search algorithms.
- To study network flow and linear programming problems.
- To learn the hill climbing and dynamic programming design techniques.
- To develop recursive backtracking algorithms.
- To get an awareness of NP completeness and randomized algorithms.
- To learn the principles of shared and concurrent objects.
- To learn concurrent data structures.

UNIT I ITERATIVE AND RECURSIVE ALGORITHMS

Iterative Algorithms: Measures of Progress and Loop Invariants-Paradigm Shift: Sequence of Actions versus Sequence of Assertions- Steps to Develop an Iterative Algorithm-Different Types of Iterative Algorithms--Typical Errors-Recursion-Forward versus Backward- Towers of Hanoi-Checklist for Recursive Algorithms-The Stack Frame-Proving Correctness with Strong Induction- Examples of Recursive Algorithms-Sorting and Selecting Algorithms-Operations on Integers- Ackermann's Function- Recursion on Trees-Tree Traversals-Examples- Generalizing the Problem - Heap Sort and Priority Queues-Representing Expressions.

UNIT II OPTIMISATION ALGORITHMS

Optimization Problems-Graph Search Algorithms-Generic Search-Breadth-First Search-Dijkstra's Shortest-Weighted-Path -Depth-First Search-Recursive Depth-First Search-Linear Ordering of a Partial Order- Network Flows and Linear Programming-Hill Climbing-Primal Dual Hill Climbing- Steepest Ascent Hill Climbing-Linear Programming-Recursive Backtracking-Developing Recursive Backtracking Algorithm- Pruning Branches-Satisfiability

UNIT III DYNAMIC PROGRAMMING ALGORITHMS

Developing a Dynamic Programming Algorithm-Subtle Points- Question for the Little Bird-Subinstances and Subsolutions-Set of Subinstances-Decreasing Time and Space-Number of Solutions-Code. Reductions and NP-Completeness-Satisfiability-Proving NPCompleteness-3-Coloring- Bipartite Matching. Randomized Algorithms-Randomness to Hide Worst Cases-Optimization Problems with a Random Structure.

UNIT IV SHARED OBJECTS AND CONCURRENT OBJECTS

Shared Objects and Synchronization -Properties of Mutual Exclusion-The Moral -The Producer-Consumer Problem -The Readers-Writers Problem-Realities of Parallelization-Parallel Programming- Principles- Mutual Exclusion-Time- Critical Sections--Thread Solutions-The Filter Lock-Fairness-Lamport's Bakery Algorithm-Bounded Timestamps-Lower Bounds on the Number of Locations-Concurrent Objects- Concurrency and Correctness-Sequential Objects-Quiescent Consistency- Sequential Consistency-Linearizability- Formal Definitions- Progress Conditions- The Java Memory Model

UNIT V CONCURRENT DATA STRUCTURES

Practice-Linked Lists-The Role of Locking-List-Based Sets-Concurrent Reasoning- Coarse-Grained Synchronization-Fine-Grained Synchronization-Optimistic Synchronization- Lazy Synchronization-Non-Blocking Synchronization-Concurrent Queues and the ABA Problem-Queues-A Bounded Partial Queue-An Unbounded Total Queue-An Unbounded Lock-Free Queue-Memory Reclamation and the ABA Problem- Dual Data Structures- Concurrent Stacks and Elimination- An Unbounded Lock-Free Stack- Elimination-The Elimination Backoff Stack

OUTCOMES: Upon completion of the course, the students will be able to

1. Design and apply iterative and recursive algorithms.
2. Design and implement optimisation algorithms in specific applications.
3. Design appropriate shared objects and γ concurrent objects for applications.

4. Implement and apply concurrent linked lists, stacks, and queues.

REFERENCES:

1. Jeff Edmonds, "How to Think about Algorithms", Cambridge University Press, 2008.
2. M. Herlihy and N. Shavit, "The Art of Multiprocessor Programming", Morgan Kaufmann, 2008.
3. Steven S. Skiena, "The Algorithm Design Manual", Springer, 2008.
4. Peter Brass, "Advanced Data Structures", Cambridge University Press, 2008.
5. S. Dasgupta, C. H. Papadimitriou, and U. V. Vazirani, "Algorithms" , Mc GrawHill, 2008.
6. J. Kleinberg and E. Tardos, "Algorithm Design", Pearson Education, 2006.
7. T. H. Cormen, C. E. Leiserson, R. L. Rivest and C. Stein, "Introduction to Algorithms", PHI Learning Private Limited, 2012.
8. Rajeev Motwani and Prabhakar Raghavan, "Randomized Algorithms", Cambridge University Press, 1995.
9. A. V. Aho, J. E. Hopcroft, and J. D. Ullman, "The Design and Analysis of Computer Algorithms", Addison-Wesley, 1975.
10. A. V. Aho, J. E. Hopcroft, and J. D. Ullman, "Data Structures and Algorithms", Pearson, 2006.

113ITPT03 - MULTICORE ARCHITECTURES

OBJECTIVES:

- To understand the recent trends in the field of Computer Architecture and identify performance related parameters
- To appreciate the need for parallel processing
- To expose the students to the problems related to multiprocessing
- To understand the different types of multicore architectures
- To expose the students to warehouse-scale and embedded architectures

UNIT I FUNDAMENTALS OF QUANTITATIVE DESIGN AND ANALYSIS

Classes of Computers – Trends in Technology, Power, Energy and Cost – Dependability – Measuring, Reporting and Summarizing Performance – Quantitative Principles of Computer Design – Classes of Parallelism - ILP, DLP, TLP and RLP - Multithreading - SMT and CMP Architectures – Limitations of Single Core Processors - The Multicore era – Case Studies of Multicore Architectures.

UNIT II DLP IN VECTOR, SIMD AND GPU ARCHITECTURES

Vector Architecture - SIMD Instruction Set Extensions for Multimedia – Graphics Processing Units - Detecting and Enhancing Loop Level Parallelism - Case Studies.

UNIT III TLP AND MULTIPROCESSORS

Symmetric and Distributed Shared Memory Architectures – Cache Coherence Issues - Performance Issues – Synchronization Issues – Models of Memory Consistency - Interconnection Networks – Buses, Crossbar and Multi-stage Interconnection Networks.

UNIT IV RLP AND DLP IN WAREHOUSE-SCALE ARCHITECTURES

Programming Models and Workloads for Warehouse-Scale Computers – Architectures for Warehouse-Scale Computing – Physical Infrastructure and Costs – Cloud Computing – Case Studies.

UNIT V ARCHITECTURES FOR EMBEDDED SYSTEMS

Features and Requirements of Embedded Systems – Signal Processing and Embedded Applications – The Digital Signal Processor – Embedded Multiprocessors - Case Studies.

OUTCOMES:

Upon completion of the course, the students will be able to

- Identify the limitations of ILP and the need for multicore architectures
- Address the issues related to multiprocessing and suggest solutions
- Bring out the salient features of different multicore architectures and how they exploit parallelism
- Analyze the different types of inter connection networks
- Explore the architecture of GPUs, warehouse-scale computers and embedded processors

REFERENCES:

1. John L. Hennessey and David A. Patterson, " Computer Architecture – A Quantitative Approach", Morgan Kaufmann / Elsevier, 5th. edition, 2012.
2. Kai Hwang, "Advanced Computer Architecture", Tata McGraw-Hill Education, 2003
3. Richard Y. Kain, "Advanced Computer Architecture a Systems Design Approach", PHI, 2011.
4. David E. Culler, Jaswinder Pal Singh, "Parallel Computing Architecture : A Hardware/ Software Approach" , Morgan Kaufmann / Elsevier, 1997.

113ITPT04 - INTERNETWORKING TECHNOLOGIES

OBJECTIVES:

- To study the design and implementation of a socket based application using either TCP, UDP and SCTP.
- To understand SCTP sockets and its options
- To study the features of Raw Sockets
- To learn to develop the DNS
- To study the security features in socket programming
- To explore the usage of sockets options and the system calls needed to support unicast, broadcast and multicast applications
- To explore the emerging technologies in socket programming.

UNIT I APPLICATION DEVELOPMENT

Introduction to Socket Programming – Overview of TCP/IP Protocols –Introduction to Sockets –Iterative TCP programming – Iterative UDP programming – Concurrent programming – fork and exec - I/O multiplexing – I/O Models – select function – shutdown function – TCP echo Server (with multiplexing) – poll function – TCP echoClient (with Multiplexing) Multiplexing TCP and UDP sockets- Threaded servers – thread creation and termination – TCP echo server using threads – Mutexes – condition variables

UNIT II ELEMENTARY SCTP SOCKETS AND SOCKET OPTIONS

Introduction to SCTP- Interface Modules – SCTP functions- sctp_bindx, sctp_connectx, sctp_getpaddr, sctp_freepaddr, sctp_getladdr, sctp_freeladdr, sctp_sendmsg, sctp_rcvmsg, sctp_opt_info, sctp_peeloff, shutdown – Notifications - Socket options – getsockopt and setsockopt functions- Socket states – generic socket options – IP socket options – ICMP socket options – TCP socket options - .SCTP socket options – fcntl functions.

UNIT III RAW SOCKETS AND DNS

Ipv4 and Ipv6 interoperability - raw sockets – raw socket creation – raw socket output – raw socket input – ping program – trace route program - Domain name system – gethostbyname function – Ipv6 support in DNS – gethostbyadr function – getservbyname and getservbyport functions – Data link Access – BPF, DLPI,libpcap, libnet.

UNIT IV ADVANCED SOCKETS-I

Routing sockets – Datalink socket address structure – Reading and writing – sysctl operations – get_ifi_info function – Interface name and index functions- Key Management sockets – Reading and writing – Dumping Security Association Database – Creating static Security Association – Dynamically maintaining SA's – Broadcasting – Broadcast addresses – Unicast versus Broadcast – (Client) Application development for broadcasting – Race conditions – Multicasting – Multicast addresses- Multicasting versus Broadcasting on a LAN – Multicasting on a WAN – Source specific Multicast – Multicast socket options – mcast_join, (Client) Application development for multicasting – Receiving IP multicast infrastructure session announcements – Sending and receiving.

UNIT V ADVANCED SOCKETS-II

Advanced UDP sockets – Receiving flags, Destination IP address and Interface index – Datagram truncation – Using UDP instead of TCP – Adding reliability to UDP – Binding interface addresses – Concurrent UDP servers – Advanced SCTP sockets – Partial delivery - Notifications – Unordered data – Binding a subset of addresses – Determining peer and local address information – Finding an association Id given an IP address – Heartbeating and Address failure – Peeling off an association – Controlling timing – Using SCTP instead of TCP – Out of band data – Signal driven I/O.

OUTCOMES:

Upon completion of the course, the students will be able to

- Design and develop network applications using sockets system calls.
- Compare IPv4 and IPv6
- Explore the features of Stream Control Transmission Protocol (SCTP)
- Design and develop network applications using raw socket
- Design and develop Domain Name Service
- Incorporate the security features in the socket programming
- Work with various networking tools such as ping, traceroute to investigate a traffic flow in the network.
- Extend network applications for broadcasting and multicasting
- Create innovative network design by applying advanced socket concepts.

REFERENCES:

1. W. Richard Stevens, "Unix Network Programming Vol-I", Second Edition, Pearson Education, 1998.
2. D.E. Comer, "Internetworking with TCP/IP Vol- III", (BSD Sockets Version), Second Edition, Pearson Education, 2003.
3. Michael Donahoo, Kenneth Calvert, "TCP/IP Sockets in C, A practical guide for programmers", Second Edition, Elsevier, 2009
4. Forouzan, " TCP/IP Protocol Suite" Second Edition, Tata MC Graw Hill, 2003.

113ITPT05 - OBJECT ORIENTED SOFTWARE ENGINEERING

OBJECTIVES:

- To provide information about wider engineering issues that form the background to develop complex, evolving (software-intensive) systems
- To gain basic knowledge about object-oriented analysis and to familiarize UML concepts
- To study the requirements of various domain applications
- To design, implement and test the software in object oriented approach
- To discuss the issues in managing the software projects
- To explore the standards related to life cycle process

UNIT I INTRODUCTION

System Concepts – Software Engineering Concepts – Development Activities – Managing Software Development –Modelling with UML – Project Organization and Communication – Case Study

UNIT II REQUIREMENT ELICITATION AND ANALYSIS

Requirements Elicitation Concepts – Requirements Elicitation Activities – Managing Requirements Elicitation– Analysis Concepts – Analysis Activities – Managing Analysis - Case Study

UNIT III SYSTEM DESIGN

Decomposing the system – Overview of System Design – System Design Concepts – System Design Activities: Objects to Subsystems – System Design Activities: Addressing Design Goals – Managing System Design - Case Study

UNIT IV OBJECT DESIGN, IMPLEMENTATION AND TESTING

Object Design Overview – Reuse Concepts – Design Patterns – Reuse Activities – Managing Reuse – Interfaces Specification Concepts – Interfaces Specification Activities – Managing Object Design – Mapping Models to Code Overview – Mapping Concepts – Mapping Activities – Managing Implementation – Testing – Case Study

UNIT V MANAGING CHANGE

Rationale Management Overview – Rationale Concepts – Rationale Activities: From Issues to Decisions – Managing Rationale – Configuration Management Overview – Configuration Management Concepts – Configuration Management Activities – Managing Configuration Management – Project Management Overview – Project Management Activities – Standard for Developing Life Cycle Process (IEEE 1074) – Overview of Capability Maturity Model (CMM) – Life Cycle Models

OUTCOMES:

Upon Completion of the course, the students should be able to

- Apply Object Oriented Software Engineering approach in every aspect of software project
- Analyse the requirements from various domains
- Evaluate the relationships between Software Design and Software Engineering
- Adapt appropriate object oriented design aspects in the development process
- Implement and test the software project using object oriented approach
- Manage the issues regarding the decision making and changes in the different stage of software development
- Implement mini projects incorporating the principles of object oriented software engineering

REFERENCES:

1. Bernd Bruegge, Alan H Dutoit, Object-Oriented Software Engineering, Pearson Education, 2011.
2. Jacobson, Ivar & Christerson, Magnus & Jonsson, Patrik & Overgaard, Gunnar "Object Oriented Software Engineering", Pearson Education, Delhi, 2007
3. Craig Larman, Applying UML and Patterns, 3rd ed, Pearson Education, 2005.

113ITPT06 - ADVANCED DATABASES

OBJECTIVES:

- To learn the modeling and design of databases.
- To acquire knowledge on parallel and distributed databases and its applications.
- To study the usage and applications of Object Oriented database
- To understand the principles of intelligent databases.
- To understand the usage of advanced data models.
- To learn emerging databases such as XML, Cloud and Big Data.
- To acquire inquisitive attitude towards research topics in databases.

UNIT I PARALLEL AND DISTRIBUTED DATABASES

Database System Architectures: Centralized and Client-Server Architectures – Server System Architectures – Parallel Systems- Distributed Systems – Parallel Databases: I/O Parallelism – Inter and Intra Query Parallelism – Inter and Intra operation Parallelism – Design of Parallel Systems- Distributed Database Concepts - Distributed Data Storage – Distributed Transactions – Commit Protocols – Concurrency Control – Distributed Query Processing – Case Studies

UNIT II OBJECT AND OBJECT RELATIONAL DATABASES

Concepts for Object Databases: Object Identity – Object structure – Type Constructors – Encapsulation of Operations – Methods – Persistence – Type and Class Hierarchies – Inheritance – Complex Objects – Object Database Standards, Languages and Design: ODMG Model – ODL – OQL – Object Relational and Extended – Relational Systems: Object Relational features in SQL/Oracle – Case Studies.

UNIT III INTELLIGENT DATABASES

Active Databases: Syntax and Semantics (Starburst, Oracle, DB2)- Taxonomy- Applications- Design Principles for Active Rules- Temporal Databases: Overview of Temporal Databases- TSQ2- Deductive Databases: Logic of Query Languages – Data log- Recursive Rules- Syntax and Semantics of Datalog Languages- Implementation of Rules and Recursion- Recursive Queries in SQL- Spatial Databases- Spatial Data Types- Spatial Relationships- Spatial Data Structures-Spatial Access Methods- Spatial DB Implementation.

UNIT IV ADVANCED DATA MODELS

Mobile Databases: Location and Handoff Management - Effect of Mobility on Data Management - Location Dependent Data Distribution - Mobile Transaction Models - Concurrency Control - Transaction Commit Protocols- Multimedia Databases- Information Retrieval- Data Warehousing- Data Mining- Text Mining.

UNIT V EMERGING TECHNOLOGIES

XML Databases: XML-Related Technologies-XML Schema- XML Query Languages- Storing XML in Databases-XML and SQL- Native XML Databases- Web Databases- Geographic Information Systems- Biological Data Management- Cloud Based Databases: Data Storage Systems on the Cloud- Cloud Storage Architectures-Cloud Data Models- Query Languages- Introduction to Big Data-Storage-Analysis.

OUTCOMES:

Upon completion of the course, the students will be able to

- Select the appropriate high performance database like parallel and distributed database
- Model and represent the real world data using object oriented database
- Design a semantic based database to meaningful data access
- Embed the rule set in the database to implement intelligent databases
- Represent the data using XML database for better interoperability
- Handle Big data and store in a transparent manner in the cloud
- To solve the issues related to the data storage and retrieval

REFERENCES:

1. R. Elmasri, S.B. Navathe, "Fundamentals of Database Systems", Fifth Edition, Pearson Education/Addison Wesley, 2007.
2. Thomas Cannolly and Carolyn Begg, "Database Systems, A Practical Approach to Design, Implementation and Management", Third Edition, Pearson Education, 2007.
3. Henry F Korth, Abraham Silberschatz, S. Sudharshan, "Database System Concepts", Fifth Edition, McGraw Hill, 2006.
4. C.J.Date, A.Kannan and S.Swamynathan, "An Introduction to Database Systems", Eighth Edition, Pearson Education, 2006.
5. Raghu Ramakrishnan, Johannes Gehrke, "Database Management Systems", McGraw Hill, Third Edition 2004.

113ITPP01 - ADVANCED DATA STRUCTURES LABORATORY

OBJECTIVES:

- To learn to implement iterative and recursive algorithms.
- To learn to design and implement algorithms using hill climbing and dynamic programming techniques.
- To learn to implement shared and concurrent objects.
- To learn to implement concurrent data structures.

LAB EXERCISES: Each student has to work individually on assigned lab exercises. Lab sessions could be scheduled as one contiguous four-hour session per week or two two-hour sessions per week. There will be about 15 exercises in a semester. It is recommended that all implementations are carried out in Java. If C or C++ has to be used, then the threads library will be required for concurrency. Exercises should be designed to cover the following topics:

- Implementation of graph search algorithms.
- Implementation and application of network flow and linear programming problems.
- Implementation of algorithms using the hill climbing and dynamic programming design techniques.
- Implementation of recursive backtracking algorithms.
- Implementation of randomized algorithms.
- Implementation of various locking and synchronization mechanisms for concurrent linked lists, concurrent queues, and concurrent stacks.
- Developing applications involving concurrency.

OUTCOMES:

Upon completion of the course, the students will be able to

1. Design and apply iterative and recursive algorithms.
2. Design and implement algorithms using the hill climbing and dynamic programming and recursive backtracking techniques.
3. Design and implement optimisation algorithms for specific applications.
4. Design and implement randomized algorithms.
5. Design appropriate shared objects and concurrent objects for applications.
6. Implement and apply concurrent linked lists, stacks, and queues.

REFERENCES:

1. Jeff Edmonds, "How to Think about Algorithms", Cambridge University Press, 2008.
2. M. Herlihy and N. Shavit, "The Art of Multiprocessor Programming", Morgan Kaufmann, 2008.
3. Steven S. Skiena, "The Algorithm Design Manual", Springer, 2008.
4. Peter Brass, "Advanced Data Structures", Cambridge University Press, 2008.
5. S. Dasgupta, C. H. Papadimitriou, and U. V. Vazirani, "Algorithms", McGrawHill, 2008.
6. J. Kleinberg and E. Tardos, "Algorithm Design", Pearson Education, 2006.
7. T. H. Cormen, C. E. Leiserson, R. L. Rivest and C. Stein, "Introduction to Algorithms", PHI Learning Private Limited, 2012.
8. Rajeev Motwani and Prabhakar Raghavan, "Randomized Algorithms", Cambridge University Press, 1995.
9. A. V. Aho, J. E. Hopcroft, and J. D. Ullman, "The Design and Analysis of Computer Algorithms", Addison-Wesley, 1975.
10. A. V. Aho, J. E. Hopcroft, and J. D. Ullman, "Data Structures and Algorithms", Pearson, 2006.

113ITPP02 - INTERNETWORKING LABORATORY

OBJECTIVES:

- To learn network programming
- To establish connection among the system in the network.
- To analyze network traffic
- To acquire knowledge about various networking tools.

EXERCISES:

1. You are an employee of an XYZ company, working in its branch office and have been given the task to provide service for a valued enterprise customer. If you are unable to solve queries of the customer, obtain the help from your boss at head quarters. The nodes are installed at three different locations. One is at the customer's headquarters, one is at their branch office and one is at a top secret R&D department. Implement the above scenario using TCP socket.

2. Develop a Client/server application in C/C++ for implementing the following scenario: Consider nodes A, B, C and D. Visualize Node A as the source, node B as the router. Nodes C and D are two nodes acting as two different networks connected with two interfaces of the router B. Node A should send an IP datagram to B. B extracts the network address from the destination IP of the datagram and forwards it to either C or D depending on the network address.

Note: Construct IP header with proper fields. Use a static routing table in node B and suitable subnet masks for forwarding to C and D. Show the output for both

Cases

Case 1: IP datagram to node C.

Case 2: IP datagram to node D.)

3. Simulation of DNS using UDP sockets. Consider a root server, three Top Level Domain (TLD) servers (say, corresponding to com, edu and gov), three Authoritative Servers. The client sends a DNS query to the root server. The root server sends the IP address of TLD server to the client. The client sends the DNS query to the TLD server. The TLD server sends the IP address of the authoritative server to the client. The client sends the DNS query to the authoritative server and gets the IP address of the required host. Implement this scenarios in C/C++ language

Note: A domain name consists of one or more parts, technically called *labels*. Each label may contain up to 63 characters. The full domain name may not exceed a total length of 253 characters. The characters allowed in a label follow the LDH rule (letters, digits, hyphen). Domain names are interpreted in case- independent manner. Labels should not start or end with a hyphen

4. Write a C/C++ code in TCP protocol stack, which sends a SYN packet with a randomly spoofed IP to avoid the firewall blocking. This will result in all the entries in the spoofed IP list, sending RST segments to the victim server, upon getting the SYN-ACK from the victim. This can choke the target server and often form a crucial part of a Denial Of Service (**DOS**) attack. Launch the DOS attack by many hosts from various location, all target the same victim to launch Distributed DOS (**DDOS**).

5. Develop a protocol for transmitting private documents via the Internet in an encrypted form. Ensure that the information is sent, unchanged, only to the server you intended to send it to. It has to encrypt TCP/IP traffic that also incorporates authentication and data integrity. It may run on top of TCP/IP. It is based on session-key encryption. It adds a number of extra features, including authentication based on X.509 certificates and integrity checking with message authentication codes. It is an extension of sockets, which allow a client and a server to establish a stream of communication with each other in a secured manner. They begin with a handshake, which allows identities to be established and keys to be exchanged. Show that the developed protocol protects a network from attacks such as IP spoofing, IP source routing, and DNS spoofing.

6. Simulation of HTTP protocol – The client sends a HTTP request (the request is sent in the actual format with request line, header lines). The server gets the HTTP request and understands what method is to be done and responds in a HTTP response message. The response message can have a status line and header lines followed by the data.

- Client Request line format : method/ request http version
- Server reply format : http version status code followed by requested page contents

7. Simulation of FTP – understanding control and data channels. The client initiates a TCP connection to the server and sends the required FTP command to the server. For example, "get filename". The server reads the command, initiates a new TCP connection with the client using a different port number. The server then sends the result of the command to the client and closes the connection.

8. Consider software that resides in a system that reads the available main memory and hard disk at periodical intervals and pass this information on to another system on demand. This process is placed in more than one system so that network resource can be monitored. Implement a monitoring program and list out the time, IP address, available Hard disk and Main memory available in each system in a tabular form.

9. Use the software like wireshark in a LAN to capture the packet and do a statistical analysis such as: the number of packets (bits) flowing in/out of a designated system, a pair wise packet flow among the given IP addresses.

10. Use the packet capturing tool and measure the traffic from each node in a application wise, and pair wise traffic application

11. Repeat exercise 10 for protocol wise traffic analysis

OUTCOMES:

At the end of the course, the students will be able to

- Design network applications using TCP and UDP
- Demonstrate the usage of various networking tools.
- Analyze network traffic
- Analyze packets transmitted over the network.

113ITPP03 - MINI PROJECT

The objective of the mini project is to enable the students to apply the techniques they have learnt over the courses:

- The students can select any of the problem of their choice related to direct application
- A team can be formed with 2-3 members
- Should inculcate the software engineering methodologies
- At least two hours per week should be spent in doing this mini project
- The students should design and experiment using object oriented concept
- A database has to be designed and implemented
- The developed project must be web enabled
- The developed mini project must be tested for different cases of operations
- A documentation has to be prepared in the form of a report
- Evaluation pattern is like Lab examination

213ITPT01 - WEB TECHNOLOGIES

OBJECTIVES:

- To understand the basics of web technology and design methods.
- To learn to design interactive web pages using Markup languages, CSS, Java Script.
- Understand the web platform for creating information sharing and functionality using PHP
- Learn to develop Enterprise Applications for Online communities in the Business World.
- To understand the representation of Web Data using XML technologies
- To understand the techniques of Server side programming

UNIT I WEB TECHNOLOGY AND DESIGN METHODS

Web Technology – The concept of Tiers – Web Pages Types and Issues – Static, Dynamic, Active web Pages. Problems with Statelessness – Sessions and Sessions Management – Techniques for maintaining State Information.

Web Design Methods – Design Issues – OOWS Model_Driven approach – OOADM – UML based Web Engineering.

UNIT II WEB APPLICATION DEVELOPMENT

Client Side Scripting: HTML5 – Elements – canvas – Video on the Web - Cascading Style Sheet 3- Client side scripting – JavaScript – Arrays - JavaScript Objects.

Server Side Scripting: PHP – Introduction – Creating PHP Pages – PHP with MySQL – Tables to display data – Form elements.

UNIT III REPRESENTING WEB DATA

XML Basics – XML Namespaces - XML Schema - DOM – SAX – XPath - XSL: Extensible Style sheet Language – Extensible Style sheet Language Transformations (XSLT).

UNIT IV ENTERPRISE APPLICATION DEVELOPMENT

Working With Model-View-Controller – Introduction to J2EE – JMS - Server Side Component Architecture – EJB overview –Types – Session Beans – Message driven Bean - Entity Beans – Persistent Entity Beans –Building business logic with session bean – messaging with message driven bean - Transactions and Security – Bean managed transaction –Exposing EJB as a web services – JAX-WS.

UNIT V SERVER SIDE APPLICATIONS

Overview of servlets – Servlet API – Servlet life cycle – Servlet configuration – Java Database Connectivity (JDBC) - Running Servlet with database connectivity - Servlet support for cookies – Session tracking- JSP – JSP with JDBC and applications

OUTCOMES:

Upon completion of the course, the students will be able to

- Create interactive web pages using Markup languages, CSS, Java Script.
- Implement server-Side Programming for creating information sharing and functionality using PHP.
- Work on XML Technologies
- Develop Enterprise Applications for online communities in the business world.

REFERENCES:

1. Gustavo Rossi, Oscar Pastor, Daniel Schwabe, Luis Olsina, "Web Engineering Modeling and Implementing web Applications", Springer, 2008. (Unit -1)
2. Achyut S. Godbole, Atul Kahate, "Web Technologies – TCP/IP to Internet Application Architecture", TMGH,2003.(Unit – 1).
3. Deitel & Deitel Internet & World Wide Web How to Program, Pearson Education India - Third Edition -2004 (Unit - 2).
4. Mark Pilgrim, "HTML5: Up and Running", O'Reilly,2010. (Unit - 2).
5. Michael K. Glass, Yann Le Scouarnec, Elizabeth Naramore, Gary Mailer,J.Stolz, J.Gerner, "Beginning PHP, Apache, MySQL, WebDevelopment", Wrox, Wiley dreamtech India Ltd., 2004. (Unit - 2).
6. Deitel & Deitel, Neito, Lin, Sadhu, "XML How to Program", Pearson Education,2008(Unit – 3).
7. Debu Panda, Reza Rahman, Derek Lane, "EJB 1.3 in Action", Dreamtech Press, 2007. (Unit-4)

213ITPT02 - CLOUD COMPUTING

OBJECTIVES:

- To introduce the broad perspective of cloud architecture and model
- To understand the concept of Virtualization
- To be familiar with the lead players in cloud.
- To understand the features of cloud simulator
- To apply different cloud programming model as per need.
- To be able to set up a private cloud.
- To understand the design of cloud Services.
- To learn to design the trusted cloud Computing system

UNIT I CLOUD ARCHITECTURE AND MODEL

Technologies for Network-Based System – System Models for Distributed and Cloud Computing – NIST Cloud Computing Reference Architecture.

Cloud Models:- Characteristics – Cloud Services – Cloud models (IaaS, PaaS, SaaS) – Public vs Private Cloud – Cloud Solutions - Cloud ecosystem – Service management – Computing on demand.

UNIT II VIRTUALIZATION

Basics of Virtualization - Types of Virtualization - Implementation Levels of Virtualization - Virtualization Structures - Tools and Mechanisms - Virtualization of CPU, Memory, I/O Devices - Virtual Clusters and Resource management – Virtualization for Data-center Automation.

UNIT III CLOUD INFRASTRUCTURE

Architectural Design of Compute and Storage Clouds – Layered Cloud Architecture Development – Design Challenges - Inter Cloud Resource Management – Resource Provisioning and Platform Deployment – Global Exchange of Cloud Resources.

UNIT IV PROGRAMMING MODEL

Parallel and Distributed Programming Paradigms – MapReduce , Twister and Iterative Map Reduce – Hadoop Library from Apache – Mapping Applications - Programming Support - Google App Engine, Amazon AWS - Cloud Software Environments -Eucalyptus, Open Nebula, OpenStack, Aneka, CloudSim

UNIT V SECURITY IN THE CLOUD

Security Overview – Cloud Security Challenges and Risks – Software-as-a-Service Security – Security Governance – Risk Management – Security Monitoring – Security Architecture Design – Data Security – Application Security – Virtual Machine Security - Identity Management and Access Control – Autonomic Security.

OUTCOMES:

- Compare the strengths and limitations of cloud computing
- Identify the architecture, infrastructure and delivery models of cloud computing
- Apply suitable virtualization concept.
- Choose the appropriate cloud player
- Choose the appropriate Programming Models and approach.
- Address the core issues of cloud computing such as security, privacy and interoperability
- Design Cloud Services
- Set a private cloud

REFERENCES:

1. Kai Hwang, Geoffrey C Fox, Jack G Dongarra, "Distributed and Cloud Computing, From Parallel Processing to the Internet of Things", Morgan Kaufmann Publishers, 2012.
2. John W.Rittinghouse and James F.Ransome, "Cloud Computing: Implementation, Management, and Security", CRC Press, 2010.
3. Toby Velte, Anthony Velte, Robert Elsenpeter, "Cloud Computing, A Practical Approach", TMH, 2009.
4. Kumar Saurabh, "Cloud Computing – insights into New-Era Infrastructure", Wiley India,2011.
5. George Reese, "Cloud Application Architectures: Building Applications and Infrastructure in the Cloud" O'Reilly
6. James E. Smith, Ravi Nair, "Virtual Machines: Versatile Platforms for Systems and Processes", Elsevier/Morgan Kaufmann, 2005.
7. Katarina Stanoevska-Slabeva, Thomas Wozniak, Santi Ristol, "Grid and Cloud Computing – A Business Perspective on Technology and Applications", Springer.
8. Ronald L. Krutz, Russell Dean Vines, "Cloud Security – A comprehensive Guide to Secure Cloud Computing", Wiley – India, 2010.
9. Rajkumar Buyya, Christian Vecchiola, S.Thamarai Selvi, 'Mastering Cloud Computing', TMGH,2013.
10. Gautam Shroff,Enterprise Cloud Computing,Cambridge University Press,2011
11. Michael Miller, Cloud Computing,Que Publishing,2008
12. Nick Antonopoulos, Cloud computing,Springer Publications,2010

213ITPT03 - NETWORK AND INFORMATION SECURITY

OBJECTIVES:

- To understand the fundamentals of Cryptography
- To acquire knowledge on standard algorithms used to provide confidentiality, integrity and authenticity.
- To understand the various key distribution and management schemes.
- To understand how to deploy encryption techniques to secure data in transit across data networks
- To design security applications in the field of Information technology.

UNIT I INTRODUCTION

An Overview of Computer Security-Security Services-Security Mechanisms-Security Attacks-Access Control Matrix, Policy-Security policies, Confidentiality policies, Integrity policies and Hybrid policies.

UNIT II CRYPTOSYSTEMS & AUTHENTICATION

Classical Cryptography-Substitution Ciphers-permutation Ciphers-Block Ciphers-DESModes of Operation- AES-Linear Cryptanalysis, Differential Cryptanalysis- Hash Function - SHA 512- Message Authentication Codes-HMAC - Authentication Protocols

UNIT III PUBLIC KEY CRYPTOSYSTEMS

Introduction to Public key Cryptography- Number theory- The RSA Cryptosystem and Factoring Integer- Attacks on RSA-The ElGamal Cryptosystem- Digital Signature Algorithm-Finite Fields-Elliptic Curves Cryptography- Key management – Session and Interchange keys, Key exchange and generation-PKI

UNIT IV SYSTEM IMPLEMENTATION

Design Principles, Representing Identity, Access Control Mechanisms, Information Flow and Confinement Problem.

Secure Software Development: Secured Coding - OWASP/SANS Top Vulnerabilities - Buffer Overflows - Incomplete mediation - XSS - Anti Cross Site Scripting Libraries - Canonical Data Format - Command Injection - Redirection - Inference – Application Controls

UNIT V NETWORK SECURITY

Secret Sharing Schemes-Kerberos- Pretty Good Privacy (PGP)-Secure Socket Layer (SSL)- Intruders – HIDS- NIDS - Firewalls – Viruses

OUTCOMES:

Upon completion of the course, the students will be able to

- Implement basic security algorithms required by any computing system.
- Analyze the vulnerabilities in any computing system and hence be able to design a security solution.
- Analyze the possible security attacks in complex real time systems and their effective countermeasures
- Identify the security issues in the network and resolve it.
- Evaluate security mechanisms using rigorous approaches, including theoretical derivation, modeling, and simulations
- Formulate research problems in the computer security field.

REFERENCES:

1. William Stallings, "Cryptography and Network Security: Principles and Practices", Third Edition, Pearson Education, 2006.
2. Matt Bishop, "Computer Security art and science ", Second Edition, Pearson Education, 2002
3. Wade Trappe and Lawrence C. Washington, "Introduction to Cryptography with Coding Theory" Second Edition, Pearson Education, 2007
4. Jonathan Katz, and Yehuda Lindell, Introduction to Modern Cryptography, CRC Press, 2007Douglas R. Stinson, "Cryptography Theory and Practice", Third Edition, Chapman & Hall/CRC, 2006
5. Douglas R. Stinson, "Cryptography Theory and Practice", Third Edition, Chapman & Hall/CRC, 2006
6. Wenbo Mao, "Modern Cryptography – Theory and Practice", Pearson Education, First Edition, 2006.
7. Network Security and Cryptography, Menezes 22Bernard, Cengage Learning, New Delhi, 2011

8. Man Young Rhee, Internet Security, Wiley, 2003

9. OWASP top ten security vulnerabilities: <http://xml.coverpages.org/OWASPTopTen.pdf>

213ITPT04 - DATA WAREHOUSING AND DATA MINING

OBJECTIVES :

- To expose the students to the concepts of Data warehousing Architecture and Implementation
- To Understand Data mining principles and techniques and Introduce DM as a cutting edge business intelligence
- To learn to use association rule mining for handling large data
- To understand the concept of classification for the retrieval purposes
- To know the clustering techniques in details for better organization and retrieval of data
- To identify Business applications and Trends of Data mining

UNIT I DATA WAREHOUSE

Data Warehousing - Operational Database Systems vs. Data Warehouses - Multidimensional Data Model - Schemas for Multidimensional Databases – OLAP Operations – Data Warehouse Architecture – Indexing – OLAP queries & Tools.

UNIT II DATA MINING & DATA PREPROCESSING

Introduction to KDD process – Knowledge Discovery from Databases - Need for Data Preprocessing – Data Cleaning – Data Integration and Transformation – Data Reduction – Data Discretization and Concept Hierarchy Generation.

UNIT III ASSOCIATION RULE MINING

Introduction - Data Mining Functionalities - Association Rule Mining - Mining Frequent Itemsets with and without Candidate Generation - Mining Various Kinds of Association Rules - Constraint-Based Association Mining.

UNIT IV CLASSIFICATION & PREDICTION

Classification vs. Prediction – Data preparation for Classification and Prediction – Classification by Decision Tree Introduction – Bayesian Classification – Rule Based Classification – Classification by Back Propagation – Support Vector Machines – Associative Classification – Lazy Learners – Other Classification Methods – Prediction – Accuracy and Error Measures – Evaluating the Accuracy of a Classifier or Predictor – Ensemble Methods – Model Section.

UNIT V CLUSTERING

Cluster Analysis: - Types of Data in Cluster Analysis – A Categorization of Major Clustering Methods – Partitioning Methods – Hierarchical methods – Density-Based Methods – Grid-Based Methods – Model-Based Clustering Methods – Clustering High- Dimensional Data – Constraint-Based Cluster Analysis – Outlier Analysis.

OUTCOMES:

Upon Completion of the course, the students will be able to

- Store voluminous data for online processing
- Preprocess the data for mining applications
- Apply the association rules for mining the data
- Design and deploy appropriate classification techniques
- Cluster the high dimensional data for better organization of the data
- Discover the knowledge imbibed in the high dimensional system
- Evolve Multidimensional Intelligent model from typical system
- Evaluate various mining techniques on complex data objects

REFERENCES:

1. Jiawei Han and Micheline Kamber, "Data Mining Concepts and Techniques" Second Edition, Elsevier, Reprinted 2008.
2. K.P. Soman, Shyam Diwakar and V. Ajay, "Insight into Data mining Theory and Practice", Easter Economy Edition, Prentice Hall of India, 2006.
3. G. K. Gupta, "Introduction to Data Mining with Case Studies", Easter Economy Edition, Prentice Hall of India, 2006.
4. BERSON, ALEX & SMITH, STEPHEN J, Data Warehousing, Data Mining, and OLAP, TMH Pub. Co. Ltd, New Delhi, 2012
5. Pang-Ning Tan, Michael Steinbach and Vipin Kumar, "Introduction to Data Mining", Pearson Education, 2007
6. PRABHU Data Warehousing, PHI Learning Private Limited, New Delhi, 2012, ,
7. PONNIAH, PAULRAJ, Data Warehousing Fundamentals, John Wiley & Sons, New Delhi, 2011
8. MARAKAS, GEORGE M, Modern Data Warehousing, Mining, and Visualization, Pearson Education, 2011

213ITPP01 -WEB TECHNOLOGY LABORATORY

COURSE OBJECTIVES

- To learn how to create a simple web page using html along with the usage of style sheets, lists, creation or tables with borders, padding and colors.
- To get acquainted with JavaScript and how to embed JavaScript in HTML code.
- To learn to construct dynamic server-side web pages and integrate the web application with many of the other Java2 Enterprise Edition application server methodologies
- To learn to develop Java Enterprise Applications using EJB3 and other Java EE Technology

EXPERIMENT

1. Web programming with HTML tags, CSS for styling, Page layout
 - a. Create a tourist web site using HTML and embed an in a world map in it. Using image map fix hot spots for each continent and show all the related information when the hot spots are clicked.
 - b. Design a HTML form for a railway ticket reservation system with various form controls.
 - c. Create a web site for Online Shopping application. The home page must be divided into three frames. The top frame must be 15% and should display the logo and the moto of the Shopping application along with some pictures. The remaining 85% portion must be divided into two frames. The left frame must be 25% and is meant for appropriate navigation links. The right frame must take the remaining width and must be the target for all the hosted links in the left frame.
 - d. Apply CSS property to a table with alternate rows displayed in different colors.
 - e. Design the style rule which can be applied for hyperlinks using pseudo class properties.
 - f. Write a style rule that will place a nice frame around an image element. Choose suitable color, design and size for the border.
 - g. Design the style rule which can be applied for an online shopping application. The shopping application displays the image for each item and the details of the item. When the mouse over event occurs on the image of an item, the magnified image must be displayed dynamically.
2. Develop webpage using HTML forms and JavaScript for client side programming
 - a. Design a registration page which accepts name, address, pin code, phone no., and e-mail Id. Write suitable JavaScript functions to validate the form whether
 - all fields are entered
 - e-mail Id is a valid one (check for @ and . symbol)
 - phone no. is a valid one (only 10 digits)
 - pin code is having only 6 digits.
 - b. Using a Reg Exp instance, write a JavaScript function is Valid() that accepts a String argument and returns true if the argument matches one of the following phonenummer formats and returns false otherwise:
 - (123)456-7890
 - (123) 456-7890
 - 123/456-7890
 - 123-456-7890
 - 123 456 7890
 - 123456789
 - c. Date has to be printed in several common formats. Write a JavaScript that accepts a date from an XHTML form and convert it to a Date object. Use various methods of the Date object that convert the given date into string and display it in several formats.

- d. Write a JavaScript to simulate the rolling of two dice. The script should use `Math.random()` to roll the first die and again to roll the second die. The sum of the two values should then be calculated. [Note: Since each die can show an integer value from 1 to 6, the sum of the values will vary from 2 to 12, with 7 being the most frequent sum, and 2 and 12 the least frequent sums. Your program should roll the dice 36,000 times. Use a one-dimensional array to tally the number of times each possible sum appears. Display the results in an XHTML table. Also determine whether the totals are reasonable (e.g., there are six ways to roll a 7, so approximately 1/6 of all the rolls should be 7).
3. Using The DOM and the JavaScript object models.
 - a. Create a decorative cursor trail which follows the cursor movement in the window.
 - b. Design a dynamic dropdown menu in a web page
 - c. Write a program to detect which browser is used by the user to host a web page.
 4. Website optimization: crunching HTML, using CSS to replace HTML and light-weight graphics to speed up websites.
 - a. Use DHTML to obtain different transition with different filters.
 - b. Use sequencer control to create a slideshow of images.
 - c. Use path control to have the logo of a webpage follow an Oval path around the page.
 5. Creating XML document with XML DTD and XML schema, SAX, XSL
 - a. Create an XML file for storing the book details in a library and write DTD for the same to validate the XML document.
 - b. Create an XML file for storing the employee details in an organization and write XML Schema for the same.
 - c. Write a program using SAX to count the number of employees in the organization from the above XML data.
 - d. Write a program using SAX to list out the employees who are in the Manager designation.
 - e. Write an XSL to display the XML document for book details and Employees in a HTML table.
 6. Web site creation with PHP for server side programming for storing current date-time using cookies and for storing page views using sessions.
 7. Web application development using PHP
 - a. Write a test application to check an e-mail address. Verify that the input begins with series of characters, followed by @ character, another series of characters, a period (.) and a final series of characters.
 - b. Write a program that logs the address information from the web server, using environment variables.
 8. Working with PHP and MySQL for a ticket reservation system: Develop a client form to get the passenger queries such as timings, availability, fare and get it from the backend MySQL server
 9. Construct dynamic server-side web pages using JSF and integrate the Web application with Enterprise Java Beans, and SOAP.
 10. Developing Java Enterprise Applications Using EJB3 Session beans, entity beans and message-driven beans for a simple inventory application.
 11. Develop routines with JMS to receive message from different sources and store in the database using JDBC connectivity

COURSE OUTCOMES:

- Develop Web application using HTML and scripting technologies.
- Work on Web application development using advanced features.
- Design and development of dynamic server-side web pages.
- Develop web services using J2EE and related technologies
- Design and development applications using JMS and JDBC

213ITPP02 - CLOUD COMPUTING LABORATORY

COURSE OBJECTIVES

- To learn how to use Cloud Services.
- To implement Virtualization
- To implement Task Scheduling algorithms.
- Apply Map-Reduce concept to applications.
- To build Private Cloud.

EXPERIMENT

12. Create a Collaborative learning environment for a particular learning topic using Google Apps. Google Drive, Google Docs and Google Slides must be used for hosting e-books, important articles and presentations respectively. The instructor must use the Google Sheets to convey the timetable for different events and for analyzing the scores for individual assignment submission.

13. Modeling and simulation Cloud computing environments, including Data Centers, Hosts and Cloudlets and perform VM provisioning using CloudSim: Design a host with two CPU cores, which receives request for hosting two VMs, such that each one requires two cores and plans to host four tasks units. More specifically, tasks t1, t2, t3 and t4 to be hosted in VM1, while t5, t6, t7, and t8 to be hosted in VM2. Implement space-shared allocation policy and time-shared allocation policy. Compare the results.

14. Model a Cloud computing environment having Data center that had 100 hosts. The hosts are to be modeled to have a CPU core (1000 MIPS), 2 GB of RAM and 1 TB of storage. Consider the workload model for this evaluation included provisioning requests for 400 VMs, with each request demanding 1 CPU core (250 MIPS), 256 MB of RAM and 1 GB of storage. Each VM hosts a *web-hosting application service*, whose CPU utilization distribution was generated according to the uniform distribution. Each instance of a webhosting service required 150,000 MIPS or about 10 minutes to complete execution assuming 100% utilization.

Simulate Energy-conscious model for power consumption and power management techniques such as Dynamic Voltage and Frequency Scaling (DVFS).

Initially, VMs are to be allocated according to requested parameters (4 VMs on each host). The Cloud computing architecture that is to be considered for studying energyconscious resource management techniques/policies included a data center, CloudCoordinator, and Sensor component. The CloudCoordinator and Sensor perform their usual roles. Via the attached Sensors (which are connected with every host), CloudCoordinator must periodically monitor the performance status of active VMs such as load conditions, and processing share. This real time information is to be passed to VMM, which can use it for performing appropriate resizing of VMs and application of DVFS and soft scaling. CloudCoordinator continuously has to adapt allocation of VMs by issuing VM migration commands and changing power states of nodes according to its policy and current utilization of resources.

15. Model and simulate the environment consisting of a data center with 10,000 hosts where each host was modeled to have a single CPU core (1200MIPS), 4GB of RAM memory and 2TB of storage. Consider the provisioning policy for VMs as space-shared, which allows one VM to be active in a host at a given instance of time. Make a request from the end-user (through the DatacenterBroker) for creation and instantiation of 50 VMs that had following constraints: 1024MB of physical memory, 1 CPU core and 1GB of storage. The application granularity was modeled to be composed of 300 task units, with each task unit requiring 1,440,000 million instructions (20 minutes in the simulated hosts) to be executed on a host. Minimal data transfer (300 KB) overhead can be considered for the task units (to and from the data center). After the creation of VMs, task units were submitted in small groups of 50 (one for each VM) at inter-arrival delay of 10 minutes.

16. Implement Map Reduce concept for

- a. Strassen's Matrix Multiplication for a huge matrix.
- b. Computing the average number of citation index a researcher has according to age

among some 1 billion journal articles.

17. Consider a network of entities and relationships between them. It is required to calculate a state of each entity on the basis of properties of the other entities in its neighborhood. This state can represent a distance to other nodes, indication that there is a neighbor with the certain properties, characteristic of neighborhood density and so on. A network is stored as a set of nodes and each node contains a list of adjacent node IDs. Mapper emits messages for each node using ID of the adjacent node as a key. Reducer must recompute state and rewrite node with the new state. Implement this scenario.

18. Setup a Private Cloud Using OpenStack or Eucalyptus. Develop a simple applications and make it available to the intended user

19. Install and configure OpenStack Object Storage - Swift in Ubuntu. Consider a huge storage requirements and store it in the cloud in a transparent manner

20. Install and configure OpenStack Nova-Compute. Enable a connected user to get a virtual machine of a selected performance such as CPU, Memory

21. Install and configure Stack Image services to query for information on available disk images, and use the Image Service's client library for streaming virtual disk images.

COURSE OUTCOMES:

- Demonstrate and experiment simple Cloud Applications
- Apply resource allocation, scheduling algorithms.
- Implement Map-Reduce concept.
- Create virtual machines from available physical resources.
- Setup a private cloud.
- Familiarize with Open Stack.

213ITPP03 - TECHNICAL SEMINAR

The objective of this technical seminar is to enable the students to read technical article, comprehend and to share with others

The students should read a recent technical article from any of the leading reputed journals like:

- IEEE Transactions
 - ACM
 - Springer
 - Elsevier publications
- In the area of
- Web Technology
 - Cloud Computing
 - Security
 - Data mining

and present to the fellow students with a technical report. External assessment should be conducted along with web technology lab.

313ITPT01 - SOFT COMPUTING

OBJECTIVES:

- To learn the key aspects of Soft computing
- To know about the components and building block hypothesis of Genetic algorithm.
- To understand the features of neural network and its applications
- To study the fuzzy logic components
- To gain insight onto Neuro Fuzzy modeling and control.
- To gain knowledge in machine learning through Support vector machines.

UNIT I INTRODUCTION TO SOFT COMPUTING

Evolution of Computing - Soft Computing Constituents – From Conventional AI to Computational Intelligence - Machine Learning Basics

UNIT II GENETIC ALGORITHMS

Introduction, Building block hypothesis, working principle, Basic operators and Terminologies like individual, gene, encoding, fitness function and reproduction, Genetic modeling: Significance of Genetic operators, Inheritance operator, cross over, inversion & deletion, mutation operator, Bitwise operator, GA optimization problems, JSPP (Job Shop Scheduling Problem), TSP (Travelling Salesman Problem), Differences & similarities between GA & other traditional methods, Applications of GA.

UNIT III NEURAL NETWORKS

Machine Learning using Neural Network, Adaptive Networks – Feed Forward Networks – Supervised Learning Neural Networks – Radial Basis Function Networks - Reinforcement Learning – Unsupervised Learning Neural Networks – Adaptive Resonance Architectures – Advances in Neural Networks.

UNIT IV FUZZY LOGIC

Fuzzy Sets – Operations on Fuzzy Sets – Fuzzy Relations – Membership Functions-Fuzzy Rules and Fuzzy Reasoning – Fuzzy Inference Systems – Fuzzy Expert Systems – Fuzzy Decision Making

UNIT V NEURO-FUZZY MODELING

Adaptive Neuro-Fuzzy Inference Systems – Coactive Neuro-Fuzzy Modeling – Classification and Regression Trees – Data Clustering Algorithms – Rule base Structure Identification – Neuro-Fuzzy Control – Case Studies.

OUTCOMES:

- Implement machine learning through neural networks.
- Write Genetic Algorithm to solve the optimization problem
- Develop a Fuzzy expert system.
- Model Neuro Fuzzy system for clustering and classification.

REFERENCES:

1. Jyh-Shing Roger Jang, Chuen-Tsai Sun, Eiji Mizutani, "Neuro-Fuzzy and Soft Computing", Prentice-Hall of India, 2003
2. Kwang H.Lee, "First course on Fuzzy Theory and Applications", Springer-Verlag Berlin Heidelberg, 2005.
3. George J. Klir and Bo Yuan, "Fuzzy Sets and Fuzzy Logic-Theory and Applications", Prentice Hall, 1995.
4. James A. Freeman and David M. Skapura, "Neural Networks Algorithms, Applications, and Programming Techniques", Pearson Edn., 2003.
5. David E. Goldberg, "Genetic Algorithms in Search, Optimization and Machine Learning", Addison Wesley, 2007.
6. Mitsuo Gen and Runwei Cheng, "Genetic Algorithms and Engineering Optimization", Wiley Publishers 2000.
7. Mitchell Melanie, "An Introduction to Genetic Algorithm", Prentice Hall, 1998.
8. S.N.Sivanandam, S.N.Deepa, "Introduction to Genetic Algorithms", Springer, 2007.
9. A.E. Eiben and J.E. Smith "Introduction to Evolutionary Computing" Springer, 2003
10. E. Sanchez, T. Shibata, and L. A. Zadeh, Eds., "Genetic Algorithms and Fuzzy Logic Systems: Soft Computing Perspectives, Advances in Fuzzy Systems - Applications and Theory", Vol. 7, River Edge, World Scientific, 1997.

213ITPT05 - SOFTWARE METRICS AND RELIABILITY

OBJECTIVES:

- To gain basic knowledge about metrics, measurement theory and related terminologies
- To measure the quality level of internal and external attributes of the software product
- To introduce the basics of software reliability and to illustrate how to perform planning, executing and testing for software reliability
- To explore various metrics and models of software reliability
- To compare various models of software reliability based on its application

UNIT I FUNDAMENTALS OF MEASUREMENTS

Measurements in Software Engineering – Scope of Software Metrics – Fundamentals of Measurements Theory – Goal Based Framework – Software Measurement Validation.

UNIT II METRICS AND MODELS

Measurement of Internal Product Attributes – Size and Structure – External Product Attributes - Measurement of Quality– Reliability Model – Exponential Distribution and Reliability Growth Model – Availability Metrics.

UNIT III INTRODUCTION TO SOFTWARE RELIABILITY

Basic Concepts – Failure and Faults – Environment – Availability – Modeling –uses.

UNIT IV SOFTWARE RELIABILITY MODELING

Concepts – General Model Characteristic – Historical Development of models – Model Classification scheme – Markovian models – General concepts – General Poisson Type Models – Binomial Type Models – Poisson Type models – Fault reduction factor for Poisson Type models.

UNIT V COMPARISON OF SOFTWARE RELIABILITY MODELS

Comparison Criteria – Failure Data – Comparison of Predictive Validity of Model Groups – Recommended Models – Comparison of Time Domains – Calendar Time Modeling – Limiting Resource Concept – Resource Usage model – Resource Utilization – Calendar Time Estimation and confidence Intervals.

OUTCOMES:

Upon Completion of the course, the students should be able to:

- Identify and apply various software metrics, which determines the quality level of software
- Identify and evaluate the quality level of internal and external attributes of the software product
- Compare and Pick out the right reliability model for evaluating the software
- Evaluate the reliability of any given software product
- Design new metrics and reliability models for evaluating the quality level of the software based on the requirement

REFERENCES:

1. John D. Musa, "Software Reliability Engineering", Tata McGraw Hill, 2004.
2. Norman E . Fenton, Shari Lawrence Pfleeger, "Software metrics", Second Edition, International Student Edition, 2003.
3. John D. Musa, Anthony Iannino, Kazuhira Okumoto, "Software Reliability – Measurement, Prediction, Application, Series in Software Engineering and Technology", McGraw Hill, 1987.
4. Stephen H. Kan, "Metrics and Models in Software Quality Engineering", Second Edition, Addison-Wesley Professional, 2002
5. N.Fenton and B.Little Wood, Software Reliability and Metrics, Springer, 1991
6. Michael R.Lyu, Handbook of Software Reliability Engineering, McGraw-Hill, 1996
7. Steven R.Rakitin, Software Verification and Validation for Practitioners and Managers, Artech House, Inc. Norwood, MA, USA, 2001

213ITPT06 - NETWORK MANAGEMENT

OBJECTIVES

- The objective of this course is to
- To understand the need for interoperable network management
- To learn to the concepts and architecture behind standards based network management
- To understand the concepts and terminology associated with SNMP and TMN
- To understand network management as a typical distributed application
- To study the current trends in network management technologies

UNIT I FUNDAMENTALS OF COMPUTER NETWORK TECHNOLOGY

Network Topology, LAN, Network node components- Hubs, Bridges, Routers, Gateways, Switches, WAN, ISDN Transmission Technology, Communications protocols and standards. Network Management: Goals, Organization, and Functions, Network and System Management, Network Management System Platform, Current Status and future of Network

UNIT II OSI NETWORK MANAGEMENT

OSI Network management model-Organizational model-Information model, communication model. Abstract Syntax Notation - Encoding structure, Macros Functional model CMIP/CMIS

UNIT III INTERNET MANAGEMENT(SNMP)

SNMP(V1 and V2)-Organizational model-System Overview, The information model, communication model-Functional model, SNMP proxy server, Management information, protocol remote monitoring- , RMON SMI and MIB, RMON1,RMON2 - A Case Study of Internet Traffic Using RMON.

UNIT IV BROADBAND NETWORK MANAGEMENT

Broadband networks and services, ATM Technology-VP,VC, ATM Packet, Integrated service, ATMLAN emulation, Virtual Lan. ATM Network Management-ATM Network reference model, integrated local management Interface. ATM Management Information base, Role of SNMD and ILMI in ATM Management, M1, M2, M3, M4 Interface. ATM Digital Exchange Interface Management- , TMN conceptual Model- TMN Architecture, TMN Management Service Architecture

UNIT V NETWORK MANAGEMENT APPLICATIONS

Configuration management, Fault management, performance management, Event Correlation Techniques security Management, Accounting management, Report Management, Policy Based Management Service Level Management- Network Management Tools, Network Statistics Measurement Systems – Web Based Management, XML Based Network Management - : Future Directions.

OUTCOMES:

Upon completion of this course, the students will be able to

- Analyze the issues and challenges pertaining to management of emerging network technologies such as wired/wireless networks and high-speed internets.
- Apply network management standards to manage practical networks.
- Formulate possible approaches for managing OSI network model.
- Use on SNMP for managing the network
- Use RMON for monitoring the behavior of the network
- Explore the possibilities of improving the speed of the network and managing them
- Identify the various components of network and formulate the scheme for the managing them

REFERENCES:

1. Mani Subramanian, "Network Management Principles and practice ", Pearson Education, New Delhi, 2010.
2. STALLINGS, WILLIAM, "SNMP, SNMPv2, SNMPv3, and RMON 1 and 2," Pearson Education, 2012
3. Salah Aaidarous, Thomas Plevayk, "Telecommunications Network Management Technologies and Implementations ", eastern Economy Edition IEEE press, New Delhi, 1998.

4. Lakshmi G. Raman, "Fundamentals of Telecommunication Network Management ", Eastern Economy Edition IEEE Press, New Delhi, 1999.

213ITPT07 - BIOINFORMATICS

OBJECTIVES:

- To get exposed to the domain of bioinformatics
- To understand the role of data warehousing and data mining for bioinformatics
- To learn to model bioinformatics based applications
- To understand how to deploy the pattern matching and visualization techniques in bioinformatics
- To study the Microarray technologies for genome expression

UNIT I INTRODUCTION

Need for Bioinformatics technologies – Overview of Bioinformatics technologies – Structural bioinformatics – Data format and processing – secondary resources- Applications – Role of Structural bioinformatics - Biological Data Integration System.

UNIT II DATAWAREHOUSING AND DATAMINING IN BIOINFORMATICS

Bioinformatics data – Data ware housing architecture – data quality – Biomedical data analysis – DNA data analysis – Protein data analysis – Machine learning – Neural network architecture- Applications in bioinformatics

UNIT III MODELING FOR BIOINFORMATICS

Hidden markov modeling for biological data analysis – Sequence identification – Sequence classification – multiple alignment generation – Comparative modeling – Protein modeling – genomic modeling – Probabilistic modeling – Bayesian networks – Boolean networks - Molecular modeling – Computer programs for molecular modeling

UNIT IV PATTERN MATCHING AND VISUALIZATION

Gene regulation – motif recognition and motif detection – strategies for motif detection – Visualization – Fractal analysis – DNA walk models – one dimension – two dimension – higher dimension – Game representation of Biological sequences – DNA, Protein, Amino acid sequences

UNIT V MICROARRAY ANALYSIS

Microarray technology for genome expression study – image analysis for data extraction – preprocessing – segmentation – gridding , spot extraction , normalization, filtering – cluster analysis – gene network analysis – Compared Evaluation of Scientific Data Management Systems – Cost Matrix – Evaluation model ,Benchmark , Tradeoffs

OUTCOMES:

Upon Completion of the course, the students will be able to

- Deploy the data warehousing and data mining techniques in Bioinformatics
- Model bioinformatics based applications
- Deploy the pattern matching and visualization techniques in bioinformatics
- Work on the protein sequences
- Use the Microarray technologies for genome expression

REFERENCES:

1. Yi-Ping Phoebe Chen (Ed), "Bio Informatics Technologies", First Indian Reprint, Springer Verlag, 2007.
2. N.J. Chikhale and Virendra Gomase, "Bioinformatics- Theory and Practice", Himalaya Publication House, India, 2007
3. Zoe Iacox and Terence Critchlow, "Bio Informatics – Managing Scientific data", First Indian Reprint, Elsevier, 2004
4. Bryan Bergeron, "Bio Informatics Computing", Second Edition, Pearson Education, 2003.
5. Arthur M Lesk, "Introduction to Bioinformatics", Second Edition, Oxford University Press, 2005
6. Burton. E. Tropp, "Molecular Biology: Genes to Proteins ", 4th edition, Jones and Bartlett Publishers, 2011

7. Dan Gusfield, "Algorithms on Strings Trees and Sequences", Cambridge University Press, 1997.
8. P. Baldi, S Brunak , Bioinformatics, "A Machine Learning Approach ", MIT Press, 1998.

213ITPT08 - XML AND WEBSERVICES

OBJECTIVES:

To provide an in-depth knowledge of XML and Web Services.

- To understand the fundamental concepts of Web services.
- To Understand the fundamental concepts of XML Technology.
- To design Web service Architecture.
- To Study Building Blocks of Web services.
- To understand the XML security issues.

UNIT I WEB FUNDAMENTALS

History of Web – Protocols – Web Applications - Web servers-Web Browsers-HTTP-Java Network Programming-HTML-CCS.

UNIT II XML TECHNOLOGY

XML-XML DTD-W3C XML Schema-Parsing XML - X path- XML Transformation-Other XML Technologies.

UNIT III ARCHITECTING WEB SERVICES

Business motivations for web services – B2B – B2C- Technical motivations -- Service oriented Architecture (SOA) – Architecting web services – Implementation view – web services technology stack – logical view – composition of web services – deployment view – from application server to peer to peer – process view – life in the runtime

UNIT IV WEB SERVICES BUILDING BLOCK

Transport protocols for web services – messaging with web services – protocols – SOAP – describing web services – WSDL – Anatomy of WSDL – manipulating WSDL – web service policy – Discovering web services – UDDI – Anatomy of UDDI

UNIT V XML SECURITY

Security Overview - Canonicalization - XML Security Framework - XML Encryption - XML Digital Signature - XKMS Structure - Guidelines for Signing XML Documents - XML in Practice.

OUTCOMES:

Upon Completion of the course, the students will be able

- To Know the fundamental elements in Web Technology and XML services.
- To design the Architecture of Web Services.
- To construct building blocks of Web services.
- To analyze security in XML.

REFERENCE BOOKS:

1. Uttam K.Roy , "Web Technologies", Oxford University Press,2010
2. Ron schmelzer et al, "XML and Web Services", Pearson Education, 2002.
3. Sandeep Chatterjee and James Webber, "Developing Enterprise Web Services: An Architect's Guide", Prentice Hall, 2004.
4. Frank. P. Coyle, XML, Web Services And The Data Revolution, Pearson Education, 2002
5. Keith Ballinger, ".NET Web Services Architecture and Implementation", Pearson Education,2003
6. Henry Bequet and Meeraj Kunnumpurath, "Beginning Java Web Services", Apress, 2004.
7. Russ Basiura and Mike Batongbacal, "Professional ASP.NET Web Services", Apress2,2001.

213ITPT09 - ENTERPRISE APPLICATION INTEGRATION

OBJECTIVES:

- Describe approaches to enterprise application integration
- Understand the integration middleware
- Evaluate the integration approaches suitable for a given problem

UNIT I INTRODUCTION

Requirements for EAI - Challenges in EAI – Integration with legacy systems – Integration with partners - Heterogeneous environment – Implementation approaches – Web services, messaging, ETL, direct data integration – Middleware requirements – Approaches to integration – services oriented and messaging.

UNIT II INTEGRATION PATTERNS

Introduction to integration patterns – Architecture for application integration – Integration patterns – Point to point, broker, message bus, publish/subscribe, Challenges in performance, security, reliability - Case studies

UNIT III SERVICE ORIENTED INTEGRATION

Business process integration - Composite applications-services – Web services – Service choreography and orchestration - Business process modeling - BPMN, Business process execution - BPEL – Middleware infrastructure - Case studies

UNIT IV MESSAGING BASED INTEGRATION

Messaging – Synchronous and asynchronous – Message structure – Message oriented middleware – Reliability mechanisms – Challenges – Messaging infrastructure – Java Messaging Services – Case studies

UNIT V ENTERPRISE SERVICE BUS

Enterprise Service Bus – routing, scalable connectivity, protocol and message transformations, data enrichment, distribution, correlation, monitoring – Deployment configurations – Global ESB, Directly connected, Federated, brokered ESBs – Application server based – Messaging system based – Hardware based ESBs – Support to SOA, message based and event based integrations - Case studies.

OUTCOMES:

Upon Completion of the course, the students will be able to

- Describe different approaches to integration enterprise applications
- Analyze specifications and identify appropriate integration approaches
- Develop a suitable integration design for a given problem
- Identify appropriate integration middleware for a given problem
- Evaluate the integration approaches against specified requirements

REFERENCES:

1. George Mentzas and Andreas Frezen (Eds), "Semantic Enterprise Application Integration for Business Processes: Service-oriented Frameworks", Business Science Reference, 2009
2. Waseem Roshen, "SOA Based Enterprise Integration", Tata McGrawHill, 2009.
3. GHohpe and B Woolf, "Enterprise Integration Patterns: Designing, Building, and Deploying Messaging Solutions", Addison Wesley Professional, 2003
4. D Linthicum, "Next Generation Application Integration: From Simple Information to Web Services", Addison Wesley, 2003
5. Martin Fowler, "Patterns of Enterprise Application Architecture", Addison- Wesley, 2003
6. Kapil Pant and Matiaz Juric, "Business Process Driven SOA using BPMN and BPEL: From Business Process Modeling to Orchestration and Service Oriented Architecture", Packt Publishing, 2008

213ITPT10 - VIDEO ANALYTICS

OBJECTIVES:

- To know the fundamental concepts of big data and analytics
- To learn various techniques for mining data streams
- To acquire the knowledge of extracting information from surveillance videos.
- To learn Event Modelling for different applications.
- To understand the models used for recognition of objects in videos.

UNIT I INTRODUCTION TO BIG DATA & DATA ANALYSIS

Introduction to Big Data Platform – Challenges of Conventional systems – Web data- Evolution of Analytic scalability- analytic processes and tools- Analysis Vs Reporting- Modern data analytic tools- Data Analysis: Regression Modeling- Bayesian Modeling- Rule induction.

UNIT II MINING DATA STREAMS

Introduction to Stream concepts- Stream data model and architecture – Stream Computing- Sampling data in a Stream- Filtering Streams- Counting distinct elements in a Stream- Estimating moments- Counting oneness in a window- Decaying window- Real time Analytics platform(RTAP) applications- case studies.

UNIT III VIDEO ANALYTICS

Introduction- Video Basics - Fundamentals for Video Surveillance- Scene Artifacts- **Object Detection and Tracking:** Adaptive Background Modelling and Subtraction- Pedestrian Detection and Tracking-Vehicle Detection and Tracking- Articulated Human Motion Tracking in Low-Dimensional Latent Spaces

UNIT IV BEHAVIOURAL ANALYSIS & ACTIVITY RECOGNITION

Event Modelling- Behavioural Analysis- Human Activity Recognition-Complex Activity Recognition- Activity modelling using 3D shape, Video summarization, shape based activity models- Suspicious Activity Detection

UNIT V HUMAN FACE RECOGNITION & GAIT ANALYSIS

Introduction: Overview of Recognition algorithms – Human Recognition using Face: Face Recognition from still images, Face Recognition from video, Evaluation of Face Recognition Technologies- Human Recognition using gait: HMM Framework for Gait Recognition, View Invariant Gait Recognition, Role of Shape and Dynamics in Gait Recognition

OUTCOMES:

On successful completion of this course, students will be able to:

1. Work with big data platform and its analysis techniques.
2. Design efficient algorithms for mining the data from large volumes.
3. Work with surveillance videos for analytics.
4. Design of optimization algorithms for better analysis and recognition of objects in a scene.
5. Model a framework for Human Activity Recognition

REFERENCES:

1. Michael Berthold, David J.Hand, Intelligent Data Analysis, Springer, 2007.
2. Anand Rajaraman and Jeffrey David Ullman, Mining of Massive Datasets, Cambridge University Press, 2012.
3. Yunqian Ma, Gang Qian, "Intelligent Video Surveillance: Systems and Technology", CRC Press (Taylor and Francis Group), 2009.
4. Rama Chellappa, Amit K.Roy-Chowdhury, Kevin Zhou.S, "Recognition of Humans and their Activities using Video", Morgan&Claypool Publishers, 2005.

213ITPT11 - SOFTWARE PROJECT MANAGEMENT

OBJECTIVES:

- To define and highlight importance of software project management.
- To formulate strategy in managing projects
- To estimate the cost associated with a project
- To plan, schedule and monitor projects for the risk management
- To define the software management metrics
- To train software project managers and other individuals involved in software project planning and tracking and oversight in the implementation of the software project management process

UNIT I PROJECT MANAGEMENT CONCEPTS

Evolution of Software Economics – Software Management Process Framework (Phases, Artifacts, Workflows, Checkpoints) – Software Management Disciplines (Planning / Project Organization and Responsibilities / Automation / Project Control) – Modern Project Profiles

UNIT II SOFTWARE ESTIMATION & COSTING

Problems in Software Estimation – Algorithmic Cost Estimation Process, Function Points, SLIM (Software Life cycle Management), COCOMO II (Constructive Cost Model) – Estimating Web Application Development – Concepts of Finance, Activity Based Costing and Economic Value Added (EVA) – Balanced Score Card.

UNIT III RISK MANAGEMENT

Risk Definition – Risk Categories – Risk Assessment (Identification / Analysis / Prioritization) – Risk Control (Planning / Resolution / Monitoring) – Failure Mode and Effects Analysis (FMEA)

UNIT IV METRICS

Need for Software Metrics – Classification of Software Metrics: Product Metrics (Size Metrics, Complexity Metrics, Halstead's Product Metrics, Quality Metrics), and Process metrics (Empirical Models, Statistical Models, Theory-based Models, Composite Models and Reliability Models).

UNIT V MANAGING PEOPLE AND ORGANIZING TEAMS

Introduction – Understanding Behavior – Organizational Behavior: A Background – Selecting the right person for the job – Instruction in the best methods – Motivation – The Oldman – Hackman Job Characteristics Model – Working in Groups – Becoming a team – Decision Making – Leadership – Organizational Structures – Stress – Health and Safety – Case Studies.

OUTCOMES:

At the end of this course students will be able to:

- Evaluate a project to develop the scope of work, provide accurate cost estimates and to plan the various activities
- Apply risk management analysis techniques that identify the factors that put a project at risk and to quantify the likely effect of risk on project timescales
- Identify the resources required for a project and to produce a work plan and resource schedule
- Monitor the progress of a project and to assess the risk of slippage, revising targets counteract drift
- Use appropriate metrics to management the software development outcome
- Engage and motivate the stakeholders of the project
- Develop research methods and techniques appropriate to defining, planning and carrying out a research project within your chosen specialist area within the management of software projects.

REFERENCES:

1. McConnell, S. "Software Project: Survival Guide", Microsoft Press, 1998.
2. Bob Hughes, Mikecatterell, "Software Project Management", Third Edition, Tata McGraw Hill, 2004.
3. Ramesh, Gopaldaswamy, "Managing Global Projects", Tata McGraw Hill, 2001.
4. Royce, W. "Software Project management: A Unified Framework", Addison-Wesley, 1998.
5. Cooper, R., "The Rise of Activity-Based Costing- Part One: What is an Activity-Based Cost System?" Journal of Cost Management, Vol.2, No.2 (Summer 1988), pp.45 -54
6. Grant, J.L. "Foundations of Economic Value Added", John Wiley & Sons, 1997. Kaplan, R.S., Norton, D.P. "The Balanced Scorecard: Translating Strategy into Action", Harvard Business School Press, 1996.
7. Boehm, B. W. "Software Risk Management: Principles and Practices" in IEEE Software, January 1991, pp32-41.
8. Fenton, N.E., and Pfleeger, S.L. "Software Metrics: A Rigorous and Practical Approach, Revised" Brooks Cole, 1998.
9. Demarco, T. and Lister, T. "Peopleware: Productive Projects and Teams, 2nd Ed.", Dorset House, 1999.
10. Jalote, "Software Project Management in Practice", Pearson Education, 2002.

213ITPT12 - MOBILE AND PERVASIVE COMPUTING

OBJECTIVES :

- To understand the basics of Mobile Computing and Personal Computing
- To learn the role of cellular networks in Mobile and Pervasive Computing
- To expose to the concept of sensor and mesh networks
- To expose to the context aware and wearable computing
- To learn to develop applications in mobile and pervasive computing environment

UNIT I INTRODUCTION

Differences between Mobile Communication and Mobile Computing – Contexts and Names – Functions – Applications and Services – New Applications – Making Legacy Applications Mobile Enabled – Design Considerations – Integration of Wireless and Wired Networks – Standards Bodies – Pervasive Computing – Basics and Vision – Principles of Pervasive Computing – Categories of Pervasive Devices

UNIT II 3G AND 4G CELLULAR NETWORKS

Migration to 3G Networks – IMT 2000 and UMTS – UMTS Architecture – User Equipment – Radio Network Subsystem – UTRAN – Node B – RNC functions – USIM – Protocol Stack – CS and PS Domains – IMS Architecture – Handover – 3.5G and 3.9G a brief discussion – 4G LAN and Cellular Networks – LTE – Control Plane – NAS and RRC – User Plane – PDCP, RLC and MAC – WiMax IEEE 802.16d/e – WiMax Internetworking with 3GPP

UNIT III SENSOR AND MESH NETWORKS

Sensor Networks – Role in Pervasive Computing – In Network Processing and Data Dissemination – Sensor Databases – Data Management in Wireless Mobile Environments – Wireless Mesh Networks – Architecture – Mesh Routers – Mesh Clients – Routing – Cross Layer Approach – Security Aspects of Various Layers in WMN – Applications of Sensor and Mesh networks

UNIT IV CONTEXT AWARE COMPUTING & WEARABLE COMPUTING

Adaptability – Mechanisms for Adaptation - Functionality and Data – Transcoding – Location Aware Computing – Location Representation – Localization Techniques – Triangulation and Scene Analysis – Delaunay Triangulation and Voronoi graphs – Types of Context – Role of Mobile Middleware – Adaptation and Agents – Service Discovery Middleware Health BAN- Medical and Technological Requirements-Wearable Sensors-Intra-BAN communications

UNIT V APPLICATION DEVELOPMENT

Three tier architecture - Model View Controller Architecture - Memory Management – Information Access Devices – PDAs and Smart Phones – Smart Cards and Embedded Controls – J2ME – Programming for CLDC – GUI in MIDP – Application Development ON Android and iPhone

OUTCOMES:

At the end of the course the student should be able to

- Design a basic architecture for a pervasive computing environment
- Design and allocate the resources on the 3G-4G wireless networks
- Analyze the role of sensors in Wireless networks
- Work out the routing in mesh network
- Deploy the location and context information for application development
- Develop mobile computing applications based on the paradigm of context aware computing and wearable computing

REFERENCES:

1. Asoke K Talukder, Hasan Ahmed, Roopa R Yavagal, "Mobile Computing: Technology, Applications and Service Creation", 2nd ed, Tata McGraw Hill, 2010.
2. Reto Meier, "Professional Android 2 Application Development", Wrox Wiley, 2010.
3. .Pei Zheng and Lionel M Li, 'Smart Phone & Next Generation Mobile Computing', Morgan Kaufmann Publishers, 2006.
4. Frank Adelstein, 'Fundamentals of Mobile and Pervasive Computing', TMH, 2005
5. Jochen Burthardt et al, 'Pervasive Computing: Technology and Architecture of Mobile Internet Applications', Pearson Education, 2003
6. Feng Zhao and Leonidas Guibas, 'Wireless Sensor Networks', Morgan Kaufmann Publishers, 2004
7. Uwe Hansmaan et al, 'Principles of Mobile Computing', Springer, 2003
8. Reto Meier, "Professional Android 2 Application Development", Wrox Wiley, 2010.
9. Mohammad s. Obaidat et al, "Pervasive Computing and Networking", John Wiley
10. Stefan Poslad, "Ubiquitous Computing: Smart Devices, Environments and Interactions", Wiley, 2009
11. Frank Adelstein Sandeep K. S. Gupta Golden G. Richard III Loren Schwiebert "Fundamentals of Mobile and Pervasive Computing, ", McGraw-Hill, 2005

213ITPT13 - PRINCIPLES OF PROGRAMMING LANGUAGES

COURSE OBJECTIVES:

1. To understand and describe syntax and semantics of programming languages
2. To understand data, data types, and basic statements
3. To understand call-return architecture and ways of implementing them
4. To understand object-orientation, concurrency, and event handling in programming languages
5. To develop programs in non-procedural programming paradigms

UNIT I SYNTAX AND SEMANTICS

Evolution of programming languages – describing syntax – context-free grammars – attribute grammars – describing semantics – lexical analysis – parsing – recursive-decent – bottomup parsing

UNIT II DATA, DATA TYPES, AND BASIC STATEMENTS

Names – variables – binding – type checking – scope – scope rules – lifetime and garbage collection – primitive data types – strings – array types – associative arrays – record types – union types – pointers and references – Arithmetic expressions – overloaded operators – type conversions – relational and boolean expressions – assignment statements – mixedmode assignments – control structures – selection – iterations – branching – guarded statements

UNIT III SUBPROGRAMS AND IMPLEMENTATIONS

Subprograms – design issues – local referencing – parameter passing – overloaded methods – generic methods – design issues for functions – semantics of call and return – implementing simple subprograms – stack and dynamic local variables – nested subprograms – blocks – dynamic scoping

UNIT IV OBJECT-ORIENTATION, CONCURRENCY, AND EVENT HANDLING

Object-orientation – design issues for OOP languages – implementation of object-oriented constructs – concurrency – semaphores – monitors – message passing – threads – statement level concurrency – exception handling – even handling

UNIT V FUNCTIONAL AND LOGIC PROGRAMMING LANGUAGES

Introduction to lambda calculus – fundamentals of functional programming languages – Programming with Scheme – Programming with ML – Introduction to logic and logic programming – Programming with Prolog – multi-paradigm languages

COURSE OUTCOMES:

Upon Completion of the course, the students will be able to

1. Describe syntax and semantics of programming languages
2. Explain data, data types, and basic statements of programming languages
3. Design and implement subprogram constructs
4. Apply object-oriented, concurrency, and event handling programming constructs
5. Develop programs in Scheme, ML, and Prolog
6. Understand and adopt new programming languages

REFERENCES:

1. Robert W. Sebesta, "Concepts of Programming Languages", Tenth Edition, Addison Wesley, 2012.
2. Michael L. Scott, "Programming Language Pragmatics", Third Edition, Morgan Kaufmann, 2009.
3. R. Kent Dybvig, "The Scheme programming language", Fourth Edition, MIT Press, 2009.
4. Jeffrey D. Ullman, "Elements of ML programming", Second Edition, Prentice Hall, 1998.
5. Richard A. O'Keefe, "The craft of Prolog", MIT Press, 2009.
6. W. F. Clocksin and C. S. Mellish, "Programming in Prolog: Using the ISO Standard", Fifth Edition, Springer, 2003.

213ITPT14 - MULTIMEDIA TECHNOLOGIES

OBJECTIVES:

- To familiarize with various elements of multimedia
- To understand the functions of the various of elements in multimedia
- To understand the various multimedia systems
- To learn to use various tools for developing multimedia
- To learn to develop a multimedia application

UNIT I INTRODUCTION

Introduction to Multimedia – Characteristics – Utilities – Creation -Uses – Promotion – Digital Representation – Media and Data streams – Multimedia Architecture – Multimedia Documents

UNIT II ELEMENTS OF MULTIMEDIA

Multimedia Building Blocks: Text, Graphics, Video Capturing, Sound Capturing, and Editing- Intro to 2D & 3D Graphics -surface characteristics and texture - lights – Animation :key frames & Tweening, techniques, principles of animation, 3Danimation, file formats.

UNIT III MULTIMEDIA SYSTEMS

Visual Display Systems – CRT - video adapter card - video adapter cable – LCD – PDP - optical storage media - CD technology - DVD Technology - Compression Types and Techniques – CODEC - GIF coding standards – lossy and lossless – JPEG - MPEG-1 - MPEG-2 - MP3 - Fractals – MMDBS

UNIT IV MULTIMEDIA TOOLS

Authoring tools – features and types - card and page based tools - icon and object based tools - time based tools - cross platform authoring tools – Editing tools - text editing and word processing tools - OCR software - painting and drawing tools - 3D modeling and animation tools - image editing tools –sound editing tools - digital movie tools – plug -ins and delivery vehicles for www

UNIT V MULTIMEDIA APPLICATION DEVELOPMENT

Software life cycle – ADDIE Model – conceptualization – content collection and processing – story – flowline – script - storyboard - implementation - multiplatform issues – authoring – metaphors – testing – report writing - documentation - case study: -Web Application – Console Application – Distributed Application – Mobile Application - games consoles – iTV – kiosks – education

OUTCOMES: Upon Completion of the course, the students should be able to:

- Design a multimedia architecture for handling the stream
- Work with the various elements of multimedia system
- Develop multimedia stream for the various standards
- Develop animation, images, Sound using Multimedia Tools.
- Develop a multimedia application

REFERENCES:

1. Parekh R "Principles Of Multimedia" Tata McGraw-Hill, 2006.
2. Ralf Steinmetz, Klara Nahrstedt, "Multimedia: Computing, Communications and Applications" Prentice Hall, 1995.
3. John Villamil and Louis Molina, "Multimedia ; An Introduction", Prentice Hall, New Delhi 1998.
4. Tay Vaughan, "Multimedia: Making It Work" McGraw-Hill Professional, 2006
5. Deitel & Deitel "Internet & World Wide Web How to Program", Fourth Edition – Prentice Hall, 2008
6. BANERJI ASHOK & GHOSH ANANDA MOHAN, Multimedia Technologies, TMH, New Delhi, 2010

213ITPT15 - AUTOMATA THEORY AND COMPILER DESIGN

OBJECTIVES:

- To understand a finite automata for a given language.
- To understand the principle of a Turing machine
- To understand the relation between grammar and language
- To understand the basic principles of working of a compiler
- To study about the type checking procedure during the compilation
- To understand the storage structure of the running program
- To understand the various scope of optimization techniques used in compiler design

UNIT I AUTOMATA

Introduction to formal proof – Additional forms of proof – Inductive proofs – Finite Automata (FA) – Deterministic Finite Automata (DFA) – Non-deterministic Finite Automata (NFA) – Finite Automata with Epsilon transitions - Equivalence and minimization of Automata.

UNIT II CONTEXT FREE GRAMMARS AND LANGUAGES

Context-Free Grammar (CFG) – Parse Trees – Ambiguity in grammars and languages – Definition of the Pushdown automata – Languages of a Pushdown Automata – Equivalence of Pushdown automata and CFG– Deterministic Pushdown Automata- Normal forms for CFG – Pumping Lemma for CFL – Closure Properties of CFL – Turing Machines – Programming Techniques for TM.

UNIT III BASICS OF COMPILATION

Compilers – Analysis of source program – Phases of a compiler – Grouping of phases – Compiler construction tools – Lexical Analyzer : Token Specification – Token Recognition – A language for Specifying lexical analyzer – Top down parser : Table implementation of Predictive Parser – Bottom up Parser : SLR(1) Parser – Parser generators.

UNIT IV TYPE CHECKING AND RUNTIME ENVIRONMENTS

Syntax directed definitions – Construction of syntax trees – Type systems – Specification of a simple type checker- Equivalence of type expressions – Type conversions – Attribute grammar for a simple type checking system – Runtime Environments : Source language issues – Storage organization – Storage allocation strategies – Parameter passing.

UNIT V CODE GENERATION AND OPTIMIZATION

Issues in the design of a code generator - The target machine - Run-time storage management - Basic blocks and flow graphs - Next-use information - A simple code generator - Register allocation and assignment - The dag representation of basic blocks - Generating code from DAG – Dynamic programming code generation algorithm – Code generator generators - The principle sources of optimization - Peephole optimization - Optimization of basic blocks - Loops in flow graphs - Introduction to global data - flow analysis - Iterative solutions of data flow equations - Code improving transformations

OUTCOMES:

Upon completion of the course, the students should be able to :

- Design a finite automaton for a specific language.
- Design a Turing machine.
- Select appropriate grammar for the implementation of compiler phases
- Design a lexical analyzer
- Design a simple parser
- Design and implement techniques used for optimization by a compiler.
- Write a very simple code generator

REFERENCES:

1. J.E. Hopcroft, R. Motwani and J.D. Ullman, "Introduction to Automata Theory, Languages and Computations", Second Edition, Pearson Education, 2007.
2. H.R. Lewis and C.H. Papadimitriou, "Elements of the theory of Computation", Second Edition, Pearson Education, 2003.
3. Thomas A. Sudkamp, "An Introduction to the Theory of Computer Science, Languages and Machines", Third Edition, Pearson Education, 2007.
4. Raymond Greenlaw and H. James Hoover, "Fundamentals of Theory of Computation, Principles and Practice", Morgan Kaufmann Publishers, 1998.
5. Michael Sipser, "Introduction of the Theory and Computation", Thomson Brooks/Cole, 1997.
6. J. Martin, "Introduction to Languages and the Theory of computation" Third Edition, Tata Mc Graw Hill, 2007
7. Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman, "Compilers :Principles, Techniques and Tools", Second Edition, Pearson Education, 2008.
8. Randy Allen, Ken Kennedy, "Optimizing Compilers for Modern Architectures: A Dependence-based Approach", Morgan Kaufmann Publishers, 2002.
9. Steven S. Muchnick, "Advanced Compiler Design and Implementation", Morgan Kaufmann Publishers - Elsevier Science, India, Indian Reprint 2003.

213ITPT16 - BIG DATA ANALYTICS

COURSE OBJECTIVES

- To explore the fundamental concepts of big data analytics
- To learn to analyze the big data using intelligent techniques.
- To understand the various search methods and visualization techniques.
- To learn to use various techniques for mining data stream.
- To understand the applications using Map Reduce Concepts.

UNIT I INTRODUCTION TO BIG DATA

Introduction to BigData Platform – Challenges of Conventional Systems - Intelligent data analysis – Nature of Data - Analytic Processes and Tools - Analysis vs Reporting - Modern Data Analytic Tools - Statistical Concepts: Sampling Distributions - Re-Sampling - Statistical Inference - Prediction Error.

UNIT II DATA ANALYSIS

Regression Modeling - Multivariate Analysis – Bayesian Methods – Bayesian Paradigm - Bayesian Modeling - Inference and Bayesian Networks - Support Vector and Kernel Methods - Analysis of Time Series: Linear Systems Analysis - Nonlinear Dynamics - Rule Induction - Fuzzy Logic: Extracting Fuzzy Models from Data - Fuzzy Decision Trees

UNIT III SEARCH METHODS AND VISUALIZATION

Search by simulated Annealing – Stochastic, Adaptive search by Evaluation – Evaluation Strategies – Genetic Algorithm – Genetic Programming – Visualization – Classification of Visual Data Analysis Techniques – Data Types – Visualization Techniques – Interaction techniques – Specific Visual data analysis Techniques.

UNIT IV MINING DATA STREAMS

Introduction To Streams Concepts – Stream Data Model and Architecture - Stream Computing - Sampling Data in a Stream – Filtering Streams – Counting Distinct Elements in a Stream – Estimating Moments – Counting Oneness in a Window – Decaying Window - Real time Analytics Platform(RTAP) Applications - Case Studies - Real Time Sentiment Analysis, Stock Market Predictions.

UNIT V FRAMEWORKS

MapReduce – Hadoop, Hive, MapR – Sharding – NoSQL Databases - S3 - Hadoop Distributed File Systems – Case Study.

COURSE OUTCOMES:

At the end of this course the students will be able to:

- Work with big data platform and its analysis techniques.
- Analyze the big data for useful business applications.
- Select visualization techniques and tools to analyze big data
- Implement search methods and visualization techniques
- Design efficient algorithms for mining the data from large volumes.
- Explore the technologies associated with big data analytics such as NoSQL, Hadoop and Map Reduce.

REFERENCES:

1. Michael Berthold, David J. Hand, "Intelligent Data Analysis", Springer, 2007.
2. Anand Rajaraman and Jeffrey David Ullman, "Mining of Massive Datasets", Cambridge University Press, 2012.
3. Bill Franks, "Taming the Big Data Tidal Wave: Finding Opportunities in Huge Data Streams with Advanced Analytics", John Wiley & sons, 2012.
4. Glenn J. Myatt, "Making Sense of Data", John Wiley & Sons, 2007
5. Pete Warden, "Big Data Glossary", O'Reilly, 2011.
6. Jiawei Han, Micheline Kamber "Data Mining Concepts and Techniques", Second Edition, Elsevier, Reprinted 2008.
7. Da Ruan, Guoqing Chen, Etienne E.Kerre, Geert Wets, Intelligent Data Mining, Springer,2007
8. Paul Zikopoulos ,Dirk deRoos , Krishnan Parasuraman , Thomas Deutsch , James Giles, David Corrigan , Harness the Power of Big Data The IBM Big Data Platform, Tata McGraw Hill Publications, 2012
9. Michael Minelli (Author), Michele Chambers (Author), Ambiga Dhiraj (Author) , Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today's Businesses,Wiley Publications,2013
10. Zikopoulos, Paul, Chris Eaton, Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data, Tata McGraw Hill Publications, 2011

313ITPT02 - SOFTWARE QUALITY AND TESTING

OBJECTIVES:

- To explore the basics and goals of software testing.
- To discuss various types of software testing and its techniques
- To list out various tools which can be used for automating the testing process
- To introduce various software quality standards for establishing quality environment
- To discuss various methods and evaluation procedures for improving the quality models

UNIT I INTRODUCTION

Basics of Software Testing – Testing Principles – Goals – Testing Life Cycle– Phases of Testing–Test Plan(IEEE format) – Importance of Testing in Software Production Cycle.

UNIT II SOFTWARE TESTING METHODOLOGY

Software Test Plan – Components of Plan - Types of Technical Reviews - Static and Dynamic Testing- – Software Testing in Spiral Manner - Information Gathering - Test Planning - Test Case Design - Test Development - Test Coverage - Test Evaluation - Prepare for Next Spiral - Conduct System Test - Acceptance Test - Summarize Testing Results.

UNIT III EMERGING SPECIALIZED AREAS IN TESTING

Test Process Assessment – Test Automation Assessment - Test Automation Framework – Nonfunctional Testing – SOA Testing – Agile Testing – Testing Center of Excellence – Onsite/Offshore Model - Taxonomy of Testing tools, Methodology to evaluate automated testing tools, Rational Testing Tools, Java Testing Tools, JMetra, JUNIT and Cactus.

UNIT IV SOFTWARE QUALITY MODELS

Software quality –Verification versus Validation– Components of Quality Assurance – SQA Plan – Quality Standards – CMM – PCMM – CMMI – Malcolm Baldrige National Quality Award.

UNIT V QUALITY THROUGH CONTINUOUS IMPROVEMENT PROCESS

Role of Statistical Methods in Software Quality – Transforming Requirements into Test Cases – Deming’s Quality Principles – Continuous Improvement through Plan Do Check Act (PDCA).

OUTCOMES:

Upon Completion of the course, the students should be able to

- Compare and pick out the right type of software testing process for any given real world problem
- Carry out the software testing process in efficient way
- Automate the testing process by using several testing tools
- Establish a quality environment as specified in standards for developing quality software
- Analyze and improve the quality procedures based on the past experience

REFERENCES:

1. William E.Lewis, "Software Testing and Continuous Quality Improvement", Third edition, Auerbach Publications, 2011.
2. Kshirasagar Naik, Priyadarshi Tripathy, "Software Testing and Quality Assurance - Theory and Practice", John Wiley & Sons publication, 2011.
3. Ron Patton, "Software testing", Second edition, Pearson Education, 2007
4. Elfriede Dustin, Jeff Rashka, John Paul, "Automated Software Testing: Introduction, Management and Performance", Addison-Wesley, 1999.
5. Effective Methods for Software Testing, 2nd Edition, William E. Perry , Second Edition, Wiley India, 2006.
6. Software Testing Tools, K.V.K.K. Prasad, Dream tech press, 2008
7. Testing and Quality Assurance for Component-based Software, by Gao, Tsao and Wu, Artech House Publishers

8. Software Testing, Srinivasan Desikan & Gopalaswamy Ramesh, Pearson Education, 2006.
9. Software Testing Techniques, Scott Loveland & Geoffrey Miller, Shroff Publishers, 2005.
10. Software Testing Techniques, by Bories Beizer, Second Edition, Dreamtech Press
Managing the Testing Process, by Rex Black, Wiley

313ITPT03 - WIRELESS ADHOC AND SENSOR NETWORKS

UNIT I ADHOC NETWORKS AND ROUTING PROTOCOLS

Ad hoc Wireless Networks – What is an Ad Hoc Network? Heterogeneity in Mobile Devices – Wireless Sensor Networks – Traffic Profiles – Types of Ad hoc Mobile Communications – Types of Mobile Host Movements – Challenges Facing Ad hoc Mobile Networks – Ad hoc wireless Internet .

Issues in Designing a Routing Protocol for Ad Hoc Wireless Networks – Classifications of Routing Protocols – Table-Driven Routing Protocols – Destination Sequenced Distance Vector (DSDV) – Wireless Routing Protocol (WRP) – Cluster Switch Gateway Routing (CSGR) – Source-Initiated On-Demand Approaches – Ad hoc On-Demand Distance Vector Routing (AODV) – Dynamic Source Routing (DSR) – Temporally Ordered Routing Algorithm (TORA) – Signal Stability Routing (SSR) – Location-Aided Routing (LAR) – Power-Aware Routing (PAR) – Zone Routing Protocol (ZRP).

UNIT II MULTICAST ROUTING AND SECURITY

Issues in Designing a Multicast Routing Protocol – Operation of Multicast Routing Protocols – An Architecture Reference Model for Multicast Routing Protocols – Classifications of Multicast Routing Protocols – Tree-Based Multicast Routing Protocols– Mesh-Based Multicast Routing Protocols – Summary of Tree and Mesh based Protocols – Energy-Efficient Multicasting – Multicasting with Quality of Service Guarantees – Application – Dependent Multicast Routing –

Comparisons of Multicast Routing Protocols - Design Goals of a Transport Layer Protocol for Ad hoc Wireless Networks – Classification of Transport Layer Solutions – TCP over Ad hoc Wireless Networks- Security in Ad Hoc Wireless Networks – Network Security Requirements – Issues and Challenges in Security Provisioning – Network Security Attacks – Key Management – Secure Routing in Ad hoc Wireless Networks.

UNIT III QoS AND ENERGY MANAGEMENT

Issues and Challenges in Providing QoS in Ad hoc Wireless Networks – Classifications of QoS Solutions – MAC Layer Solutions – Network Layer Solutions – QoS Frameworks for Ad hoc Wireless Networks Energy Management in Ad hoc Wireless Networks – Introduction – Need for Energy Management in Ad hoc Wireless Networks – Classification of Energy Management Schemes – Battery Management Schemes – Transmission Power Management Schemes – System Power Management Schemes.

UNIT IV SENSOR NETWORKS – ARCHITECTURE AND MAC PROTOCOLS

Single node architecture – Hardware components, energy consumption of sensor nodes, Network architecture – Sensor network scenarios, types of sources and sinks, single hop versus multi-hop networks, multiple sinks and sources, design principles, Development of wireless sensor networks. , physical layer and transceiver design consideration in wireless sensor networks, Energy usage profile, choice of modulation, Power Management - MAC protocols – fundamentals of wireless MAC protocols, low duty cycle protocols and wakeup concepts, contention-based protocols, Schedule-based protocols - SMAC, BMAC, Traffic adaptive medium access protocol (TRAMA), Link Layer protocols – fundamentals task and requirements, error control, framing, link management.

UNIT V SENSOR NETWORKS – ROUTING PROTOCOLS AND OPERATING SYSTEMS

Gossiping and agent-based uni-cast forwarding, Energy-efficient unicast, Broadcast and multicast, geographic routing, mobile nodes, Data-centric routing – SPIN, Directed Diffusion, Energy aware routing, Gradient-based routing – COUGAR, ACQUIRE, Hierarchical Routing – LEACH, PEGASIS, Location Based Routing – GAF, GEAR, Data aggregation – Various aggregation techniques. Introduction to TinyOS – NesC, Interfaces, modules, configuration, Programming in TinyOS using NesC, Emulator TOSSIM.

REFERENCES:

1. C. Siva Ram Murthy and B. S. Manoj, "Ad Hoc Wireless Networks Architectures and Protocols", Prentice Hall, PTR, 2004.
2. C. K. Toh, "Ad Hoc Mobile Wireless Networks Protocols and Systems", Prentice Hall, PTR, 2001.
3. Charles E. Perkins, "Ad Hoc Networking", Addison Wesley, 2000.
4. Kazem Sohraby, Daniel Minoli and Taieb Znati, "Wireless Sensor Networks Technology- Protocols and Applications", John Wiley & Sons, 2007.
5. Feng Zhao, Leonidas Guibas, "Wireless Sensor Networks: an information processing approach", Elsevier publication, 2004.
6. C.S.Raghavendra Krishna, M.Sivalingam and Tarib znati, "Wireless Sensor Networks", Springer publication, 2004.
7. Holger Karl , Andreas willig, "Protocol and Architecture for Wireless Sensor Networks", John wiley publication, Jan 2006.
8. K.Akkaya and M.Younis, " A Survey of routing protocols in wireless sensor networks", Elsevier Adhoc Network Journal, Vol.3, no.3,pp. 325-349, 2005.
9. Philip Levis, " TinyOS Programming", 2006 – www.tinyos.net.
10. I.F. Akyildiz, W. Su, Sankarasubramaniam, E. Cayirci, "Wireless sensor networks: a survey", computer networks, Elsevier, 2002, 394 - 422.
11. Jamal N. Al-karaki, Ahmed E. Kamal, "Routing Techniques in Wireless sensor networks: A survey", IEEE wireless communication, December 2004, 6 – 28.

313ITPT04 - WEB MINING

OBJECTIVES:

- To understand the different knowledge discovery issues in data mining from the world wide web.
- To analyze the different algorithms commonly used by Web application.
- To apply the role played by Web mining in Information retrieval and extraction
- To learn the documents structures and grouping
- To learn to use the probabilistic model for web mining
- To learn to develop applications using Web mining

UNIT I INTRODUCTION

Overview of Data mining – Data mining from a Business Perspective – Data types, Input and output of data mining algorithms- Decision Tree- Classification and Regression Trees – Preprocessing and Post processing in Data mining

UNIT II INFORMATION RETRIEVAL

Information Retrieval and Text Mining - Keyword Search - Nearest-Neighbor Methods - Measuring Similarity - Web-Based Document Search - Document-Matching - Inverted Lists - Evaluation of Performance - Structure in a Document Collection - Clustering Documents by Similarity- Evaluation of Performance - Information Extraction - Patterns and Entities from Text- Co reference and Relationship Extraction - Template Filling and Database Construction

UNIT III WEB SEARCH

Crawling the web – HTML and HTTP Basics – Crawling Basics – Engineering Large Scale Crawlers- Putting together a Crawler- Boolean Queries and the Inverted Index – Relevance Ranking – Similarity Search

UNIT IV LEARNING

Similarity and Clustering – Formulations and approaches- Bottom up and Top down Partitioning Paradigms – Clustering and Visualization via Embeddings – Probabilistic Approaches to clustering – Collaborative Filtering – Supervised Learning – Semi Supervised Learning

UNIT V APPLICATIONS

Social Network Analysis- Social Sciences and Bibliometry – Page Rank and HITS – Short comings of coarse Grained Graph model- Enhanced Models and Techniques- Evaluation of Topic Distillation- Measuring and Modeling the Web – Resource Discovery – Collecting Important Pages Preferentially – Similarity Search Using Link Topology – Topical Locality and Focused Crawling – Discovering Communities- The Future of Web Mining.

OUTCOMES:

At the end of the course the student should be able to:

- Identify the application areas for web content mining, web structure mining and web usage mining.
- Design to retrieval the web data
- Develop schemes to crawl the web data, organize and index
- Cluster the documents for fast access
- Develop algorithms used by web mining applications.
- Select between different approaches and techniques of web mining

REFERENCES :

1. Sholom Weiss, "Text Mining: Predictive Methods for Analyzing Unstructured Information", Springer, 2005
2. Hercules Antonio do Prado, Edilson Fernada, " Emerging Technologies of Text Mining: Techniques and Applications", Information Science Reference (IGI), 2008
3. Min Song, Yi-fang Brok Wu, "Handbook of Research on Text and Web Mining Technologies", Vol I & II, Information Science Reference (IGI),2009
4. Soumen Chakrabarti " Mining the Web : Discovery Knowledge from Hypertext Data" Elsevier Science 2003
5. K.P.Soman,Shyam Diwakar, V.Ajay " Insight into Data Mining Theory and Practice " Prentice Hall of India Private Ltd 2006
6. Anthony Scime, "Web Mining Applications and Techniques", Idea Group Publishing, 2005
7. Margret H.Dunham "DATA MINING - Introductory and Advanced Concepts", Pearson Education,2003.
8. R. Kosala and H. Blockeel, "Web Mining Research: A Survey", SIGKDD Exploration, vol. 2, issue 1, 2000.
9. J. Srivastava et al, "Web Usage Mining: Discovery and Applications of Usage Patterns from Web Data", SIGKDD Exploration, vol. 2, issue 1, 1999

313ITPT05 - IMAGE PROCESSING AND PATTERN ANALYSIS

OBJECTIVES:

To introduce the student to various Image processing and Pattern recognition techniques.

- To study the Image fundamentals.
- To study the mathematical morphology necessary for Image processing and Image segmentation.
- To study the Image Representation and description and feature extraction.
- To study the principles of Pattern Recognition.
- To know the various applications of Image processing.

UNIT I INTRODUCTION

Elements of an Image Processing System- Mathematical Preliminaries- Image Enhancement-Grayscale Transformation- Piecewise Linear Transformation-Bit Plane Slicing- Histogram Equalization--Histogram Specification- Enhancement by Arithmetic Operations- Smoothing Filter- Sharpening Filter- Image Blur Types and Quality Measures.

UNIT II MATHEMATICAL MORPHOLOGY and IMAGE SEGMENTATION

Binary Morphology-Opening and Closing- Hit-or-Miss Transform- Grayscale Morphology- Basic morphological Algorithms- Morphological Filters-Thresholding-Object (Component) Labeling-Locating Object Contours by the Snake Model- Edge Operators-Edge Linking by Adaptive Mathematical morphology- Automatic Seeded Region Growing- A Top-Down Region Dividing Approach.

UNIT III IMAGE REPRESENTATION AND DESCRIPTION and FEATURE EXTRACTION

Run-Length Coding- Binary Tree and Quadtree- Contour Representation-Skeletonization by Thinning- Medial Axis Transformation-Object Representation and Tolerance- Fourier Descriptor and Moment Invariants-Shape Number and Hierarchical Features-Corner Detection- Hough Transform-Principal Component Analysis-Linear Discriminate Analysis- Feature Reduction in Input and Feature Spaces.

UNIT IV PATTERN RECOGNITION

The Unsupervised Clustering Algorithm-Bayes Classifier- Support Vector Machine- Neural Networks-The Adaptive Resonance Theory Network-Fuzzy Sets in Image Analysis- Document image processing and classification-Block Segmentation and Classification- Rule-Based Character Recognition system- Logo Identification-Fuzzy Typographical Analysis for Character Pre classification-Fuzzy Model for Character Classification.

UNIT V APPLICATIONS:

Face and Facial Feature Extraction-Extraction of Head and Face Boundaries and Facial Features-Recognizing Facial Action Units-Facial Expression Recognition in JAFFE Database-Image Steganography- Types of Steganography- Applications of Steganography- Embedding Security and Imperceptibility- Examples of Steganography Software-Genetic Algorithm Based Steganography.

OUTCOMES:

Upon Completion of the course, the students will be able to:

- Process the image for better appearance.
- Segment the image using different techniques
- Represent the images in different forms
- Develop algorithms for Pattern Recognition
- Extract and deploy the features in various Image processing applications

REFERENCES:

1. Image Processing and Pattern Recognition: Fundamentals and Techniques- Frank Y Shih, Willey IEEE Press, April 2010
2. Rafael C. Gonzalez, Richard E. Woods, Steven Eddins, " Digital Image Processing using MATLAB", Pearson Education, Inc., 2004.
3. D.E. Dudgeon and R.M. Mersereau, "Multidimensional Digital Signal Processing", Prentice Hall Professional Technical Reference, 52 1990.

4. William K. Pratt, " Digital Image Processing", John Wiley, New York, 2002.
5. Milan Sonka et al, "Image Processing, Analysis and Machine Vision", Brookes/Cole, Vikas Publishing House, 2nd edition, 1999;
6. Sid Ahmed, M.A., " Image Processing Theory, Algorithms and Architectures", McGrawHill, 1995

313ITPT06 - INTELLIGENT AGENTS

OBJECTIVES:

- To understand the basics of User Interface Design.
- To design the user interface, design, menu creation and windows creation
- To understand the concept of menus, windows, interfaces, business functions, various problems in windows design with color, text, Non-anthropomorphic Design.
- To study the design process and evaluations.

UNIT I AGENTS – OVERVIEW

Agent Definition – Agent Programming Paradigms – Agent Vs Object – Aglet – Mobile Agents – Agent Frameworks – Agent Reasoning.

UNIT II JAVA AGENTS

Processes – Threads – Daemons – Components – Java Beans – ActiveX – Sockets – RPCs – Distributed Computing – Aglets Programming – Jini Architecture – Actors and Agents – Typed and proactive messages.

UNIT III MULTIAGENT SYSTEMS

Interaction between agents – Reactive Agents – Cognitive Agents – Interaction protocols – Agent coordination – Agent negotiation – Agent Cooperation – Agent Organization – Self-Interested agents in Electronic Commerce Applications.

UNIT IV INTELLIGENT SOFTWARE AGENTS

Interface Agents – Agent Communication Languages – Agent Knowledge Representation – Agent Adaptability – Belief Desire Intension – Mobile Agent Applications.

UNIT V MOBILE AGENTS AND SECURITY

Mobile Agent Paradigm - Mobile Agent Concepts -Mobile Agent Technology-Agent Security Issues – Mobile Agents Security – Protecting Agents against Malicious Hosts – Untrusted Agent – Black Box Security – Authentication for agents

OUTCOMES:

- Understand the notion of an agent, how agents are distinct from other software paradigms
- Knowledge on the characteristics of applications that lend themselves to an agentoriented solution.
- Know the key issues associated with constructing agents capable of intelligent autonomous actions.
- Learn the main application areas of agent-based solutions, and be able to develop a meaningful agent-based system using a contemporary agent development platform.

REFERENCES:

1. Bigus & Bigus, " Constructing Intelligent agents with Java ", Wiley, 1997.
2. Bradshaw, " Software Agents ", MIT Press, 2000.
3. Russel, Norvig, "Artificial Intelligence: A Modern Approach", Second Edition, Pearson Education, 2003.
5. Richard Murch, Tony Johnson, "Intelligent Software Agents", Prentice Hall, 2000.
6. Gerhard Weiss, "Multi Agent Systems – A Modern Approach to Distributed Artificial Intelligence", MIT Press, 2000.
7. William R. Cockayne, Michael Zyda, "Mobile Agents", Prentice Hall, 1998

313ITPT07 - INTERNET OF THINGS

COURSE OBJECTIVES:

- To learn the basic issues, policy and challenges in the Internet
- To understand the components and the protocols in Internet
- To build a small low cost embedded system with the internet
- To understand the various modes of communications with internet
- To learn to manage the resources in the Internet
- To deploy the resources into business
- To understand the cloud and internet environment.

UNIT I INTRODUCTION

Definition – phases – Foundations – Policy– Challenges and Issues - identification - security – privacy. Components in internet of things: Control Units – Sensors – Communication modules – Power Sources – Communication Technologies – RFID – Bluetooth – Zigbee – Wifi – Rflinks – Mobile Internet – Wired Communication

UNIT II PROGRAMMING THE MICROCONTROLLER FOR IOT

Basics of Sensors and actuators – examples and working principles of sensors and actuators – Cloud computing and IOT – Arduino/Equivalent Microcontroller platform – Setting up the board - Programming for IOT – Reading from Sensors

Communication: Connecting microcontroller with mobile devices – communication through bluetooth and USB – connection with the internet using wifi / Ethernet

UNIT III RESOURCE MANAGEMENT IN THE INTERNET OF THINGS

Clustering - Software Agents - Data Synchronization - Clustering Principles in an Internet of Things Architecture - The Role of Context - Design Guidelines -Software Agents for Object - Data Synchronization- Types of Network Architectures - Fundamental Concepts of Agility and Autonomy-Enabling Autonomy and Agility by the Internet of Things-Technical Requirements for Satisfying the New Demands in Production - The Evolution from the RFIDbased EPC Network to an Agent based Internet of Things- Agents for the Behaviour of Objects

UNIT IV BUSINESS MODELS FOR THE INTERNET OF THINGS

The Meaning of DiY in the Network Society- Sensor-actuator Technologies and Middleware as a Basis for a DiY Service Creation Framework - Device Integration - Middleware Technologies Needed for a DiY Internet of Things Semantic Interoperability as a Requirement for DiY Creation -Ontology- Value Creation in the Internet of Things- Application of Ontology Engineering in the Internet of Things-Semantic Web-Ontology - The Internet of Things in Context of EURIDICE - Business Impact

UNIT V FROM THE INTERNET OF THINGS TO THE WEB OF THINGS:

Resource-oriented Architecture and Best Practices- Designing REST ful Smart Things - Web-enabling Constrained Devices - The Future Web of Things - Set up cloud environment – send data from microcontroller to cloud – Case studies – Open Source e-Health sensor platform – Be Close Elderly monitoring – Other recent projects.

COURSE OUTCOMES:

At the end of this course the students will be able to:

1. Identify the components of IOT
2. Design a portable IOT using appropriate boards
3. Program the sensors and controller as part of IOT
4. Develop schemes for the applications of IOT in real time scenarios
5. Establish the communication to the cloud through wifi/ Bluetooth
6. Manage the internet resources
7. Model the Internet of things to business

REFERENCES:

1. Charalampos Doukas , Building Internet of Things with the Arduino, Create space, April 2002
2. Dieter Uckelmann et.al, "Architecting the Internet of Things", Springer, 2011

3. Luigi Atzor et.al, "The Internet of Things: A survey, ", Journal on Networks, Elsevier Publications, October, 2010
4. <http://postscapes.com/>
5. <http://www.theinternetofthings.eu/what-is-the-internet-of-things>

313ITPT08 - WEB ENGINEERING

OBJECTIVES:

- To understand the issues and process of Web design.
- To learn the concepts of Web design patterns and page design.
- To understand and learn the scripting languages with design of web applications.
- To learn the maintenance and evaluation of web design management.

UNIT I INTRODUCTION TO WEB ENGINEERING

History of web Development, Evolution and Need for Web Engineering, World Wide Web, Introduction to TCP/IP and WAP , DNS, Email, TelNet, HTTP and FTP, Introduction to Browser and search engines, Web Servers, Features of web servers, caching, case study-IIS, Apache, Configuring web servers.

UNIT II INFORMATION ARCHITECTURE

The role of Information Architect, Collaboration & Communication, Organizing Information, Organizational Challenges, Organizing Web sites parameters and Intranets, Creating Cohesive Websites, Architectural Page Mockups, Design Sketches, Navigation Systems, Searching Systems Good & bad web design, Process of Web Publishing, Phases of Web Site development, Requirements Engineering for Web Applications.

UNIT III HTML & DHTML

HTML Basic Concept, Static & dynamic HTML, Structure of HTML documents, HTML Elements, Linking in HTML, Anchor Attributes, Image Maps, Meta Information, Image Preliminaries, Layouts, backgrounds, Colors and Text, Fonts, Tables, Frames and layers, Audio and Video Support with HTML Database integration, CSS, Positioning with Style sheets, Forms Control, Form Elements, Introduction to CGI PERL, JAVA SCRIPT, PHP, ASP, Cookies Creating and Reading Cookies.

UNIT IV XML

Introduction of XML, Validation of XML documents, DTD, Ways to use XML, XML for data files HTML Vs XML, Embedding XML into HTML documents, Converting XML to HTML for Display, Displaying XML using CSS and XSL, Rewriting HTML as XML, Relationship between HTML, SGML and XML, web personalization , Semantic web, Semantic Web Services, Ontology.

UNIT V APPLICATIONS AND SECURITY

E-commerce Business Models, The Internet and World Wide Web, Modes of Electronic Commerce, Approaches to safe Electronic Commerce, Electronic Cash and Electronic Payment Schemes, Online Security and Payment Systems, E-commerce Marketing Concepts, Advertising on the Internet, Electronic Publishing issues, approaches, Legalities & technologies, Privacy & Security, Web Security, Encryption schemes, Secure Web document, Digital Signatures and Firewalls, Cyber crime and laws, IT Act.

OUTCOMES:

Upon completion of the course, the students will be able to

- Identify the various issues of web design process and evaluation.
- Determine templates for web pages and layout.
- Develop simple web applications using scripting languages.
- Determine the various issues of web project development.
- Address the core issues of web page maintenance and evaluation.

REFERENCES:

1. Roger S. Pressman, David Lowe, "Web Engineering", Tata McGraw Hill Publication, 2007
2. Web Engineering: A Practitioner's Approach by Roger Pressman and David Lowe, McGraw-Hill, 2009.

3. Achyut S Godbole and Atul Kahate, "Web Technologies", Tata McGraw Hill
4. NEIL GRAY, "Web server Programming" Wiley
5. CHRIS BATES Web Programming :Building Internet applications, Wiley
6. Moller, "An Introduction to XML and Web Technologies", Pearson Education New Delhi, 2009

313ITPT09 - NETWORK INFRASTRUCTURE AND CYBER COMPUTING

UNIT I INTRODUCTION AND ROUTER ARCHITECTURE

Introduction: Networks, architecture, applications, ISO model. Routers: Function of Router – Types – Elements – Packet flow – Packet Processing – Algorithms and Data Structures (packet buffer allocation) – Packet processing functions (Bridge Algorithm, Table Lookup and Hashing) – Protocol Software (threads, Interrupts) – Hardware Architectures For Protocol Processing – Classification and Forwarding – Switching Fabrics.

UNIT II NETWORK PROCESSORS

Scalability with Parallelism and Pipelining – Complexity of Network Processor Design (Packet processing, ingress & egress processing, Macroscopic Data Pipelining and Heterogeneity) – Network Processor Architectures: architectural variety, Primary Architectural characteristics, Packet Flow, Clock Rates, software architecture, Assigning Functionality to the Processor Hierarchy. Examples of Commercial Network Processors: Multi-Chip Pipeline, Augmented RISC Processor, Embedded Processor Plus Coprocessors.

UNIT III CYBER COMPUTING

Computer Forensics Fundamentals – Types of Computer Forensics Technology – Types of Vendor and Computer Forensics Services - Fighting against Macro Threats – Information Warfare Arsenal – Tactics of the Military – Tactics of Terrorist and Rogues – Tactics of Private Companies.

UNIT IV RECOVERY AND EVIDENCES

Data Recovery – Evidence Collection and Data Seizure – Duplication and Preservation of Digital Evidence – Computer Image Verification and Authentication – Discover of Electronic Evidence – Identification of Data – Reconstructing Past Events.

UNIT V ETHICAL HACKING

Hacking windows – Network hacking – Web hacking – Password hacking. A study on various attacks: Input validation attacks – SQL injection attacks – Buffer overflow attacks - Privacy attacks – Fundamentals of Computer Fraud – Threat concepts – Framework for predicting inside attacks – Managing the threat – Architecture strategies for computer fraud prevention – Protection of Web sites – Intrusion detection system.

Text Books:

1. Douglas E. Comer "Network System Design using Network Processors" Prentice Hall, 2006.
2. Deepankar Medhi, Karthikeyan Ramasamy, "Network Routing: Algorithms, Protocols, and Architecture", Elsevier, 2007.
3. John R. Vacca, "Computer Forensics", Firewall Media, 2004.
4. Ankit Fadia "Ethical Hacking" second edition Macmillan India Ltd, 2006.

Reference Books:

1. Chad Steel, "Windows Forensics", Wiley India, 2006.
2. Majid Yar, "Cybercrime and Society", Sage Publications, 2006.

313ITPT10 - WEB ADVERTISING AND MARKETING

UNIT I

Introduction: Internet Principles – Basic Web Concepts – Client/Server model – Retrieving data from Internet – HTML and Scripting Languages – Standard Generalized Mark-up Language –Next Generation Internet – Protocols and applications. Introduction to How Web Advertising Works, Banner Ads – Sidebar Ads – Varied Shapes and Sizes – PopUp and PopUnder – Floating Ads.

UNIT II

Enterprise Application development environment : Web servers – Server Administration – IDL – Database Connectivity – Web Application architecture – Distributed Web Applications – Remote method Invocation – Web Customization – Mark Up Languages.

UNIT III

E-Business Applications: E-Business Frame Work – E-Business Cycle – E-Commerce Strategies – E-Business Architectures – Stored Procedures – SQL Procedures – Electronic Payment Services - Shopping Functions.

UNIT IV

Business and the Marketing Concept : How to Make a Web Page - Elements of Good Web Site Design - Starting a Business Online - Server Services - Domain Names -Web Oriented Industries. Online Marketing – Email Marketing – Search Engine Marketing – Banner Ad Placement – Link Exchange.

UNIT V

Real Time Applications: Role of scripting languages – Shopping Cart – Home Banking Applications – Design and Implementation – Fire Wall – Business models – Tools usage.

Reference Books:

1. Ed Roman, "Mastering EJB and the Java 2 Platform Enterprise edition", John Wiley and Sons.
2. Stephen Aubury, Scott R.Weiner, "Developing Java Enterprise Applications" Wiley Computer publishing, 2001.
3. "Professional Java Server Programming", Wrox Press Ltd.
4. Daniel J. Berg, J. Steven Fritzinger, "Advanced Java Techniques for Java Developers", John Wiley and Sons, Revised Edition.
5. Rickard Oberg, "Mastering RMI: Developing Enterprise Applications in Java and EJB", John Wiley and Sons, Book and CDROM edition 2001.
6. Harold, Elliotte Rusty Harold, "XML Bible", 2nd Edition Hungry Minds, Inc.

313ITPT11 - PARALLEL PROGRAMMING PARADIGMS

OBJECTIVES:

- To understand models of and issues in concurrency in computing
- To develop message-passing parallel programs using MPI
- To develop shared-memory parallel programs using Pthreads
- To develop shared-memory parallel programs using OpenMP
- To use GPU for parallel programming using OpenCL and CUDA

UNIT I FOUNDATIONS OF PARALLEL PROGRAMMING

Motivation for parallel programming - Concurrency in computing – basics of processes, multiprocessing, and threads – cache – cache mappings – caches and programs – virtual memory – instruction level parallelism – hardware multi-threading – SIMD – MIMD – interconnection networks – cache coherence – shared-memory model – issues in sharedmemory model – distributed-memory model – issues in distributed-memory model – hybrid model – I/O – performance of parallel programs – parallel program design

UNIT II MESSAGE PASSING PARADIGM

Basic MPI programming – MPI_Init and MPI_Finalize – MPI communicators – SPMD programs – message passing – MPI_Send and MPI_Recv – message matching – MPI I/O – parallel I/O – collective communication – MPI_Reduce – MPI_Allreduce – broadcast – scatter – gather – allgather – derived types – remote memory access – dynamic process management – MPI for grids – performance evaluation of MPI programs

UNIT III SHARED MEMORY PARADIGM: PTHREADS

Basics of Pthreads – thread synchronization – critical sections – busy-waiting – mutexes – semaphores – barriers and condition variables – read-write locks – Caches, cache coherence and false sharing – thread safety – Pthreads case study

UNIT IV SHARED MEMORY PARADIGM: OPENMP

Basic OpenMP constructs – scope of variables – reduction clause – parallel for directive – loops in OpenMP – scheduling loops – synchronization in OpenMP – Case Study: Producer-Consumer problem – cache issues – threads safety in OpenMP – OpenMP best Practices

UNIT V GRAPHICAL PROCESSING PARADIGMS: OPENCL AND CUDA

Introduction to CUDA – CUDA programming examples – CUDA execution model – CUDA memory hierarchy – CUDA case study - introduction to OpenCL – OpenCL programming examples – Programs and Kernels – Buffers and Images – Event model – OpenCL case study

OUTCOMES:

Upon completion of the course, the students will be able to

- Explain models of parallel programming
- Explain hardware level support for concurrency
- Explain issues in parallel programming
- Develop message-passing parallel programs using MPI framework
- Develop shared-memory parallel programs using Pthreads
- Develop shared-memory parallel programs using OpenMP
- Develop CUDA programs
- Develop OpenCL programs

REFERENCES:

1. Peter S. Pacheco, "An introduction to parallel programming", Morgan Kaufmann, 2011.
2. M. J. Quinn, "Parallel programming in C with MPI and OpenMP", Tata McGraw Hill, 2003.
3. W. Gropp, E. Lusk, and R. Thakur, "Using MPI-2: Advanced features of the message passing interface", MIT Press, 1999.
4. W. Gropp, E. Lusk, and A. Skjellum, "Using MPI: Portable parallel programming with the message passing interface", Second Edition, MIT Press, 1999.
5. B. Chapman, G. Jost, and Ruud van der Pas, "Using OpenMP", MIT Press, 2008.
6. D. R. Butenhof, "Programming with POSIX Threads", Addison Wesley, 1997.

7. B. Lewis and D. J. Berg, "Multithreaded programming with Pthreads", Sun Microsystems Press, 1998.
8. A. Munshi, B. Gaster, T. G. Mattson, J. Fung, and D. Ginsburg, "OpenCL programming guide", Addison Wesley, 2011.
9. Rob Farber, "CUDA application design and development", Morgan Kaufmann, 2011.

313ITPT12 - SOCIAL NETWORK ANALYSIS

OBJECTIVES:

- To understand the concepts of Social networks and Web Social Networks
- To appreciate the modeling and visualizing techniques associated with Social Networks
- To understand the different techniques used to mine communities from Web Social Networks
- To appreciate concepts of evolution and prediction in Social Networks
- To understand the application of text mining techniques for Content and Opinion mining

UNIT I INTRODUCTION

Introduction to Web - Limitations of current Web – Development of Semantic Web – Emergence of the Social Web – Statistical Properties of Social Networks -Network analysis - Development of Social Network Analysis - Key concepts and measures in network analysis - Discussion networks - Blogs and online communities - Web-based networks.

UNIT II MODELING AND VISUALIZATION

Visualizing Online Social Networks - A Taxonomy of Visualizations - Graph Representation - Centrality- Clustering - Node-Edge Diagrams - Visualizing Social Networks with Matrix-Based Representations- Node-Link Diagrams - Hybrid Representations - Modelling and aggregating social network data - RandomWalks and their Applications –Use of Hadoop and MapReduce - Ontological representation of social individuals and relationships.

UNIT III MINING COMMUNITIES

Aggregating and reasoning with social network data, Advanced Representations - Extracting evolution of Web Community from a Series of Web Archive - Detecting Communities in Social Networks - Evaluating Communities – Core Methods for Community Detection & Mining - Applications of Community Mining Algorithms - Node Classification in Social Networks.

UNIT IV EVOLUTION

Evolution in Social Networks – Framework - Tracing Smoothly Evolving Communities - Models and Algorithms for Social Influence Analysis - Influence Related Statistics - Social Similarity and Influence - Influence Maximization in Viral Marketing - Algorithms and Systems for Expert Location in Social Networks - Expert Location without Graph Constraints - with Score Propagation – Expert Team Formation - Link Prediction in Social Networks - Feature based Link Prediction - Bayesian Probabilistic Models - Probabilistic Relational Models

UNIT V TEXT AND OPINION MINING

Text Mining in Social Networks -Opinion extraction – Sentiment classification and clustering - Temporal sentiment analysis - Irony detection in opinion mining - Wish analysis - Product review mining – Review Classification – Tracking sentiments towards topics over time.

OUTCOMES:

Upon Completion of the course, the students will be able to

- Build a social network data set from existing social networking sites
- Identify the different components of a web social network that can be used for analyzing and mining
- Identify the different data structures and graph algorithms that can be used for web social network mining
- Implement a community detection algorithm
- Process Social Network data using MapReduce paradigm
- Design an application that uses various aspects of Social Network Mining to improve its functionality and to harvest information available on the web to build recommender systems
- Analyze social media data using appropriate data/web mining techniques

REFERENCES:

1. Charu C. Aggarwal, "Social Network Data Analytics", Springer; 2011
2. Peter Mika, "Social Networks and the Semantic Web", Springer, 1st edition 2007.
3. Borko Furht, "Handbook of Social Network Technologies and Applications", Springer, 1st edition, 2010.
4. Guandong Xu , Yanchun Zhang and Lin Li, "Web Mining and Social Networking – Techniques and applications", Springer, 1st edition, 2011.
5. Giles, Mark Smith, John Yen, "Advances in Social Network Mining and Analysis", Springer, 2010.
6. Ajith Abraham, Aboul Ella Hassanien, Václav Snášel, "Computational Social Network Analysis: Trends, Tools and Research Advances", Springer, 2009.
7. Toby Segaran, "Programming Collective Intelligence", O'Reilly, 2012

313ITPT13 - KNOWLEDGE ENGINEERING

OBJECTIVES:

- To learn about proposition logic and predicate logics.
- To acquire knowledge about modal and non monotonic logics.
- To understand object oriented abstractions for various expert systems.
- To understand various planning strategies for problem solving.
- To explain the basic knowledge representation and problem solving techniques of Artificial Intelligence.
- To understand the uncertainties and expressiveness associated with the knowledge representation
- To study the various actions taken on the knowledge that is represented

UNIT I INTRODUCTION

Knowledge Representation and Reasoning – First order Logic – Syntax, Semantics Pragmatics – Expressing Knowledge – Levels of Representation – Knowledge Acquisition and Sharing – Sharing Ontologies – Language Ontologies –Language Patterns – Tools for Knowledge Acquisition

UNIT II RESOLUTION AND REASONING

Proportional Case – Handling Variables and Quantifiers – Dealing with Intractability – Reasoning with Horn Clauses - Procedural Control of Reasoning – Rules in Production- Description Logic - Issues in Engineering -Vivid Knowledge – Beyond Vivid.

UNIT III REPRESENTATION

Object Oriented Representations – Frame Formalism – Structured Descriptions – Meaning and Entailment - Taxonomies and Classification – Inheritance – Networks – Strategies for Defeasible Inheritance – Formal Account of Inheritance Networks.

UNIT IV DEFAULTS, UNCERTAINTY AND EXPRESSIVENESS

Defaults – Introduction – Closed World Reasoning – Circumscription – Default Logic Limitations of Logic – Fuzzy Logic – Nonmontonic Logic – Theories and World – Semiotics – Auto epistemic Logic - Vagueness – Uncertainty and Degrees of Belief – Noncategorical Reasoning – Objective and Subjective Probability.

UNIT V ACTIONS AND PLANNING

Explanation and Diagnosis – Purpose – Syntax, Semantics of Context – First Order Reasoning – Modal Reasoning in Context – Encapsulating Objects in Context – Agents – Actions – Situational Calculus – Frame Problem – Complex Actions – Planning – Strips – Planning as Reasoning – Hierarchical and Conditional Planning.

OUTCOMES:

- Identify the various components in a knowledge system.
- Apply the Ontology in the representation of knowledge
- Resolve the problem by logical reasoning
- Select appropriate strategies for representing the knowledge
- Resolve uncertainties associated with the knowledge representation
- Deploy the context information for the better interpretation of knowledge

REFERENCES:

1. Ronald Brachman, Hector Levesque "Knowledge Representation and Reasoning" The Morgan Kaufmann Series in Artificial Intelligence 2004
2. John F. Sowa, " Knowledge Representation: Logical, Philosophical, and Computational Foundations", 2000.
3. Arthur B. Markman, "Knowledge Representation", Lawrence Erlbaum Associates, 1998
4. Simon Kendal, Malcolm Creen, An Introduction to Knowledge Engineering, Springer, 2007
5. Thomas B.Cross, Knowledge Engineering 61 2010, Techtionary Corporation, 2010

313ITPT14 - ENERGY AWARE COMPUTING

COURSE OBJECTIVES:

This course examines the design of power efficient architecture, power and performance tradeoffs, restructuring of software and applications and standards for energy aware Hardware and Software. The objective of this course is:

- To know the fundamental principles energy efficient devices
- To study the concepts of Energy efficient storage
- To introduce energy efficient algorithms
- Enable the students to know energy efficient techniques involved to support real-time systems.
- To study Energy aware applications.

UNIT I INTRODUCTION

Energy efficient network on chip architecture for multi core system-Energy efficient MIPS CPU core with fine grained run time power gating – Low power design of Emerging memory technologies.

UNIT II ENERGY EFFICIENT STORAGE

Disk Energy Management-Power efficient strategies for storage system-Dynamic thermal management for high performance storage systems-Energy saving technique for Disk storage systems

UNIT III ENERGY EFFICIENT ALGORITHMS

Scheduling of Parallel Tasks – Task level Dynamic voltage scaling – Speed Scaling – Processor optimization- Memetic Algorithms – Online job scheduling Algorithms.

UNIT IV REAL TIME SYSTEMS

Multi processor system – Real Time tasks- Energy Minimization – Energy aware scheduling- Dynamic Reconfiguration- Adaptive power management-Energy Harvesting Embedded system.

UNIT V ENERGY AWARE APPLICATIONS

On chip network – Video codec Design – Surveillance camera- Low power mobile storage.

COURSE OUTCOMES:

Upon Completion of the course,□the students will be able to

- Design Power efficient architecture Hardware and Software.
- Analyze power and performance trade off between various energy aware storage devices.
- Implement various energy aware algorithms.
- Restructure the software and Hardware for Energy aware applications.
- Explore the Energy aware applications

REFERENCE BOOKS:

1. Ishfaq Ah mad, Sanjay Ranka, Handbook of Energy Aware and Green Computing, Chapman and Hall/CRC, 2012
2. Chong-Min Kyung, Sungjoo yoo, Energy Aware system design Algorithms and Architecture, Springer, 2011.
3. Bob steiger wald ,Chris:Luro, Energy Aware computing, Intel Press,2012.

313ITPT15 - 4G TECHNOLOGIES

OBJECTIVES:

- To learn various generations of wireless and cellular networks
- To study about fundamentals of 3G Services, its protocols and applications
- To study about evolution of 4G Networks, its architecture and applications
- To study about WiMAX networks, protocol stack and standards
- To Study about Spectrum characteristics & Performance evaluation

UNIT I INTRODUCTION

Introduction: History of mobile cellular systems, First Generation, Second Generation, Generation 2.5, Overview of 3G & 4G, 3GPP and 3GPP2 standards

UNIT II 3G NETWORKS

3G Networks: Evolution from GSM, 3G Services & Applications, UMTS network structure, Core network, UMTS Radio access, HSPA – HSUPA, HSDPA, CDMA 1X , EVDO Rev -0, Rev-A, Rev-B, Rev-C Architecture, protocol stack.

UNIT III 4G LTE NETWORKS

4G Vision, 4G features and challenges, Applications of 4G, 4G Technologies – Multi carrier modulation, Smart Antenna Techniques, OFDM-MIMO Systems, Adaptive Modulation and Coding with Time-Slot Scheduler, Bell Labs Layered Space Time (BLAST) System, Software-Defined Radio, Cognitive Radio.

UNIT IV WiMAX NETWORKS

WiMax: Introduction – IEEE 802.16, OFDM, MIMO, IEEE 802.20

UNIT V SPECTRUM & PERFORMANCE

Spectrum for LTE-Flexibility-Carrier Aggregation-Multi standard Radio base stations-RF requirements for LTE-Power level requirements-Emission requirements-Sensitivity and Dynamic range-Receiver susceptibility. Performance Assessment-Performance Evaluation

OUTCOMES:

Upon completion of the course, the students should be able to:

- Acquaint with the latest 3G/4G and WiMAX networks and its architecture.
- Interpret the various protocols and standards in various layers in Wireless networks.
- Design and implement wireless network environment for any application using latest wireless protocols and standards
- Analyze the performance of networks
- Explore the benefits of WiMax networks
- Exploit various diversity schemes in LTE

REFERENCES:

Introduction to 3G Mobile Communication, Juha Korhonen, Artech House, (www.artechhouse.com), Jan 2003, ISBN-10: 1580535070

1. 4G LTE/LTE – Advanced for Mobile Broadband, Erik Dahlman, Stefan Parkvall, Johan Skold, Academic Press 2011.

2. 3G Evolution HSPA and LTE for Mobile Broadband, Erik Dahlman, Stefan Parkvall, Johan Skold and Per Beming, Academic Press, Oct 2008, ISBN-10: 0123745381

3. UMTS Mobile Communication for the Future, Flavio Muratore, John Wiley & Sons Ltd, Jan 2001, ISBN-10: 0471498297

4. HSDPA/HSUPA for UMTS, Harri Holma and Antti Toskala, Johan Wiley & Sons Ltd, May 2006, ISBN-10: 0470018844

5. Savo G.Glisic, "Advanced Wireless Networks- 4GTechnologies", Wiley, 2006

6. Magnus Olsson, Catherine Mulligan, "EPC and 4G packet network", Elsevier 2012

7. Vijay Garg, "Wireless Communications and Networking", Elsevier, Morgan kufmann

313ITPT16 - PERFORMANCE EVALUATION OF COMPUTER SYSTEMS

OBJECTIVES:

- To understand the mathematical foundations needed for performance evaluation of computer systems
- To understand the metrics used for performance evaluation
- To understand the analytical modeling of computer systems
- To enable the students to develop new queueing analysis for both simple and complex systems
- To appreciate the use of smart scheduling and introduce the students to analytical techniques for evaluating scheduling policies

UNIT I OVERVIEW OF PERFORMANCE EVALUATION

Need for Performance Evaluation in Computer Systems – Overview of Performance Evaluation Methods – Introduction to Queueing – Probability Review – Generating Random Variables for Simulation – Sample Paths, Convergence and Averages – Little’s Law and other Operational Laws – Modification for Closed Systems.

UNIT II MARKOV CHAINS AND SIMPLE QUEUES

Discrete-Time Markov Chains – Ergodicity Theory – Real World Examples – Google, Aloha – Transition to Continuous-Time Markov Chain – M/M/1 and PASTA.

UNIT III MULTI-SERVER AND MULTI-QUEUE SYSTEMS

Server Farms: M/M/k and M/M/k/k – Capacity Provisioning for Server Farms – Time Reversibility and Burke’s Theorem – Networks of Queues and Jackson Product Form – Classed and Closed Networks of Queues.

UNIT IV REAL-WORLD WORKLOADS

Case Study of Real-world Workloads – Phase-Type Distributions and Matrix-Analytic Methods – Networks with Time-Sharing Servers – M/G/1 Queue and the Inspection Paradox – Task Assignment Policies for Server Farms.

UNIT V SMART SCHEDULING IN THE M/G/1

Performance Metrics – Scheduling Non-Preemptive and Preemptive Non-Size-Based Policies - . Scheduling Non-Preemptive and Preemptive Size-Based Policies – Scheduling - SRPT and Fairness.

OUTCOMES:

Upon completion of the course, the students will be able to

1. Identify the need for performance evaluation and the metrics used for it
2. Discuss open and closed queueing networks
3. Deploy Little’s law and other operational laws
4. Apply the operational laws to open and closed systems
5. Use discrete-time and continuous-time Markov chains to model real world systems
6. Develop analytical techniques for evaluating scheduling policies

REFERENCES:

1. Mor Harchol - Balter, "Performance Modeling and Design of Computer Systems – Queueing Theory in Action", Cambridge University Press, 2013.
2. Raj Jain, "The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation and Modeling", Wiley-Interscience, 1991.
3. Lieven Eeckhout, "Computer Architecture Performance Evaluation Methods", Morgan and Claypool Publishers, 2010.
4. Paul J. Fortier and Howard E. Michel, "Computer Systems Performance Evaluation and Prediction", Elsevier, 2003.
5. David J. Lilja, "Measuring Computer Performance: A Practitioner's Guide", Cambridge University Press, 2000.
6. Krishna Kant, "Introduction to Computer System Performance Evaluation", McGraw-Hill, 1992.
7. K. S. Trivedi, "Probability and Statistics with Reliability, Queueing and Computer Science Applications", John Wiley and Sons, 2001.

313ITPT17 - SPATIAL DATABASES

Unit 1 : Introduction

Overview – Benefits -Application Domains of Geographical Information Systems (GIS), SDBMS
Application –Stroll through spatial databases

Unit 2 : Data Models & query language

Models of spatial databases – three step data base design – ER model with spatial concepts – Data modeling with UML – Database Query language – Relational algebra – SQL primer – Extend SQL – Object relational SQL

Unit 3 : Storage Methods and Indexing

Disks –Files – Indexing – Trends – Spatial operations – Query Optimization – Index structures – Distributed Databases- parallel databases

Unit 4: Spatial networks

Network databases- Data Models-Query language for graphs- algorithms-Access methods

Unit 5 : Mining spatial databases

Pattern discovery – Motivation-Classification-Association Rule techniques-clustering –Outlier detection-Support for field entities- Content based retrieval-spatial warehouses

Text Books:

1. Spatial Databases: A Tour, S. Shekhar and S. Chawla, Prentice Hall, 2003.

References Books:

1. Moving Objects Databases, by Ralf Hartmut Guting, Markus SchneiderMorgan kaufman, 2005.
2. Spatial Databases with Applications to GIS, P. Rigaux, M. Scholl, A. Voisard, Morgan Kaufmann, 2002.
3. Spatio-Temporal Database,M. Koubarakis, T. Selles at al (ed.), Springer 2003.
4. References : Selected papers from the bibliography available at:
<http://www.spatial.cs.umn.edu/Courses/Fall07/8715/paperList.html>

313ITPT18 - MANAGEMENT FOR INFORMATION TECHNOLOGY

UNIT I Nature Of It Management

IT organisation and information - IT Management - A model for the study of IT Management.
Strategic importance of IT: Developing an IT strategy - IT planning.

UNIT II IT Trends

Software and hardware trends-Advances in telecomm-Application portfolio resources:
Introduction-Application and data resource-process for portfolio management.
Managing application development: Traditional life cycle approach-Application project management
-Resource allocation and control-Alternatives to traditional development-Successful end user
computing.

UNIT III Tactical And Operational Considerations

Developing and managing customer expectations. Problem-Change and recovery management-
Managing production operations-Network management.

UNIT IV Controlling The Information Resources

Accounting for IT resources-IT controls and asset protection.

UNIT V Preparing For Advances In It

People-Organ-Management systems. People Management-Management processes-Role of CIO.

Text Books :

1. Carrol W.Frenzel, "Management of Information Technology", Boyd and Frosee Publishing
Company, 1992

Reference Books

1. Dickson, Gary N, James Wetherbe, "Management Information System", McGraw Hill, 1985

313ITPT19 - INTRODUCTION TO MACHINE LEARNING

UNIT I

Introduction-Machine Learning-Supervised Learning-Model Selection and Generalization-Bayesian Decision Theory-Losses and Risks-Association Rules-Parametric Methods-Likelihood Estimation-Bayes' Estimator-Multivariate Methods-Multivariate Data-Discrete Features

UNIT II

Dimensionality Reduction-Factor Analysis-Multidimensional Scaling-Locally Linear Embedding-Clustering-Mixture Densities-k-means Clustering-Expectation Maximization Algorithm-Supervised learning after clustering—Histogram Estimator-Kernel Estimator-k-nearest neighbor Estimator-Generalization to multivariate data-Condensed nearest neighbor-Nonparametric Regression-Running line smoother-Decision Trees-Univariate Trees-Pruning-Rule Extraction for trees

UNIT III

Linear Discrimination-Generalizing Linear Model-Pair wise Separation-Parametric Discrimination revisited-Gradient Descent-Multilayer Perceptrons-Introduction-Training the perception-Boolean Functions-Multilayer Perceptions-MLP as universal Approximator - Back Propagation Algorithm-Tuning the network size-Dimensionality reduction-Recurrent networks-Local Models

UNIT IV

Kernel Machines-Estimation of parameter Distribution - Regression-Gaussian Process-Hidden Markov Models-three basic problems of HMM-HMM-Graphical Models-Example-d-Separation-Belief Propagation-Undirected graphs-Combining multiple learning-rationale-generating diverse learners-model combination schemes-voting-error correcting-Boosting -Cascading-Reinforcement Learning-Model based learning-Temporal Difference learning-Partially observable states

UNIT V

Design and Analysis of Machine Learning Experiments—Response Surface design-Cross validation and resampling methods-Measuring classifier performance-Comparison of classification algorithms-Comparing multiple algorithms-Probability

Text Book:

1. Introduction to Machine Learning ,Second Edition -Ethem Alpaydin

Reference Books

1. Bayesian Reasoning and Machine Learning, David Barber
2. Gaussian Processes for Machine Learning ,Carl E. Rasmussen, Christopher K. I. Williams - The MIT Press , 2005
3. Introduction to machine learning, Yves Kodratoff.

Registrar