# M.Tech 1 Year I Semester (Computer Science & Engineering)

## COURSE STRUCTURE

<table>
<thead>
<tr>
<th>Sl.No.</th>
<th>Code</th>
<th>COURSE</th>
<th>L</th>
<th>P</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CSEP1 1401</td>
<td>Data Structures and Algorithms</td>
<td>3+1*</td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>CSEP1 1402</td>
<td>Database Management Systems</td>
<td>3+1*</td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>CSEP1 1403</td>
<td>Operating Systems</td>
<td>3+1*</td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>MEP1 1401</td>
<td>Advanced Optimization Techniques</td>
<td>3+1*</td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>Elective – 1</td>
<td></td>
<td>3+1*</td>
<td></td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>CSEP1 1404</td>
<td>Computer Organization &amp; Architecture</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CSEP1 1405</td>
<td>Cryptography and Network Security</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CSEP1 1406</td>
<td>Information Retrieval Systems</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Elective – 2</td>
<td></td>
<td>3+1*</td>
<td></td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>CSEP1 1407</td>
<td>Bio – Informatics</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CSEP1 1408</td>
<td>Embedded Systems</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>CSEP1 1409</td>
<td>Object Oriented Software Engineering</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>CSEP11210</td>
<td>Data Structures and Algorithms Laboratory</td>
<td></td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>8</td>
<td>GMRP10206</td>
<td>Term Paper</td>
<td></td>
<td></td>
<td>3</td>
</tr>
</tbody>
</table>

**Total Credits**: 30

---

# M.Tech 1 Year II Semester (Computer Science & Engineering)

## COURSE STRUCTURE

<table>
<thead>
<tr>
<th>Sl.No.</th>
<th>Code</th>
<th>COURSE</th>
<th>L</th>
<th>P</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CSEP1 1411</td>
<td>Data Warehousing and Data Mining</td>
<td>3+1*</td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>CSEP1 1412</td>
<td>Mobile Computing</td>
<td>3+1*</td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>CSEP1 1413</td>
<td>Soft Computing Techniques</td>
<td>3+1*</td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>Sl.No.</td>
<td>Code</td>
<td>COURSE</td>
<td>L</td>
<td>P</td>
<td>C</td>
</tr>
<tr>
<td>-------</td>
<td>-----------</td>
<td>---------------------------------------------</td>
<td>---</td>
<td>----</td>
<td>-----</td>
</tr>
<tr>
<td>4</td>
<td>CSEP1 1414</td>
<td>Web Technologies</td>
<td></td>
<td></td>
<td>3+1*</td>
</tr>
<tr>
<td>5</td>
<td>Elective – 3</td>
<td></td>
<td>3+1*</td>
<td></td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>CSEP1 1415</td>
<td>Compiler Design</td>
<td></td>
<td></td>
<td>3+1*</td>
</tr>
<tr>
<td></td>
<td>CSEP1 1416</td>
<td>Distributed Systems</td>
<td></td>
<td></td>
<td>3+1*</td>
</tr>
<tr>
<td></td>
<td>CSEP1 1417</td>
<td>Open Source Software</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Elective – 4</td>
<td></td>
<td></td>
<td></td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>CSEP1 1418</td>
<td>Cloud Computing</td>
<td></td>
<td></td>
<td>3+1*</td>
</tr>
<tr>
<td></td>
<td>CSEP1 1419</td>
<td>Image Processing &amp; Pattern Recognition</td>
<td></td>
<td></td>
<td>3+1*</td>
</tr>
<tr>
<td></td>
<td>CSEP1 1420</td>
<td>Social Networks and Semantic Web</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>CSEP11221</td>
<td>Web Technologies Lab</td>
<td></td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>8</td>
<td>GMRP10202</td>
<td>Comprehensive Viva – voce</td>
<td></td>
<td></td>
<td>3</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Total Credits</strong></td>
<td></td>
<td></td>
<td>30</td>
</tr>
</tbody>
</table>

**M.Tech  II Year III & IV Semesters (Computer Science & Engineering)**

**COURSE STRUCTURE**

<table>
<thead>
<tr>
<th>Sl.No.</th>
<th>Code</th>
<th>COURSE</th>
<th>L</th>
<th>P</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>GMRP20403</td>
<td>Internship and project work</td>
<td></td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>GMRP22005</td>
<td>Project Work</td>
<td></td>
<td></td>
<td>20</td>
</tr>
</tbody>
</table>

*Tutorial
Course Title: DATA STRUCTURES AND ALGORITHMS

Course Code: CSEP 1401

L T P C
3 1 0 4

Objectives:

At the end of the course students are expected to:

This course couples work on program design, analysis, and verification with an introduction to the study of data structures. Data structures capture common ways to store and manipulate data, and they are important in the construction of sophisticated computer programs.

- It improves the expected performance of a new data structure technique using randomization technique and to impart knowledge to students on Hashing.
- It deals with a data structure technique operations applied to elements are based on its priority and also analyzed the methods of external sorting.
- It starts with non-linear data structure and it provides knowledge to develop the tree structures suitable for the representation of a dictionary and also developed the balanced search tree structure to provide better worst-case performance.
- Developed more on balanced search tree structure to provide improved performance.
- Finding all occurrences of a pattern in a text is a problem that arises frequently in text editing programs like document being edited or searching a particular pattern/word/string given by the user.

Outcomes:

- At the end of this course student should be able to write large, reliable programs composed from reusable pieces, students can also develop modular programs on data structures and algorithms that are easy to read, debug, verify, analyze, and modify using C/C++ language.
- At the end of this course student should be able to perform basic analysis of algorithms, understand how various data structures and algorithms, be able to implement them in a high-level language, and be able to pick an appropriate data structure or algorithm for a given task.
Unit I:

**Searching**: Linear and Binary search methods, Sorting - Bubble sort, Selection sort, Insertion sort, Quick sort and Merge sort.

Unit II:
**Trees** - Binary tree, properties, representation and traversals (DFT, BFT) expression trees (infix, prefix, postfix). Graphs - basic concepts, storage structures and traversals.

**Dictionaries**, ADT, The list ADT, Stack ADT, Queue ADT, hash table representation, hash functions collision resolution - separate chaining, open addressing - linear probing, double hashing.

Unit III:
**Priority Queues** - Definition, ADT, Realizing a Priority Queue using Heaps, Definition, insertion, Deletion. External Sorting - Model for external sorting, Multiway merge.

**Search Trees**: Binary Search Trees, Definition, ADT, Implementation, Operations – Searching, Insertion and Deletion.

Unit IV:
**Search Trees**: AVL Trees, Definition, Height of an AVL Tree, Operations – Insertion, Deletion and Searching.

**Search Trees**: Introduction to Red - Black and Splay Trees, B - Trees, height of a B - Tree, insertion, deletion and searching, Comparison of Search Trees.

Text Books:
2. Data Structures, Algorithms and Applications in C++, 2/e, SartajSahni, University Press.

Reference Books:
1. Data structures and Algorithm Analysis, 2/e, Mark Allen Weiss, Pearson.
2. Data structures and Algorithms, 3/e, Adam Drozdek, Cengage.
4. Data Structures, Algorithms and OOP, Heileman, TMH.

9. Data Structures, Seymour Lipschutz, Schaum's outlines, TMH.
Course Title: DATABASE MANAGEMENT SYSTEMS       Course Code : CSEP1 1402

L      T    P    C
3      1  0    4

OBJECTIVES:

- Understand the differences between File system and DBMS, Data Models and database system structure.
- Learn how to use the integrity constraints over the relations and know the expressive power of Algebra and calculus.
- Learn the query language features which are the core of SQL’s DML, Join operations and Triggers.
- Students know the concept of the transaction management which is the foundation for concurrent execution and recovery from the system failure in a DBMS.
- Learn the recovery techniques for managing the database effectively and avoid the data loss.
- Know how to arrange the records in a file when the file is stored on the external storage.

OUTCOMES:

- Apply normalization process on existing database for eliminating redundancy.
- Carryout initial database design for large enterprises by learning feature of ER-Models.
- Apply the recovery techniques for managing the database effectively and avoid the data loss.
- Design database for large enterprises by learning eliminate redundancy by using the features of Normal Forms.
- Develop efficient database management systems.
Unit I:
INTRODUCTION - Basics of database systems, Traditional file oriented approach, Motivation for database approach, Evolution of database systems, Database basics, Views of data, Three level architecture of DBMS, Relational database systems, Data models, Database languages, Client-server and multi-tier architectures, Multimedia data, Information integration, Data-definition language commands, Overview of query processing, Storage and buffer management, Transaction processing, The query processor. Use Of SQL, DDL Statements, DML Statements, View Definitions, Constraints and Triggers, Keys and Foreign Keys, Constraints on Attributes and Tuples, Modification of Constraints, Cursors, Dynamic SQL.


Unit II:
REPRESENTING DATA ELEMENTS - Data Elements and Fields, Representing Relational Database Elements, Records, Representing Block and Record Addresses, Client-Server Systems, Logical and Structured Addresses, Record Modifications, Index Structures, Indexes on Sequential Files, Secondary Indexes, B-Trees, Hash Tables.

RELATIONAL ALGEBRA - Relational Algebra: Basics of Relational Algebra, Set Operations on Relations, Extended Operators of Relational Algebra, Constraints on Relations, Modification of the Database, Views Relational Calculus, Tuple Relational Calculus, Domain Relational Calculus.

NORMALIZATION - Normalization, First Normal form, Second Normal Form, Third Normal Form, BCNF, Multi-valued dependency, Fifth Normal Form.

Unit III:


Unit IV:
TRANSACTION MANAGEMENT - Introduction of Transaction management, Serializability and Recoverability, View Serializability, Resolving Deadlocks, Distributed Databases, Distributed Commit, Distributed Locking.

DATABASE SYSTEM ARCHITECTURE - Centralized and Client-Server Architectures, Server System Architectures, Parallel Systems, Distributed Systems, Network Types.

DISTRIBUTED DATABASE - Homogeneous And Heterogeneous Database, Distributed Data Storage, Distributed Transaction, Commit Protocols, Concurrency Control In Distributed Databases, Availability, Heterogeneous.

Text Books:-

Reference Books:-
1) Introduction to Database Systems, 8/e, C.J.Date, Pearson.
2) Database System design, Implementation, and Management, 5/e, Rob, Coronel, Thomson.
3) ShamKanth B. Navathe, Fundamental of Database System, Pearson Education.
M.Tech I Year I Semester (Computer Science & Engineering)
A Y 2013-14

Course Title: OPERATING SYSTEMS

Course Code: CSEP1 1403

L T P C
3 1 0 4

OBJECTIVES:

• Understand functions, objectives services of Operating Systems
• Learn the concepts of the process and threads
• Provides a broad understanding of issues related to concurrency
• Focus on principles of deadlock and related problems of starvation
• Provides an overview of the mechanisms used in memory management
• Understand about design issues related to processor scheduling
• Understand Distributed Operating Systems to some extent

OUTCOMES:

• Analyze the differences between process and threads
• Able to Implement concurrency mechanisms
• Implement Bankers Algorithms to handle deadlocks
• Design and Analyze mechanisms used in memory management
• Develop processor scheduling, Paging technique Algorithms
Unit I:

**Overview** - Introduction, design approaches, why advanced operating systems, types of advanced operating systems. Distributed Systems – Hardware and Software concepts, Architectures of Distributed Systems - System Architecture types, issues in distributed operating systems.

Unit II:

**Process description and control** - process states, process description, process control, CPU Scheduling – Scheduling criteria, Types of processor scheduling and Scheduling algorithms, Multiple-processor Scheduling, Process and Threads- Multi threading, ULTS and KLTS.

Unit III:
**Memory Management and Virtual Memory** - Logical Vs Physical address space, Swapping, Contiguous allocation, Paging, Segmentation, Segmentation with paging, Demand paging, Performance of demand paging, Page Replacement algorithms.


Unit IV:
**I/O management and disk scheduling** – I/O devices, organization of I/O functions; I/O buffering, disk scheduling, Disk cache, File management – organization, directories, file sharing, record blocking, secondary storage management; case studies-LINUX I/O.

**SHARED MEMORY AND FILE SYSTEMS**


**Text Books:**
2) Andrew S Tanenbaum, “Distributed Operating Systems”, Pearson Education India,

**Reference Books:**
3) Operating Systems, 2/e, Dhamdhere, TMH
Course Title: ADVANCED OPTIMIZATION TECHNIQUES  
Course Code: MEP1 1401

<table>
<thead>
<tr>
<th>L</th>
<th>T</th>
<th>P</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>1</td>
<td>0</td>
<td>4</td>
</tr>
</tbody>
</table>

**Objectives:**

This course is designed for first year M.Tech students. The course is intended to make the students understand the basic concepts and advanced concepts of optimization techniques.

The main objective of the course is to:

- Develop systematic approach to handle problems to design of electrical circuit etc; with a goal of maximizing the profit and minimizing cost.
- Understand the various optimization techniques such as classified optimization, linear programming, one dimensional minimization methods, unconstrained optimization techniques, constrained optimization techniques and dynamic programming.
- Understand the necessary sufficient conditions for finding the solution of the problems in classical optimization.
- Comprehend the numerical methods for finding approximate solution of complicated problems.
- Apply methods like north west corner rule, least count method etc. to solve the transportation problem.

**OUTCOMES:**

- Design of mechanical systems and interdisciplinary engineering applications and business solutions using suitable optimization technique.
- Apply numerical or iterative techniques in power systems for optimal power flow solutions.
- Optimize the parameters in control systems for desired steady state or transient response.
- Optimize the cost function in deciding economic factors of power systems.
- Design of electrical systems optimally using suitable techniques like univariate method, steepest descent method etc.
UNIT – I:
**Linear programming**-Two-phase simplex method, Big-M method, duality, interpretation, applications.

**Assignment problem**- Hungarian’s algorithm, Degeneracy, applications, unbalanced problems, traveling salesman problem.

UNIT – II:
**Classical optimization techniques**-Single variable optimization with and without constraints, multi – variable optimization without constraints, multi – variable optimization with constraints – method of Lagrange multipliers, Kuhn-Tucker conditions.

**Numerical methods for optimization**-Nelder Mead’s Simplex search method, Gradient of a function, Steepest descent method, Newton’s method, types of penalty methods for handling constraints.

UNIT – III:
**Genetic algorithm (GA)** - Differences and similarities between conventional and evolutionary algorithms, working principle, reproduction, crossover, mutation, termination criteria, different reproduction and crossover operators, GA for constrained optimization, drawbacks of GA.

**Genetic Programming (GP)**-Principles of genetic programming, terminal sets, functional sets, differences between GA & GP, random population generation, solving differential equations using GP.

UNIT – IV:
**Multi-Objective GA**-Pareto’s analysis, Non-dominated front, multi – objective GA, Nondominated sorted GA, convergence criterion, applications of multi-objective problems.

Basic Problem solving using Genetic algorithm, Genetic Programming & Multi Objective GA and simple applications of optimization for engineering systems.

**Text Books:**
2. Optimization for Engineering Design – Kalyanmoy Deb, PHI Publishers

**References:**
2. Genetic Programming- Koza
3. Multi objective Genetic algorithms - Kalyanmoy Deb, PHI Publishers
M.Tech I Year I Semester (Computer Science & Engineering)
A Y 2013-14

Course Title: COMPUTER ORGANIZATION AND ARCHITECTURE (Elective 1)

Course Code: CSEP1 1404

L    T    P   C
3      1  0    4

Objectives:
This course is designed for first year M.Tech students. The course is intended to make the students understand the basic concepts of number system and computer arithmetic, Analysis and design of sequential circuits using several types of flip flops, Analyze Intel 8086 instruction format and CPU architecture, Input-Output organization.
The main objective of the course is to:
- Understand the various binary systems representing information in digital system.
- Analysis and design of combinational and sequential circuits using several types of flip flops with minimum number of logic gates.
- Understand the concept of memory structures of shift registers
- Understand the different arithmetic operations that are involved in different hardware component in ALU.
- Acquire knowledge of basic cache memory with basic organizational techniques and performance tuning and Input-output organization.
- Understand Intel 8086 CPU architecture, Assembly language Instructions and programming.

Outcomes:
- Apply knowledge of number systems, codes and Boolean algebra to the analysis and design of digital logic circuits
- Use flip-flops in designing sequential logic circuits and counters for digital circuits
- Design a digital system, components or process to meet desired needs within realistic constraints by forming multi-disciplinary teams.
- Evaluate quality of modern computing systems
- Assess functionality of their components, design of instruction sets and their underlying execution.
- Analyze the performance of computer systems and know how to improve their efficiency (pipelining, caches, etc).

- Apply cache design to solve performance problems in other aspects of computer design.
- Design an I/O subsystem supporting processor programmed I/O, direct memory access, interrupt structures, and industry standards.
- Understand the full internal workings of a typical simple CPU including the utilization of the various hardware resources during the execution of instructions.
- Introduce the design of basic I/O hardware and microprocessor interfacing: memory chip selection, memory expansion, I/O interfacing, different I/O techniques: interrupts, DMA.
**Unit I:**
**Number Systems and Computer Arithmetic**- Signed and unsigned numbers, Addition & subtraction, multiplication division, Floating point representation, logical operation, Gray code, BCD codes, Error detecting codes. Boolean algebra, Simplification of Boolean expressions - Maps.

**Combinational and Sequential Circuits**- Decoders, Encoders, Multiplexers, Half and Full adders, shift registers, flip-flops, binary counters, memory Unit.

**Unit II:**
**Memory Organization**- Memory hierarchy, Main memory, Auxiliary Memory, Associative Memory, Cache Memory, Virtual Memory Concept.

**Intel 8086 CPU Architecture**- Diagram, code and segment registers, internal operations, machine language instructions (addressing modes, instruction formats), Instruction execution timing.

**Unit III:**
**Intel 8086 Assembly Language Instructions**- Data transfer instructions - input output instructions, address transfer Flags, arithmetic, logical shift and rotate instructions.

**Intel 8086 Assembly Language Programming**- Conditional and unconditional transfer, iteration control, interrupts and process control instructions, assembler directives Programming with assembly language instructions.

**Unit IV:**
**ALU Design**- Addition and subtraction sign and unsigned numbers, multiplication and division algorithms, BCD Adders.

**Input-Output Organization**- Peripheral devices, input-output interface, Asynchronous data transfer, Modes of transfer, priority interrupts, DMA, Input output processor, Serial communication.

**Text Books:**
2. Microprocessor and interfacing, 2/e, Douglas V. Hall, TMH.

**Reference Books:**
1. Digital Logic and computer organization, Rajaraman, Radha Krishnan, PHI.
2. Microcomputer systems: 8086/8088 family, 2/e, Liu, Gibson, PHI
3. Computer Organization and Architecture, 7/e, stallings, pearson
4. Computer Organization, 5/e, Hamacher, Vranesic, TMH
5. Computer systems organization and architecture, carpinelli,pearson
6. Computer organization and Design, pal chowdary, PHI
7. Computer system organization, jotwani, TMH
Course Title: CRYPTOGRAPHY AND NETWORK SECURITY (Elective – 1) 

Objectives:

- Understand about various Conventional Encryption Principles designed for providing security.
- Learn public key cryptography, key management principles and Learn Pretty Good Privacy (PGP) which is a computer program that provides cryptographic privacy and authentication.
- Learn IP Security fundamentals, architecture and identifying the key features IP security system.
- Describes how to achieve data confidentiality, data integrity, data authentication and authentication for participants during data transit.
- Identify and evaluate current and emerging technologies in security and assess their applicability to address the users’ needs and recognize the need for continued learning throughout their career.

Outcomes:

- Know the information security priorities and information assets.
- Design information security strategy and the architecture.
- Identify and prioritize threats to information assets.
- Understand of professional, ethical, legal, security and social issues and responsibilities.
- Apply current techniques, skills, and tools necessary for providing data and network security.

UNIT II:
Number Theory - Prime and Relatively prime numbers, Modular arithmetic, Fermat’s and Euler’s Theorems, Testing for primarily, Euclid’s Algorithm, the Chinese remainder theorem, Discrete logarithms.
Public Key Cryptography - Principles, RSA Algorithm, Key Management, Diffie-Hellman Key exchange, Elliptic Curve Cryptography.
Message Authentication and Hash Functions - Authentication requirements and functions, Message Authentication, Hash functions, Security of Hash functions and MACs.
Hash and MAC algorithms - MD file, Message digest Algorithm, Secure Hash Algorithm, RIPEMD-160, HMAC.

UNIT III:
Electronic Mail Security: Pretty Good Privacy, S/MIME.

UNIT IV:
Intruders, Viruses and Worms - Intruders, Viruses, Worms, Bacterias.
Fire Walls: Fire wall Design Principles, Trusted systems.

TEXT BOOKS

REFERENCE BOOKS
M.Tech  I Year  I Semester (Computer Science & Engineering)  
A Y 2013-14  
Course Code: CSEP1 1406

Course Title: INFORMATION RETRIEVAL SYSTEMS (Elective 1)  

L  T  P  C
3  1  0  4

Objectives:
This course is concerned with the extraction of relevant information from large collections of documents. It has applications to proprietary retrieval systems. This course will aim to provide students with an overview of the main principles and methods underlying the domain of Information Retrieval.

- Introduce the information system capabilities, Cataloging and indexing.
- This course also provides to learn document and term clustering.
- It deals with various information system evolution and multimedia information retrieval.

Outcomes:
- Students are able to understand overview of the main principles and methods underlying the domain of information retrieval.
- Students are able to know cluster the documents of the real world data.
- Understand more recent developments in IR such as collaborative filtering and Latent Semantic Indexing.
- After complete this course students are able to make construction of information retrieval will furthermore acquire practical experience in the construction of IR systems by a series of projects.
UNIT I:
Introduction- Definition, Objectives, Functional Overview, Relationship to DBMS, Digital libraries and Data Warehouses.

Information Retrieval System Capabilities - Search, Browse, Miscellaneous.


UNIT II:
Data Structures- Introduction, Stemming Algorithms, Inverted file structures, N-gram data structure, PAT data structure, Signature file structure, Hypertext data structure.

Automatic Indexing- Classes of automatic indexing, Statistical indexing, Natural language, Concept indexing, Hypertext linkages.

UNIT III:
Document and Term Clustering- Introduction, Thesaurus generation, Item clustering, Hierarchy of clusters.

User Search Techniques- Search statements and binding, Similarity measures and ranking, Relevance feedback, Selective dissemination of information search, Weighted searches of Boolean systems, Searching the Internet and hypertext.

UNIT-IV:
Information Visualization- Introduction, Cognition and perception, Information visualization technologies.

Text Search Algorithms- Introduction, Software text search algorithms, Hardware text search systems.

Information System Evaluation- Introduction, Measures used in system evaluation, Measurement example – TREC results.


TEXT BOOKS:

REFERENCE BOOKS:
M.Tech I Year I Semester (Computer Science & Engineering)
A Y 2013-14

Course Code: CSEP1 1407

Course Title: BIO-INFORMATICS (Elective 2)

<table>
<thead>
<tr>
<th>L</th>
<th>T</th>
<th>P</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>1</td>
<td>0</td>
<td>4</td>
</tr>
</tbody>
</table>

Objectives:
- Understand the functions, objectives of Molecular Biology and Bio-Informatics
- Learn the concepts of the structure of Genes & DNA and their Visualization
- Understand the structural prediction, comparison and classification of proteins
- Focus on principles of Searching from Databases
- Utilize and understand statistical methods for the analysis of large datasets where appropriate
- Understand about the Pattern Matching Concepts
- Understand the various mathematical and statistical models of gene regulatory network
- Understand the simulation and Modeling of Drug Design using various Algorithms

Outcomes:
- Understand electronic storage and sharing of biological information
- Able to Implement Search strategies from Biological Databases
- Get knowledge of laboratory technologies used in biology to generate large datasets
- Design effective data representations for the storage and manipulation of large datasets where needed
- Able to generate virtual gene expression data with statistical and biological plausibility
UNIT I:

Molecular Biology Primer - Genes, Molecules, Structure of DNA, Proteins Analysis.

UNIT II:
Searching Biological Data From databases - Finding the information stored and its retrieval methods at NCBI, EMBL and DDBJ, Protein Data Bank (PDB), NDB, CCSD, PIR, Swiss Prot, EMBL, Pfam, EST, SNP, Metabolic pathways databases, EMP, KEGG, MetaCyc, structural databases: SCOP, CATH, Retrieving microbial and viral genome information.

Data Visualization - Data Visualization, sequence visualization, structure visualization, user Interface, Animation Versus simulation, General Purpose Technologies.

UNIT III:

Data Mining - Clustering and Classification, Data Mining, Methods, Selection and Sampling, Preprocessing and Cleaning, Transformation and Reduction, Data Mining Methods, Evaluation, Visualization, Designing new queries, Pattern Recognition and Discovery, Machine Learning, Text Mining, Tools.

UNIT IV:


TEXT BOOKS

REFERENCE BOOKS
3. Bio-Informatics - Methods and Applications, Rastogi, Mendiratta, Rastogi, PHI
Course Code: CSEP1 1408

Course Title: EMBEDDED SYSTEMS (Elective-2)

<table>
<thead>
<tr>
<th>L</th>
<th>T</th>
<th>P</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>1</td>
<td>0</td>
<td>4</td>
</tr>
</tbody>
</table>

Objectives:
- Students will understand about embedded computing and study the internal architecture of 8051 microcontroller.
- To learn programming on 8051 microcontroller using various tools.
- Understand how to interface i/o devices, A/D and D/A conversions. Introduction about real time operating system.
- Learn various design concepts of real time operating system like semaphores, queues, scheduling, etc.

Outcomes:
- An understanding of embedded computing and internal architecture of 8081 microcontroller.
- Students able to do programming in 8051 using tools.
- Demonstrate the interfacing of various i/o, A/d and D/A devices with 8051 microcontroller.
- Develop various algorithms for real time operating system scheduling, semaphores, etc.
Unit I:


Unit II:


Unit III:

**Applications** - Interfacing with Keyboards, Displays, D/A and A/D Conversions, Multiple Interrupts, Serial Data Communication.

**Introduction to Real – Time Operating Systems** - Tasks and Task States, Tasks and Data, Semaphores, and Shared Data; Message Queues, Mailboxes and Pipes, Timer Functions, Events, Memory Management, Interrupt Routines in an RTOS Environment.

Unit IV:


**Introduction to advanced architectures** - ARM and SHARC, Processor and memory organization and Instruction level parallelism; Networked embedded systems: Bus protocols, I2C bus and CAN bus; Internet-Enabled Systems, Design Example-Elevator Controller.

TEXT BOOKS:
1. Computers as Components-principles of Embedded computer system design, Wayne Wolf, Elseveir.

REFERENCES:
1. Embedding system building blocks, Labrosse, via CMP publishers.
2. Embedded Systems, Raj Kamal, TMH.
3. Micro Controllers, Ajay V Deshmukhi, TMH.
5. Microcontrollers, Raj kamal, Pearson Education.
6. An Embedded Software Primer, David E. Simon, Pearson Education.
Course Code: CSEP1 1409

Course Title: OBJECT ORIENTED SOFTWARE ENGINEERING (Elective 2)  
L  T  P  C  
3  1  0  4

Objectives:
- Understand about process models and learn to know how to specify software requirements.
- Learn how to perform feasibility study of the projects under the requirement engineering process and system models.
- Students can learn design and implementation of software and the management of the software project development.
- Deriving the use cases from requirements.

Outcomes:
- An understanding of the object oriented methodology and how it is applicable in the design and implementation of software and the management of the software project.
- He is able to use UML, an object oriented design tool, in the realization of a given software engineering project.
- Design real world projects to make success of the project.
- Develop sketch out an idea on a blackboard or a scrap of paper in order to visualize a part of a system.
- Design and develop real-time software projects with effective cost estimation and plan.
Unit I:
**Introduction to Classical software Engineering** - Historical, Economic and Maintenance aspects.
Introduction to Object Oriented Paradigm. Different phases in structured paradigm and Objective Oriented Paradigm. Software Process and different life cycle models and corresponding strengths and weaknesses.

**Planning and Estimation** - Estimation of Duration and Cost, COCOMO components of software. Project Management plan, one case Study.

Unit II:
**Tools for step wise refinement** - Cost - Benefit analysis, Introduction to software metrics and CASE tools. Taxonomy and scope of CASE tools. Introduction to testing, with focus on Utility, Reliability, Robustness, Performance, Correctness.

**Modules to objects** - Cohesion and Coupling, Data Encapsulation and Information hiding aspects of Objects. Inheritance, polymorphism and Dynamic Binding aspects. Cohesion and coupling of objects. Reusability, Portability and Interoperability aspects.

Unit III:

**Analysis phase** - Use case Modeling, Class Modeling, Dynamic Modeling.

Unit IV:
**Design phase** - Data oriented design, Object Oriented design, Formal techniques for detailed design. One case study.

**IIM Phases** - Implementation, Integration and maintenance phases.

**TEXT BOOKS**
1. Object oriented and Classical Software Engineering, 7/e, Stephen R. Schach, TMH
2. Object oriented and classical software Engineering, Timothy Lethbridge, Robert Laganiere, TMH

**REFERENCE BOOKS**
M.Tech 1 Year I Semester (Computer Science & Engineering)  
A Y 2013-14

Course Code: CSEP1 1210

Course Title: DATA STRUCTURES AND ALGORITHMS LABORATORY

T   P   C
0   3   2

Objectives: At the end of the course students are expected to:

The main objective of this course is to enable the student to implement programs of DS using the concepts of C language. To implement various data structures that applied to solve the problems of real world in an efficient manner.

The learning objectives of the course are given as follows:

- To be able to implement the fundamental concepts of DS like stack, queue, linked list etc.
- To implement the concepts of avl trees, b+ trees.
- To implement concepts of hashing techniques.
- To apply the concepts of searching and sorting

Outcomes: At the end of this course, the students will demonstrate the knowledge of various concepts in DS Programming. They are capable of developing programs of for various applications. They also will have the sound knowledge of data structures and their application in the real world. They will be able to apply this knowledge in solving any problem by developing programs of better efficiency in terms of time complexity and length. Learning outcomes can be assessed through their performance in mid-term examinations and end-semester examinations. Also assignments can be given at the end of each unit from the important questions appearing in university question papers.

1. Perform various operations by using singly linked list, doubly linked list and circular linked list.
2. Implement Bubble sort, selection sort, Insertion sort, Quick sort, and merge sort.
3. Create a binary tree, perform insertion and deletion operations.
4. Perform Infix and postfix operations by using binary search tree.
5. To implement operations on graphs
   a) vertex insertion
   b) Vertex deletion
   c) finding vertex
   d) Edge addition and deletion
6. Implement Depth First Search (DFS) for a graph nonrecursively.
7. Implement Breadth First Search (BFS) for a graph non recursively.
9. Perform various operations i.e., insertions and deletions on AVL trees.
10. Perform various operations i.e., insertions and deletions on B-Tree.
Course Title: DATA WAREHOUSING AND DATA MINING

Objectives:

- Understand the role of data warehouse as information repository.
- Learn different kind of information and statistics of the data by using various operations such as roll-up, roll-down, drill-up, drill-down, etc.
- Learn how to extract the hidden knowledge from the data warehouse or repository for decision making by using one or more of the data mining tasks such as Data characterization, Association rule mining, Classification and Clustering.
- Study the characterization and describing data by generalization, summarization and comparison of the given data.
- Establish association between different data groups by discovering frequent patterns.
- Design a classifier model, first to learn from the given dataset and then to classify the given data into one of the specified classes.
- Use one of the many clustering algorithms that the students learn to classify/group the data into unspecified number of classes and also to detect outliers for the applications such as fraud detection, etc.
- Scale the existing data mining algorithms or to design new algorithms for mining of large and ever growing databases.

Outcomes:

- Demonstrate the knowledge of warehouse and information repository.
- Develop various extraction of knowledge form the data repository.
- Develop market basket analysis methods for market analysis
- Implement projects using data mining tools to consult corporate industries
UNIT I:
Data Warehouse and OLAP - Data Warehouse and OLAP Technology for Data Mining Data Warehouse, Multidimensional Data Model, Data Warehouse Architecture, Data Warehouse Implementation, Further Development of Data Cube Technology, From Data Warehousing to Data Mining.

Fundamentals of Data Mining, Data Mining Functionalities - Classification of Data Mining systems, Major issues in Data Mining.Data Preprocessing: Needs Preprocessing the Data, Data Cleaning, Data Integration and Transformation, Data Reduction, Discretization and Concept Hierarchy Generation.

UNIT II:
Measures of similarity and dissimilarity - Basics, Similarity and Dissimilarity between Simple Attributes, Dissimilarities between Data objects, Similarities between Data objects, Examples of proximity measures, Issues in proximity Calculations, selection the right proximity Measure.

Concepts Description , Characterization and Comparison - Data Generalization and Summarization-Based Characterization, Analytical Characterization: Analysis of Attribute Relevance, Mining Class Comparisons: Discriminating between Different Classes, Mining Descriptive Statistical Measures in Large Databases.

UNIT III:
Mining Association Rules in Large Databases - Association Rule Mining, Mining Single-Dimensional Boolean Association Rules from Transactional Databases, Mining Multilevel Association Rules from Transaction Databases, Mining Multidimensional Association Rules from Relational Databases and Data Warehouses, From Association Mining to Correlation Analysis, Constraint-Based Association Mining. Classification - Preliminaries, General approach to solving a classification problem, Decision tree induction, Rule-based classifier, Nearest-Neighbor classifiers, Bayesian classifiers, Artificial Neural Network.

UNIT IV:

Mining Complex Types of Data - Multidimensional Analysis and Descriptive Mining of Complex, Data Objects, Mining Spatial Databases, Mining Multimedia Databases, Mining Time-Series and Sequence Data, Mining Text Databases, Mining the World Wide Web.

TEXT BOOKS:
1. Data Mining, Concepts and Techniques, Jiawei Han & Micheline Kamber Harcourt India.
2. Introduction to Data Mining, Pang-Ning Tan, Michael Steinbach, Vipin Kumar, Pearson Education.

REFERENCE BOOKS:
1. Data Mining Introductory and advanced topics, Margaret H Dunham, Pearson Education.
M.Tech  I Year I Semester(Computer Science & Engineering)  
A Y 2013-14

Course Code: CSEP1 1412

Course Title: MOBILE COMPUTING

L  T  P  C
3  1  0  4

OBJECTIVES

- Understand various technologies & applications in wireless and mobile environments and to learn in detail about GSM.
- Learn various medium access schemes (MAC) viz. SDMA, FDMA, TDMA and CDMA.
- Get knowledge about Mobile IP and IP packet delivery from a fixed node to a mobile node and vice-versa.
- Understand in detail about Mobile Adhoc Networks (MANETs) and various routing algorithms used.
- Learn the concepts of various data delivery mechanisms and selective tuning techniques.
- Know about latest protocols and tools used in mobile environments like WAP, Bluetooth and J2ME

OUTCOMES

- Demonstrate the knowledge of protocols used in mobile computing.
- Gain the knowledge about the technologies in modern world like GSM, CDMA, MANETs, WAP, Bluetooth and J2ME
- Develop projects in mobile computing and also able to deliver a seminar on mobile computing working functionality
- Demonstrate various data delivery techniques through various protocols.
- Demonstrate how the data is transferring through various layers of the network

(Wireless) Medium Access Control - Motivation for a specialized MAC (Hidden and exposed terminals, Near and far terminals), SDMA, FDMA, TDMA, CDMA.

Unit II: Mobile Network Layer - Mobile IP (Goals, assumptions, entities and terminology, IP packet delivery, agent advertisement and discovery, registration, tunneling and encapsulation, optimizations), Dynamic Host Configuration Protocol (DHCP).

Mobile Transport Layer - Traditional TCP, Indirect TCP, Snooping TCP, Mobile TCP, Fast retransmit/fast recovery, Transmission /time-out freezing, Selective retransmission, Transaction oriented TCP.

Unit III: Database Issues - Hoarding techniques, caching invalidation mechanisms, client server computing with adaptation, power-aware and context-aware computing, transactional models, query processing, recovery, and quality of service issues.

Data Dissemination - Communications asymmetry, classification of new data delivery mechanisms, push-based mechanisms, pull-based mechanisms, hybrid mechanisms, selective tuning (indexing) techniques.

Unit IV: Mobile Ad hoc Networks (MANETs) - Overview, Properties of a MANET, spectrum of MANET applications, routing and various routing algorithms, security in MANETs.

Protocols and Tools - Wireless Application Protocol-WAP. (Introduction, protocol architecture, and treatment of protocols of all layers), Bluetooth (User scenarios, physical layer, MAC layer, networking, security, link management) and J2ME.

TEXT BOOKS:

REFERENCE BOOKS:
M.Tech  I Year I Semester(Computer Science & Engineering)  
A Y 2013-14  
Course Code:CSEP1 1413

Course Title:SOFT COMPUTING TECHNIQUES

L T P C
3 1 0 4

Objectives:
This course is designed for first year M.Tech students. The course is intended to make the students understand concepts about Soft Computing and its application in various field.

The main objective of the course is to:

- know Soft Computing basics and its branches
- understand the basic implementation details on Artificial Neural Networks
- understand fuzzy logic and its application in ANN.
- introduction of Support vector machine and its application
- elaborate discussion on applications of Soft Computing

OUTCOMES:

- differentiate between Soft Computing and Hard computing.
- understand its branches Artificial Neural Networks, Fuzzy Logic, and Support Vector machine
- understand various applications of soft computing.
- judge less complexity by using various soft computing methods
UNIT I:
Basic elements of soft Computing – Introduction to soft computing, Fuzzy logic, Neural Networks and Evolutionary Computing, Approximations of Multivariate functions, Non – linear Error surface and optimization.

Artificial Neural Networks- Introduction, Basic models of ANN, important terminologies, Basic Learning Laws, Supervised Learning Networks, Perceptron Networks, Adaptive Linear Neuron, Backpropagation Network. Radial basis function network and Hopfield Networks.

UNIT II:


UNIT-III:

Genetic Algorithm- Introduction, Traditional Optimization and search techniques, Search space, Operators: Encoding, Selection, Crossover and Mutation. Stopping Condition of GA.

UNIT IV:

Applications of Soft Computing - A fusion Approach of Multispectral Images with SAR Image for flood area analysis, Optimization of TSP using GA Approach and GA-Fuzzy system for Control of flexible Robots.

TEXT BOOKS
2. V. Kecman, “Learning and Soft computing”, Pearson Education, India

REFERENCE BOOKS:
Course Code: CSEP1 1414

Course Title: WEB TECHNOLOGIES

L  T  P  C
3  1  0  4

Objectives
- Understand markup languages
- Understand how to integrate with PHP
- To Understand the basic web application development
- To understand how to design web applications using MVC
- Various ways to collaborate web services

Outcomes
- Develop markup language and scripting language programs for web applications
- Develop advanced Java programs
- Design web applications using PHP
- Implement various challenges associated with web development
- Demonstrate different architectures and web services
UNIT I:
HTML - Common tags, linking, Lists, Tables, images, forms, Frames, Cascading Style sheets.
JavaScript - Introduction to Java Scripts, control structures, functions, objects, event handling, Dynamic HTML.

UNIT II:
XML - Introduction, importance, Document type definition, XML Schemas, XML parsers: DOM and SAX, namespaces, Style sheets: XSL and CSS.
Java Beans - Introduction to Java Beans, Advantages of Java Beans, BDK, Introspection, jar file, Bound properties, Constrained properties, Persistence, Customizer, Bean Info Interface, Java Beans API, Introduction to EJB’s.

UNIT III:
JSP Application Development: Generating Dynamic Content, The JSP 2.0 API, Action elements- standard actions and custom actions, using JSTL and custom tag libraries, EL Expressions, sharing data between JSP pages, requests and users, passing control and data between pages, Deploying JAVA Beans in a JSP Page, Error handling and debugging.

UNIT IV:
Database Connectivity - Introduction to JDBC, drivers, Database Programming, java.sql and javax.sql packages, Accessing a Database from a JSP Page, Introduction to struts framework.
PHP - Creating PHP script, running PHP script, working with variables and constants, datatypes, operators, conditional statements, arrays, functions, working with forms, database connectivity.

TEXT BOOKS:
1. Internet and World Wide Web – How to program, Dietel and Dietel, Pearson Education.
2. The complete Reference Java 2, Fifth Edition, Herbert Schildt, TMH.

REFERENCE BOOKS:
Course Title: COMPILER DESIGN (Elective – 3)  
Course Code: CSEP1 1415  

Objectives:

- To learn basic structure of compilers, basic concepts and terminology in programming language.
- To learn context free grammars, recursive descent parsing, predictive parsers on top-down parsing.
- To learn SLR, CLR, LALR parsers on bottom-up parsing.
- To learn intermediate code forms and syntax directed translation.
- To learn symbol table format, hashing and storage allocations.
- To learn different types of optimization of code.
- To learn flow graphs, live variables and global optimization.
- To learn different object code generation techniques.

Outcomes:

- Understand the lexical analysis, parsing and code generation and optimization phases of compilation, and design a compiler for a concise programming language.
- Design and implementation of the following four stages of compilation for a subset of a modern imperative programming language: lexical analysis, parsing, code generation and code optimization.
- The learning outcomes are assessed through assignments, mid-semester and final exam.
- Assessment of students understanding of the subject may be done through the quality of the programs that they write in conducted laboratory.
UNIT I:
Overview of language processing - preprocessors, compiler, assembler, interpreters, linkers & loaders,
Overview of Compilation: Phases of Compilation, pass and Phases of translation, bootstrapping.
Lexical Analysis, Role of Lexical Analysis - Lexical Analysis Vs. Parsing, Token, patterns and Lexemes,
Lexical Errors, Regular Grammar and regular expression for common programming language features,Transition diagram for recognition of tokens, Reserved words and identifiers, Examples., LEX-lexical analyzer generator.

UNIT II:
Syntax analysis - Context free grammars, Role of a parser, Top down parsing, Backtracking, LL (1),
recursive descent parsing, Predictive parsing, Preprocessing steps required for predictive parsing.
Bottom up parsing - Shift Reduce parsing, Simple LR parsers, Why LR parsers, Construction of CLR,
LALR parsing, Error recovery in parsing.
Semantic analysis - Intermediate code, three address code, quadruples, triples Intermediate forms of source
Programs, Syntax directed translation, Attributed grammars, abstract syntax tree, evolution and flow of
attribute in syntax tree, Conversion of popular Programming languages language Constructs into Intermediate
code forms.

UNIT III:
Symbol Tables - Symbol table format, use of symbol tables, Symbol attribute and management, Run-time
environment: Procedure activation, parameter passing, value return, memory allocation, and scope.
Code optimization - Consideration for Optimization, Machine independent code optimization- Common
sub expression elimination, constant folding, copy propagation, dead code elimination, strength reduction,
loop optimization, procedure inclining.
Machine dependent code optimization: Peephole optimization, register allocation

UNIT IV:
Data flow analysis - Dataflow Analysis, Intermediate representation for flow analysis, Various
dataflow analyses, Loop Optimizations: Dominators, Loop-invariant computations, Array bounds checks, Loop unrolling.
Code Generation - Processing the intermediate Code- Interpretation, Code generation, Simple code
generation, code generation for basic blocks, BURS Code generation and dynamic programming,
Register allocation by graph coloring, Evaluation of code generation techniques Preprocessing the
intermediate code, post processing the target code, machine code generation.

TEXT BOOKS :
1. Principles of compiler design, A.V. Aho . J.D Ullman; Pearson Education

REFERENCE BOOKS :
3. lex & yacc, John R. Levine, Tony Mason, Doug Brown, O’reilly
Objectives:
This course is designed for first year M.Tech students. The course is intended to make the students to understand the basic concepts of distributed system with various system models and Interprocess Communication and Communication between Distributed Objects with their operating system support, support of distributed File and various Transactions & Replications.

- Understand the distributed systems with examples
- Understand Architectural models and other Fundamental models and Internet Protocols and IPC and various types of communications.
- Acquire knowledge of Communication between Distributed Objects and RMI implementation
- Acquire knowledge of Operating System Layer, Processes and Threads and Distributed File Systems.
- Understand distributed Mutual Exclusion and elections and distributed Dead Locks.
- Understand Transaction Recovery and Replication

Course Outcomes:

- Gain the knowledge about the technologies in distributed environment.
- Develop applications in the area of distributed system.
- Demonstrate various data delivery models (peer to peer system, Napster)
- Demonstrate how Mutual Exclusion and elections and distributed Dead Locks are used.


UNIT II: Interprocess Communication – Introduction, The API for the Internet Protocols – The Characteristics of Interprocess communication, Sockets, UDP datagram Communication, TCP stream Communication; External Data Representation and Marshalling; Client Server Communication; Group Communication – IP Multicast – an implementation of group communication, Reliability and Ordering of Multicast.

Distributed Objects and Remote Invocation – Introduction, Communication between Distributed Objects – Object Model, Distributed Object Modal, Design Issues for RMI, Implementation of RMI, Distributed Garbage Collection; Remote Procedure Call, Events and Notifications, Case Study: JAVARMI


UNIT IV: Coordination and Agreement – Introduction, distributed Mutual Exclusion, Election, Multicast Communication.

Transactions & Replications – Introduction, system Model and Group Communication, Concurrency Control in distributed Transactions, distributed Dead Locks, Transaction Recovery; Replication – Introduction, Passive (Primary) Replication, active Replication.

TEXT BOOKS:

REFERENCES:
1. Distributed Operating Systems,PradeepK.Sinha,PHI.
Objectives:
- The main objective of this course is: To understand open sources and Application of open sources
- Learn open source operating system LINUX and its environment and development
- Learn open source database i.e MYSQL record selection technology, sorting query results, generating summary, generating metadata.
- Study the open source programming languages i.e PHP programming environment, File handling and data storage and database connectivity to MYSQL
- Study the open source programming language PYTHON’s syntax and style, objects, Lists and tuples, dictionaries, conditionals and loops, File handling
- Study the open source programming language PERL’s parsing rules, statements and control structures, packages and modules, files and Data manipulation

Outcomes:
At the end of the course students are able to:
- Involve open source software groups and develop new features for existing software’s
- Design a Dynamic WEB application using open sources with database Connectivity
- Develop real time projects using PHP and database interfacing
- Develop real time projects using PYTHON and PERL database interfacing
UNIT I:
INTRODUCTION—Introduction to open sources—Need of Open Sources—Advantages of Open Sources—Application of Open Sources. Open Sources Operating systems: LINUX Introduction—General Overview—Kernel Mode and user mode.


UNIT II:


UNIT III:
PHP and SQL database—PHP and LDAP—PHP Connectivity—Sending and receiving E-mails—Debugging and error handling—Security—Templates.

PYTHON—Syntax and Style—Python Objects—Numbers—Sequences—Strings—Lists and Tuples—Dictionaries—Conditionals and Loops.

UNIT IV:
Files—Input and output—Errors and Exceptions—Functions—Modules—Classes and OOP—Execution Environment.

PERL—Perl backgrounder—Perl overview—Perl parsing rules—Variables and Data—0 Statements and Control structures—Subroutines, Packages, and Modules—Working with Files—Data Manipulation.

TEXT BOOKS:

REFERENCE BOOKS:
Course Title: CLOUD COMPUTING (Elective - 4)  

L  T  P  C
3  1  0  4

Objective:

• Understand cloud computing basics
• To understand the basic implementation details about cloud services on web
• To understand how cloud computing can be applied to family and to corporations
• Various ways to collaborate cloud services online
• Cloud computing software security fundamentals and its secured architecture.

OUTCOMES:

• Understand the use of cloud computing and its services at individual level, family level and at corporate level
• able to understand how to collaborate various cloud services online
• Understand cloud computing software security fundamentals
• Understand various risks, threats and challenges associated with cloud computing for an organization
• Able to understand the influence of cloud computing architecture on its security.
UNIT I:

UNIT II:
Cloud Computing for Everyone - Centralizing Email Communications, Collaborating on Schedules, Collaborating on To-Do Lists, Collaborating Contact Lists, Cloud Computing for the Community, Collaborating on Group Projects and Events, Cloud Computing for the Corporation.

UNIT III:
Other Ways to Collaborate Online - Collaborating via Web-Based Communication Tools, Evaluating Web Mail Services, Evaluating Web Conference Tools, Collaborating via Social Networks and Groupware, Collaborating via Blogs and Wikis.

UNIT IV:

TEXT BOOKS:

REFERENCES:
Objective:
Students undergoing this course are expected to:
- learn the fundamentals of digital image processing (transforming and filtering)
- Understand statistical and spatial image compression techniques.
- Know about edge detection and segmentation techniques used for analyzing the images.
- Understand and analyze methods for pattern classification and clustering methodologies.
- Learn common feature extraction methods for pattern recognition

Outcomes:
At the end of the course students are expected to:
- Apply image processing techniques in both the spatial and frequency (Fourier) domains
- Able to implement image processing programs using C++, MATLAB and java languages
- Design systems and algorithms for pattern recognition
- Implement typical pattern recognition algorithms in MATLAB
- Apply well-known algorithms to pilot problems
UNIT I:

**Introduction** - Origins of digital image processing, Applications of digital image processing, Fundamental steps of image processing, components of an image processing system. The image model and image acquisition, sampling and quantization, relationship between pixels, distance functions, scanner.

**Transformation and Filtering** - Statistical and spatial operations, Intensity functions transformations, histogram processing, smoothing & sharpening, spatial filters Frequency domain filters, homomorphic filtering, image filtering & restoration. Inverse and weiner filtering, FIR weiner filter, Filtering using image transforms, smoothing splines and interpolation.

UNIT II:

**Image compression** - Types and requirements, statistical compression, spatial compression, contour coding, quantizing compression, image data compression-predictive technique, pixel coding, transfer coding theory, lossy and lossless predictive type coding, Digital Image Water marking.

**Morphology** - Morphological and other area operations, basic morphological operations, opening and closing operations, dilation erosion, Hit or Miss transform, morphological algorithms, extension to grey scale images.

UNIT III:

**Segmentation and Edge Detection** - Detection of discontinuities, edge linking and boundary detection, thresholding, region based segmentation, segmentation by morphological watersheds, use of motion in segmentation.

**Representation and Description** - Chain codes, Ploygonal approximation, Signature Boundary Segments, Skeltons, Boundary Descriptors, Regional Descriptors, Relational Descriptors, Principal components for Description, Relational Descriptors

UNIT IV:

**Pattern Recognition Fundamentals** - Basic Concepts of pattern recognition, Fundamental problems in pattern recognition system, design concepts and methodologies, example of automatic pattern recognition systems, a simple automatic pattern recognition model.

**Pattern classification** - Pattern classification by distance function: Measures of similarity, Clustering criteria, K-means algorithm, Pattern classification by likelihood function: Pattern classification as a Statistical decision problem, Bayes classifier for normal patterns.

**TEXT BOOKS :**
1. Digital Image Processing, 3/e,,Rafael C. Gonzalez, Richard E. Woods, PEA.

**REFERENCE BOOKS :**
6. Pattern Recognition, R.Shinghal,Oxford University Press.
Objective:

- To understand the Today’s world wide web (i.e. www) and The Next Generation Web, Development of semantic web, Semantic Web Road Map, Logic on the semantic Web.
- To learn Electronic Sources for Network Analysis, Knowledge Representation for the Semantic Web.
- To learn Ontological representation of social individuals, social relationships, Aggregating and reasoning with social network data.
- To Study Evaluation of web-based social network extraction i.e. optimizing goodness of fit, comparison across methods and networks, predicting the goodness of fit, evaluation through analysis.
- To learn Semantic-based Social Network Analysis in the sciences Methodology.

Outcomes:

At the end of the course students are able to:

- Involve developing and managing social networks.
- Apply the semantic web techniques for managing the data effectively.
- Develop tools using ontology Engineering.
UNIT I:
Social Network Analysis - What is networks analysis?, Development of Social Networks Analysis, Key concepts and measures in network analysis.

UNIT II:

UNIT III:
Modeling and aggregating social network data - State-of-art in network data representation, Ontological representation of social individuals, Ontological representation of social relationships, Aggregating and reasoning with social network data.
Developing social-semantic applications - Building Semantic Web Applications with social network features, Flink: the social networks of the Semantic Web community, Open academia: distributed. Semantic based publication management

UNIT IV:
Evaluation of web-based social network extraction - Differences between survey methods and electronics data extraction, context of the empirical study, data collection, preparing the data, optimizing goodness of fit, comparison across methods and networks, Predicting the goodness of fit, evaluation through analysis.
Semantic-based Social Network Analysis in the sciences Methodology - Data acquisition, Representation, storage and reasoning, Visualization and Analysis, Results – Descriptive analysis, Structural and cognitive effects on scientific performance.

TEXT BOOK

REFERENCES:
Course Title: WEB TECHNOLOGIES LAB

Objectives:
- Understand html tags
- Understand how to integrate with PHP & PHP databases
- To Understand the basic web application development with J2EE technology
- To understand how to design web applications using MVC
- Various ways to collaborate web services

Outcomes:
- Develop html programs for web applications
- Develop advanced Java programs like javabean, servelet and jsp
- Design web applications using PHP and connects to database
- Implement web pages design with javascript language
- Implement web applications with tomcat server

1. Design the following static web pages required for an online book store web site.
   a) Home Page  b) Login Page  c) Catalogue page

2. Design the following static web pages required for an online book store web site.
   d) Cart Page  e) Registration Page

3. VALIDATION: Write JavaScript to validate the various fields of the above registration page.

4. Design a web page using CSS (Cascading Style Sheets) which includes the following:
   a) use different font styles
   b) set background images
   c) control the repetition of images
   d) define styles for links
   e) work with layers
   f) add customized cursor

5. Write an XML file which will display the Book information which includes the following:
   1) Title of the book  2) Author Name  3) ISBN number
   4) Publisher name  5) Edition  6) Price
6 VISUAL BEANS:
Create a simple visual bean with a area filled with a color.

7 1) Install TOMCAT web server and APACHE.
2) Access the above developed static web pages for books web site, using these servers by putting the web pages developed in week-1 and week-2 in the document root.

8 User Authentication :
Assume four users user1,user2,user3 and user4 having the passwords pwd1,pwd2,pwd3 and pwd4 respectively. Write a servelet for doing the following.
1. Create a Cookie and add these four user id’s and passwords to this Cookie.
2. Read the user id and passwords entered in the Login form (week1) and authenticate with the values (user id and passwords) available in the cookies.

9 Install a database(Mysql or Oracle). Create a table which should contain at least the following fields: name, password, email-id, phone number(these should hold the data from the registration form). Practice ‘JDBC’ connectivity.
Write a java program/servlet/JSP to connect to that database and extract data from the tables and display them. Experiment with various SQL queries.
Insert the details of the users who register with the web site, whenever a new user clicks the submit button in the registration page (week2).

10 Write a JSP which does the following job: Insert the details of the 3 or 4 users who register with the web site (week9) by using registration form. Authenticate the user when he submits the login form using the user name and password from the database (similar to week8 instead of cookies).

11 Create tables in the database which contain the details of items (books in our case like Book name, Price, Quantity, Amount) of each category. Modify your catalogue page (week 2) in such a way that you should connect to the database and extract data from the tables and display them in the catalogue page using JDBC.

12 HTTP is a stateless protocol. Session is required to maintain the state. The user may add some items to cart from the catalog page. He can check the cart page for the selected items. He may visit the catalogue again and select some more items. Here our interest is the selected items should be added to the old cart rather than a new cart. Multiple users can do the same thing at a time(i.e., from different systems in the LAN using the ip-address instead of localhost). This can be achieved through the use of sessions. Every user will have his own session which will be created after his successful login to the website. When the user logs out his session should get invalidated (by using the method session.invalidate() ).

Modify your catalogue and cart JSP pages to achieve the above mentioned functionality using sessions.