Course Structure & Syllabus

Master in Science (Computer Science)

(Effective from the academic Session 2020-2021)

Department of Computer Science & Engineering and Applications
Sambalpur University Institute of Information Technology (SUIIT)
Sambalpur University, Jyoti Vihar-768019, Burla

Programme Outcome

PO-1	**Critical Thinking:** Take informed actions after identifying the assumptions that frame our thinking and actions
PO-2	**Effective Communication:** Will be able to speak, read, write and listen clearly in person and through electronic media in English and in one Indian Language
PO-3	**Social Interaction (Interpersonal Relation):** Elicit views of others, mediate disagreements and prepared to work in team
PO-4	**Entrepreneurship Capability:** Demonstrate qualities to be prepared to become an entrepreneurship
PO-5	**Ethics:** Recognize different value systems including your own, understand the moral dimensions and accept responsibility for them
PO-6	**Environment and Sustainability:** Understand the issues of environmental contexts and sustainable development
PO-7	**Life-Long Learning:** Acquire the ability to engage in independent and life-long learning in the context of socio-technological changes

Syllabus Structure M.Sc. Computer Science

<table>
<tr><td colspan="8"><b>Semester – I</b></td></tr>
<tr><td><b>Code</b></td><td colspan="2"><b>Course Title</b></td><td><b>Category</b></td><td><b>L</b></td><td><b>P</b></td><td><b>T</b></td><td><b>Credits</b></td></tr>
<tr><td><b>CS 511</b></td><td colspan="2">Mathematical Foundations of Computer Science</td><td>Foundation</td><td>4</td><td>0</td><td>0</td><td>4</td></tr>
<tr><td><b>CS 512</b></td><td colspan="2">Programming in C</td><td>Foundation</td><td>4</td><td>0</td><td>0</td><td>3</td></tr>
<tr><td><b>CS 512</b></td><td colspan="2">Computer Systems Architecture</td><td>Core</td><td>4</td><td>0</td><td>0</td><td>4</td></tr>
<tr><td><b>CS 513</b></td><td colspan="2">Database Management Systems</td><td>Core</td><td>3</td><td>0</td><td>1</td><td>3</td></tr>
<tr><td><b>CS 514</b></td><td colspan="2">Data Communication and Computer Networks</td><td>Core</td><td>4</td><td>0</td><td>0</td><td>4</td></tr>
<tr><td><b>CS 515</b></td><td colspan="2">Programming in C Lab.</td><td>Laboratory Course</td><td>0</td><td>3</td><td>0</td><td>2</td></tr>
<tr><td><b>CS 516</b></td><td colspan="2">Database Management Systems Lab.</td><td>Laboratory Course</td><td>0</td><td>3</td><td>0</td><td>2</td></tr>
<tr><td colspan="7"><b>Total Credit:</b></td><td><b>22</b></td></tr>
<tr><td colspan="8"><b>Semester – II</b></td></tr>
<tr><td><b>Code</b></td><td colspan="2"><b>Course Title</b></td><td><b>Category</b></td><td><b>L</b></td><td><b>P</b></td><td><b>T</b></td><td><b>Credits</b></td></tr>
<tr><td><b>CS 521</b></td><td colspan="2">Object Oriented Programming using Java</td><td>Foundation</td><td>3</td><td>0</td><td>0</td><td>3</td></tr>
<tr><td><b>CS 522</b></td><td colspan="2">Theory of Computations</td><td>Core</td><td>3</td><td>0</td><td>0</td><td>4</td></tr>
<tr><td><b>CS 523</b></td><td colspan="2">Software Engineering</td><td>Core</td><td>3</td><td>0</td><td>0</td><td>3</td></tr>
<tr><td><b>CS 524</b></td><td colspan="2">Data Structures</td><td>Core</td><td>3</td><td>0</td><td>1</td><td>3</td></tr>
<tr><td><b>CS 525</b></td><td colspan="2">Operating Systems</td><td>Core</td><td>3</td><td>0</td><td>0</td><td>3</td></tr>
<tr><td><b>CS 526</b></td><td colspan="2">Object Oriented Programming using Java Lab.</td><td>Laboratory Course</td><td>0</td><td>3</td><td>0</td><td>2</td></tr>
<tr><td><b>CS 527</b></td><td colspan="2">Data Structures Lab.</td><td>Laboratory Course</td><td>0</td><td>3</td><td>0</td><td>2</td></tr>
<tr><td colspan="7"><b>Total Credit:</b></td><td><b>20</b></td></tr>
<tr><td colspan="8"><b>Semester – III</b></td></tr>
<tr><td><b>Code</b></td><td colspan="2"><b>Course Title</b></td><td><b>Category</b></td><td><b>L</b></td><td><b>P</b></td><td><b>T</b></td><td><b>Credits</b></td></tr>
<tr><td><b>CS 531</b></td><td colspan="2">Compiler Design</td><td>Core</td><td>3</td><td>0</td><td>1</td><td>3</td></tr>
<tr><td><b>CS 532</b></td><td colspan="2">Design and Analysis of Algorithms</td><td>Core</td><td>4</td><td>0</td><td>0</td><td>4</td></tr>
<tr><td><b>CS 533</b></td><td colspan="2">Computer Graphics</td><td>Core</td><td>4</td><td>0</td><td>0</td><td>3</td></tr>
<tr><td><b>CS 534</b></td><td colspan="2">Web Technology</td><td>Core</td><td>4</td><td>0</td><td>0</td><td>3</td></tr>
<tr><td rowspan="6"><b>XX XXXX</b></td><td colspan="2">Elective-I</td><td>Prog. Elect.</td><td rowspan="6">4</td><td rowspan="6">0</td><td rowspan="6">0</td><td rowspan="6">3</td></tr>
<tr><td><b>CS 53E1</b></td><td colspan="2">Mobile Computing</td></tr>
<tr><td><b>CS 53E2</b></td><td colspan="2">Information Retrieval Systems</td></tr>
<tr><td><b>CS 53E3</b></td><td colspan="2">Optimization Techniques</td></tr>
<tr><td><b>CS 53E4</b></td><td colspan="2">Management Information Systems</td></tr>
<tr><td><b>CS53E5</b></td><td colspan="2">Computer Based Numerical and Statistical Methods</td></tr>
<tr><td><b>CS 535</b></td><td colspan="2">Web Technology Lab.</td><td>Laboratory Course</td><td>0</td><td>3</td><td>0</td><td>2</td></tr>
<tr><td><b>CS 536</b></td><td colspan="2">Python Programming Lab.</td><td>Laboratory Course</td><td>0</td><td>3</td><td>0</td><td>2</td></tr>
<tr><td colspan="7"><b>Total Credit:</b></td><td><b>20</b></td></tr>
<tr><td colspan="8"><b>Semester – IV</b></td></tr>
<tr><td><b>Code</b></td><td colspan="2"><b>Course Title</b></td><td><b>Category</b></td><td><b>L</b></td><td><b>P</b></td><td><b>T</b></td><td><b>Credits</b></td></tr>
<tr><td><b>CS 541</b></td><td colspan="2">Data Warehousing and Data Mining</td><td>Core Course</td><td>4</td><td>0</td><td>0</td><td>4</td></tr>
<tr><td><b>CS 542</b></td><td colspan="2">Artificial Intelligence</td><td>Core Course</td><td>4</td><td>0</td><td>0</td><td>4</td></tr>
<tr><td rowspan="6"><b>XX XXXX</b></td><td colspan="2">Elective-II</td><td>Prog. Elect.</td><td rowspan="6">3</td><td rowspan="6">0</td><td rowspan="6">0</td><td rowspan="6">3</td></tr>
<tr><td><b>CS 54E1</b></td><td colspan="2">Wireless Sensor Networks</td></tr>
<tr><td><b>CS 54E2</b></td><td colspan="2">Cloud Computing</td></tr>
<tr><td><b>CS 54E3</b></td><td colspan="2">Machine Learning</td></tr>
<tr><td><b>CS 54E4</b></td><td colspan="2">Introduction to Big Data Analytics</td></tr>
<tr><td><b>CS 54E5</b></td><td colspan="2">Information & Cyber Security</td></tr>
<tr><td><b>CS 543</b></td><td colspan="2">Project</td><td>Project Work</td><td>-</td><td>-</td><td>-</td><td>8</td></tr>
<tr><td><b>CS 544</b></td><td colspan="2">Seminar</td><td>Tech. Seminar</td><td>-</td><td>-</td><td>-</td><td>2</td></tr>
<tr><td colspan="7"><b>Total Credit:</b></td><td><b>21</b></td></tr>
</table>

SEMESTER WISE CREDIT DISTRIBUTION					
Semester	**I**	**II**	**III**	**IV**	**TOTAL**
Total Credit	**22**	**20**	**20**	**21**	**83**

Semester – I						
Code	**Course Title**	**Category**	**L**	**P**	**T**	**Credits**
CS 511	Mathematical Foundations of Computer Science	Foundation	4	0	0	4
CS 512	Programming in C	Foundation	4	0	0	3
CS 513	Computer Systems Architecture	Core	4	0	0	4
CS 514	Database Management Systems	Core	3	0	1	3
CS 515	Data Communication and Computer Networks	Core	4	0	0	4
CS 516	Programming in C Lab.	Laboratory Course	0	3	0	2
CS 517	Database Management Systems Lab.	Laboratory Course	0	3	0	2
Total Credit:						**22**

MATHEMATICAL FOUNDATIONS OF COMPUTER SCIENCE

Course Code	**CS 511**	**L-P-T-Cr.:**	4	0	0	4	**Semester:**	**I**

Category: Foundation

Prerequisite: NA

Learning Objective:

- To introduce the concepts of mathematical logic.
- To introduce the concepts of sets, relations, and functions.
- To perform the operations associated with sets, functions, and relations.
- To relate practical examples to the appropriate set, function, or relation model, and interpret the associated operations and terminology in context.
- To introduce generating functions and recurrence relations.
- To use Graph Theory for solving problems.

CO-1	Remember and understand the basic concepts/Principles of Mathematics Foundation
CO-2	Analyze the Various Concepts to understand them through case studies
CO-3	Apply the knowledge in understanding practical problems
CO-4	Execute/Create the Project or field assignment as per the knowledge gained in the course

UNIT – I: LOGIC (10 hours)

Propositional logic, predicate calculus, Theory of inference, modus pones, resolution, clause system, (CNF, DNF, PDNF, PCNF, NNF), completeness, soundness, network applications. Fuzzy logic: fuzzy relation, pattern classification, fuzzy analysis, distance between fuzzy sets, area perimeter, height, width of fuzzy subsets.

UNIT – II: NUMBER SYSTEM (10 hours)

Introduction to number system, residue arithmetic, application of residue arithmetic to computers, binary operation of residue numbers, Fermat's theorem, Euler Theorem, Text Coding, RSA.

UNIT – III: COMBINATORICS GRAPH THEORY (10 hours)

Basic concept of graph theory, directed and undirected graph, matrix representation, graph manipulation, Dijkstra algorithm, all pair shortest path problem, Centre of graph, Tree and Text coding. Counting principle, Pigeon- hole principle, principle of inclusion and exclusion, recurrence relations, Method of proofs.

UNIT – IV: ALGEBRAIC STRUCTURE (10 hours)

Semi groups and monodies, Group, subgroup, homomorphism, co-sets, normal subgroup, Lagrange's theorem, algebraic system of two binary operation, lattices, partial order set, properties of lattices, special lattices, Boolean algebra, Boolean function and simplification, group codes, parity check, single error correcting code.

TEXT BOOKS:

1. C. L. Liu, Elements of Discrete Mathematics, McGraw-Hill.
2. K. H. Rosen, Discrete Mathematics and applications, Tata Mc Graw Hill

REFERENCE BOOKS:

1. J .L. Mott, A. Kandel, T.P .Baker, Discrete Mathematics for Computer Scientists and Mathematicians, second edition 1986, Prentice Hall of India.
2. R. Grimaldi and B V Ramana, Discrete and combinatorial mathematics: An applied introduction, Pearson education

PROGRAMMING IN C

Course Code	**CS 512**	**L-P-T-Cr.:**	4	0	0	3	**Semester:**	**I**
Category:	Foundation Course							
Prerequisite:	NA							

Learning Objective:

- To understand the various steps in Program development.
- To understand the basic concepts in C Programming Language.
- To learn how to write modular and readable C Programs
- To learn to write programs (using structured programming approach) in C to solve problems.
- To make the student understand simple sorting and searching methods.

CO-1	Remember and understand the basic concepts/Principles of Programming in C
CO-2	Analyze the Various Concepts to understand them through case studies
CO-3	Apply the knowledge in understanding practical problems
CO-4	Execute/Create the Project or field assignment as per the knowledge gained in the course

UNIT – I: **(10 hours)**

Computer Fundamentals and Introduction to C: Role of computer and programming languages, compiler, interpreter, loader and linker, classification of programming languages, structured programming, concepts, algorithms and flowcharts.

Basics of C: Developing programs in C, a simple C program, structure of a C program, concept of a variable, data types in C, variables, program statement, declaration. All tokens, literals, operators and expressions, type conversions in C. Non-formatted input and output, formatted input and output.

UNIT – II: **(10 hours)**

Control Statements: Introduction, conditional execution (if, if-else, nested if), selection (switch), unconditional types (break, continue, goto).

Loops: Iteration and repetitive execution (for, while, do-while) nested loops.

Arrays and Strings: Introduction, definition, one dimensional array, two dimensional arrays, accessing elements and storing elements.

String: Introduction, C characters and strings, character handling library, string conversion functions, standard input output library functions, comparison functions of string handling, string manipulation functions, search and memory functions of string handling library.

UNIT – III: **(10 hours)**

Functions: Designing structured programs, functions, basics, parameter passing, call by value and call by reference mechanism to working with functions-example programs

Storage Class: extern, auto, register, static, scope rules, block structure, user defined functions, standard library functions, recursive functions and passing arrays to functions. Dynamic memory allocation**.**

UNIT – IV: **(10 hours)**

Structures: Declaring structures and structure variables, accessing members of a structure, arrays of structures, arrays within a structure. Structures and functions, pointers to structures.

Union: Declaring union and its members, accessing and initializing members of a union, structure versus union.

Input and output: concept of a file, opening a file, closing a file; Working with text files, reading from and writing into text files, error handling and C program examples.

TEXT BOOKS:

1. Pradip Dey and Manas Ghosh, Programming in C, 2/e, Oxford University Press, 2013.
2. A Structured Programming Approach Using C, B.A. Forouzan and R.F. Gilberg, Third Edition, Cengage Learning.

REFERENCES BOOKS:

1. R.S.Bichkar, Programming in C, University Press (India) Pvt. Ltd., 2012.
2. K.R.Venugopal and S.K.Prasad, Mastering C, McGraw Hill, 2009.
3. E.Balaguruswamy, Programming in ANSI C, 6/e, McGraw Hill.
4. Ashok N. Kamthane, Programming with ANSI and Turbo C, Pearson Education, India.

COMPUTER SYSTEMS ARCHITECTURE

Course Code	**CS 513**	**L-P-T-Cr.:**	4	0	0	4	**Semester:**	I
Category:	Core							
Prerequisite:	**Digital Electronic Circuit**							

Learning Objective:

- To understand how computers are constructed out of set of functional units.
- To understand concrete representation of data at the machine level.
- To understand how these functional units operate, interact and communicate.
- Understand the design of processors, the structure and operation of memory, pipelining, system integration and peripherals.
- To understand the system interconnection and the different I/O techniques.

CO-1	Remember and understand the basic concepts/Principles of Computer Systems Architecture
CO-2	Analyze the Various Concepts to understand them through case studies
CO-3	Apply the knowledge in understanding practical problems
CO-4	Execute/Create the Project or field assignment as per the knowledge gained in the course

UNIT – I: **(12 hours)**

Digital logic Circuits: Digital Computers, Logic gates, Boolean algebra, Map Simplification, Combinational, Flip-flops and Sequential Circuits. decoders, Encoders, Multiplexers, Half and Full adders, Shift registers, Binary counter and memory unit.

Data Representation: Data types, complements, Fixed point representation, Floating point representation, Gray code and BCD codes.

UNIT – II: **(12 hours)**

Register Transfer and Micro operations: Register transfer language, register transfer, bus and memory transfers, arithmetic micro-operations, logic micro-operations, shift micro-operations, arithmetic logic and shift unit.

Basic Computer Organization and Design: Instruction Codes, computer registers, computer Instructions, timing and control, instruction cycle, memory-references instructions, input-output and interrupt, design of the basic computer, design of accumulator logic.

UNIT – III: **(12 hours)**

Central Processing Unit: General register organization, stack organization, instruction formats, addressing modes, Data transfer and manipulation, Program control.

Computer Arithmetic: Introduction, addition and subtraction, decimal arithmetic unit, booth multiplication algorithm.

UNIT – IV: **(12 hours)**

Input/Output Organization: Peripheral devices, Input-Output Interface, Asynchronous data transfer, Modes of transfer, Priority Interrupt and DMA.

The Memory System: Some basic concepts, Memory Hierarchy, Auxiliary memory, Associative memory, Cache memories, cache memory techniques, Virtual memories.

TEXT BOOKS

1. M.Morris Mano, Computer System Architecture, 3/e, Pearson education, 2008.

REFERENCE BOOKS:

1. Carl Hamacher, ZvonkoVranesic, SafwatZaky, Computer Organization, 5/e, McGraw Hill, 2001.
2. John P. Hayes, Computer Architecture and Organization, 3/e, McGraw Hill, 1998.
3. William Stallings, Computer Organization and Architecture, 6/e, Pearson, PHI, 2012.

DATABASE MANAGEMENT SYSTEMS

Course Code	**CS 514**	**L-P-T-Cr.:**	3	0	1	3	**Semester:**	**I**

Category: Core

Prerequisite: **Basic Knowledge of Computer Programming and data structures**

Learning Objective:

- Classify modern and futuristic database applications based on size and complexity;
- Design a database from understanding an Universe of Discourse, using ER diagrams;
- Map ER model into Relational model and to normalize the relations;
- Create a physical database from a design using DDL statements with appropriate key, domain and referential integrity constraints;
- Analyze different ways of writing a query and justify which is the effective and efficient way; and compare and contrast various indexing strategies in different database systems.

CO-1	Remember and understand the basic concepts/Principles of Database Management Systems
CO-2	Analyze the Various Concepts to understand them through case studies
CO-3	Apply the knowledge in understanding practical problems
CO-4	Execute/Create the Project or field assignment as per the knowledge gained in the course

UNIT – I: INTRODUCTION TO DATABASE SYSTEMS (12 hours)

Data - Database Applications: Evolution of DB & DBMS - Need for data management –
Data models & Database Schema Architecture - components of DBMS - Key issues and challenges in Database Systems

E/R Model: Conceptual data modeling -E/R diagram notation, ER Diagrams - Relational Model - ER to Relational Mapping - Constraints - Keys - Dependencies examples.

UNIT – II: DATABASE LANGUAGE AND DATABASE DESIGN (12 hours)

Introduction to Database Languages: Relational Algebra, Relational Calculus, SQL - Introduction, data definition in SQL, table, key and foreign key definitions, update behaviors. Querying in SQL, notion of aggregation, aggregation functions group by and having clauses, embedded SQL.

Database Design: Introduction to Functional Dependency and Normalization – Concept of functional dependency, First, Second, Third & Fourth Normal Forms - BCNF – Join Dependencies-other Normal forms

UNIT – III: TRANSACTIONS & CONCURRENCY AND PHYSICAL DATABASE DESIGN (12 hours)

Introduction to Transactions: Transaction Systems - ACID Properties - System & Media Recovery - Two Phase Commit Protocol - Recovery with SQL - Need for Concurrency Locking Protocols - Deadlocks & Managing Deadlocks - SQL Support for Concurrency.

Storage Strategies: Indices, B-Trees, Hashing, Indexing.

UNIT – IV: QUERY PROCESSING AND ADVANCES IN DATABASES (12 hours)

Query Processing and Optimization: Query Tree, Evaluation of Relational Algebra Expressions, Query Equivalence, Join strategies, Query Optimization Algorithms.

Introduction to Special Topics: Spatial & Temporal Databases – Data Mining & Warehousing - Data Visualization - Mobile Databases - OODB & XML Databases - Multimedia & Web Databases.

TEXT BOOKS

1. Elmaski & Navathe -Fundamentals of Database Systems, 4th Edition, Pearson Education

REFERENCE BOOKS:

1. Database Systems, Thomas Connolly, Carolyn Begg
2. C.J. Date - An introduction to Database Systems, Pearson Education
3. Avi Silberschatz, Henry F. Korth , S. Sudarshan,Database System Concepts
4. Bipin Desai -An introduction to Database System, Galgotia Publication.
5. Raghu Ramakrishnan, “Database Management Systems”, Third Edition, McGraw Hill, 2002.

Other References: (Web)

1. http://cs.stanford.edu/people/widom/cs346/ioannidis.pdf
2. http://nptel.ac.in/courses/106106093/

DATA COMMUNICATION AND COMPUTER NETWORKS

Course Code	**CS 515**	**L-P-T-Cr.:**	4	0	0	4	**Semester:**	**I**
Category:	Core							
Prerequisite:	**Basics of Computer**							
Learning Objective:	• The objective of the course is to provide an overview of communication network functions and a good foundation for further studies in the subject. It involves understanding and application of design principles and methods for systems development and review of the underlying systems, and communications technologies and significant standardized systems.							

CO-1	Remember and understand the basic concepts/Principles of Data Communication and Computer Networks
CO-2	Analyze the Various Concepts to understand them through case studies
CO-3	Apply the knowledge in understanding practical problems
CO-4	Execute/Create the Project or field assignment as per the knowledge gained in the course

UNIT – I: **(12 hours)**

Introduction: Data Communications, Networks, The Internet, Protocols and Standards, Network Models, Layered Tasks, **The OSI Model:** Layers in the OSI Model, TCP/IP Protocol Suite, Addressing, Physical Layer and Media, Data and Signals, Analog and Digital, Periodic Analog Signals, Digital Signals, Transmission impairment, Data Rate Limits, Performance. Switching, Circuit-Switched Networks, Datagram Networks, Virtual-Circuit Networks.

UNIT – II: **(12 hours)**

Error Detection and Correction: Introduction, Block Coding, Liner Block Codes, Cyclic Codes, Checksum, Data Link Control, Framing, Flow and Error Control, Protocols (ARQ), HDLC, Point-to-Point Protocol (PPP), Multiple Access, Random Access, Aloha, Controlled Access, Channelization, IEEE Standards, Standard Ethernet, Fast Ethernet, Gigabit Ethernet, IEEE 802.11, Bluetooth.

Virtual-Circuit Networks: Frame Relay and ATM, Frame Relay.

UNIT – III: **(12 hours)**

Network Layer: Logical Addressing, IPv4 Addresses, IPv6 Addresses, Network Layer: Internet Protocol, Internetworking, IPv4, IPv6, Transition from IPv4 to IPv6, Network Layer: Address Mapping, Error Reporting and Multicasting, Address Mapping, ICMP, IGMP,

Network Layer: Delivery, Forwarding and Routing, Delivery, Forwarding, Unicast Routing Protocols (RIP, OSPF), Multicast Routing Protocols (BGP).

Transport Layer: Process-Process Delivery: UDP, TCP and SCTP, Process-to-Process Delivery, User Datagram Protocol (UDP), TCP, SCTP.

UNIT – IV: **(12 hours)**

Congestion Control and Quality of Service, Congestion, Congestion Control, Quality Service, Techniques to improve QoS, Integrated Services, Differentiated Services, QoS in Switched Networks.

Application Layer: Domain Name Systems (DNS), Remote Logging, Electronic Mail and File Transfer, Telnet.

WWW and HTTP: Architecture, Web Documents, HTTP, Network Management: SNMP, RTP, RTCP, Voice over Introduction to Network Security and Cryptography.

TEXT BOOKS

1. Data Communications and Networking, Fourth Edition by Behrouz A. Forouzan, TMH.

REFERENCE BOOKS:

1. Computer Networks: A system Approach, Larry L. Peterson, Bruce S Davie
2. Computer Networks, A.S. Tanenbaum,4th Edition, Pearson education.
3. Data and Computer Communication, W. Stallings, Prentice- Hall
4. Kurose, J.F. and Ross, K.W., "Computer Networking: A Top-Down Approach Featuring the Internet", Addison Wesley.

PROGRAMMING IN C LAB.

Course Code **CS 516** **L-P-T-Cr.:** 0 3 0 2 **Semester:** **I**

Category: Laboratory Course

Prerequisite: Understanding of fundamental programming concepts.

Learning Objective:

- To make the student learn a programming language.
- To learn problem solving techniques.
- To teach the student to write programs in C and to solve the problems.

Learning Outcome:

- Read, understand and trace the execution of programs written in C language.
- Write the C code for a given algorithm.
- Implement Programs with pointers and arrays, perform pointer arithmetic, and use the pre-processor.
- Write programs that perform operations using derived data types.

CO-1	Remember and understand the basic concepts/Principles of Programming in C lab
CO-2	Analyze the Various Concepts to understand them through case studies
CO-3	Apply the knowledge in understanding practical problems
CO-4	Execute/Create the Project or field assignment as per the knowledge gained in the course

LIST OF TOPICS FOR WRITING C PROGRAMS:

1. Print statements, variables and simple arithmetic operations, mathematical series…etc.
2. Conditional statements (if, if…else, if…else if…else, switch case statement)
3. Loops: (while(…){…}, do{…}while(…), for(…,…,…){}). Some other experiments related to like printing a pattern on the screen…etc.
4. Arrays: One dimensional, multi directional.
5. Strings
6. Pointers
7. User defined Functions
8. Structures and Unions
9. Files: Various operations on Text, Binary Files
10. Command Line Arguments
11. Sorting and Searching algorithms: Basic searching and sorting techniques on linear array.

The above Lab. exercises to be carried out in 45 Hours (15 Lab. Classes).

DATABASE MANAGEMENT SYSTEMS LAB

Course Code	**CS 517**	**L-P-T-Cr.:**	0	3	0	2	**Semester:**	**I**
Category:	Laboratory Course							
Prerequisite:	NA							

Learning Objective:

- To explain basic database concepts, applications, data models, schemas and instances.
- To demonstrate the use of constraints and relational algebra operations.
- Describe the basics of SQL and construct queries using SQL.
- To emphasize the importance of normalization in databases.
- To facilitate students in Database design
- To familiarize issues of concurrency control and transaction management.

CO-1	Remember and understand the basic concepts/Principles of Database Management System lab
CO-2	Analyze the Various Concepts to understand them through case studies
CO-3	Apply the knowledge in understanding practical problems
CO-4	Execute/Create the Project or field assignment as per the knowledge gained in the course

LIST OF TOPICS FOR EXPERIMENTS:

PART A: SQL:

1. DDL Statements (Create, Alter, Drop)
2. DML Statements (Insert, Update, Delete)
3. SELECT Statement: Information retrieval
4. Use of In-built functions (e.g., aggregate functions like Min, Max, Average… etc., time date functions…)
5. TCL statements (COMMIT, ROLL BACK, CHECK POINT)
6. Views, Sequence, Types (ORDBMS)
7. Security Management Commands (like GRANT and REVOKE)

PART B: PL/ SQL:

1. Un-named block
2. Named Blocks (FUNCTIONS, PROCEDURES)
3. Active Database Concepts (TRIGGERS)
4. PACKAGES

REFERENCE BOOKS:

1. ORACLE PL/SQL by example. Benjamin Rosenzweig, Elena Silvestrova, Pearson Education 3rd Edition
2. ORACLE DATA BASE LOG PL/SQL Programming SCOTT URMAN, Tata Mc- Graw Hill.
3. SQL & PL/SQL for Oracle 10g, Black Book, Dr.P.S. Deshpande.

Semester – II						
Code	**Course Title**	**Category**	**L**	**P**	**T**	**Credits**
CS 521	Object Oriented Programming using Java	Foundation	3	0	0	3
CS 522	Theory of Computations	Core	3	0	0	4
CS 523	Software Engineering	Core	3	0	0	3
CS 524	Data Structures	Core	3	0	1	3
CS 525	Operating Systems	Core	3	0	0	3
CS 526	Object Oriented Programming using Java Lab.	Laboratory Course	0	3	0	2
CS 527	Data Structures Lab.	Laboratory Course	0	3	0	2
MOC528	Google Cloud Computing Foundations (MOOCs-1)	MOOC	3	0	0	3
Total Credit:						**23**

OBJECT-ORIENTED PROGRAMMING USING JAVA

Course Code	**CS 521**	**L-P-T-Cr.:**	3	0	0	3	**Semester:**	**II**
Category:	Foundation							
Prerequisite:	Programming in C, Object Oriented Paradigm							

Learning Objective:

- Understand the basic object-oriented programming concepts and apply them in problem solving.
- Illustrate inheritance concepts for reusing the program.
- Demonstrate on the multi-tasking by using multiple threads.
- Develop data-centric applications using JDBC.
- Understand the basics of java console and GUI based programming.

CO-1	Remember and understand the basic concepts/Principles of Object-oriented Programming using JAVA
CO-2	Analyze the Various Concepts to understand them through case studies
CO-3	Apply the knowledge in understanding practical problems
CO-4	Execute/Create the Project or field assignment as per the knowledge gained in the course

UNIT – I: **(10 hours)**

Java Evolution and Environment: Java evolution, overview of java language, java history, features of java, how java differs from C and C++, java and World Wide Web, web browser.

Java Environment: Java Development Kit (JDK), Application Programming Interface (API), java programming structure, java tokens, constants, variables, expressions, decision making statements and looping, java statements, overview of arrays and strings, machine neutral, Java Virtual Machine (JVM), Command Line Arguments.

Arrays and Strings: One-dimensional arrays, creating an array, declaration of arrays, initialization of arrays, two-dimensional arrays, string arrays, string methods, string buffer class, vectors, wrapper classes, Basic I/O Streams: Scanner, buffered reader.

UNIT – II: **(10 hours)**

Classes, Objects and Methods: Introduction, defining a class, creating objects, accessing class members, constructors, method overloading, static members.

Inheritance: Defining a sub-class, sub-class constructor, multi-level variables, final classes and finalize methods, abstract methods and classes, visibility control.

Managing Errors and Exceptions: Introduction, types of errors: compile time and run-time errors, exceptions, types of exceptions, syntax of exception handling code, multiple catch statements, using finally statement, throwing our own exceptions.

UNIT – III: **(12 hours)**

Interfaces, Package and Multi-threaded Programming: Introduction, defining interfaces, extended interfaces, implementing interfaces. Package: Creation, importing a package and user-defined package. Threads: **Introduction to threads:** creating threads, extending the thread class, implementing the 'runnable' interface, life-cycle of a thread, priority of a thread, synchronization, and deadlock.

UNIT – IV: **(12 hours)**

Applet Programming: Introduction, how applets differ from applications, building applet code, applet life cycle, about HTML, designing a web page, passing parameters to applets, getting input from the user.

Graphics Programming: Introduction, abstract window toolkit class hierarchy, frames, event-driven programming, layout managers, panels, canvases, drawing geometric figures.

Introduction to Swings: Introduction to Swings, overview of Swing components: Jbutton, JCheckBox, JRadioButton, JLabel, JTextField, JTextArea, JList.

Introduction to Networking: InetAddress class, socket class, URL class.

TEXT BOOKS

1. Herbert Schildt, The Java Complete References, 9/e, Tata McGraw Hill, 2014.

REFERENCE BOOKS:

1. Y.Daniel Liang, An Introduction to JAVA Programming, Tata McGraw Hill, 2009.
2. Kathy Sierra, Head First java, 2/e, Shroff Publishers, 2012.
3. E. Balaguruswamy, Programming with JAVA, 2/e, Tata McGraw Hill, 2014.

THEORY OF COMPUTATIONS

Course Code	**CS 522**	**L-P-T-Cr.:**	3	0	0	4	**Semester:**	**II**
Category:	Core							
Prerequisite:	**Fundamental of Computer Science and Mathematics**							

Learning Objective:

- To introduce concepts in automata theory and theory of computation.
- To identify different formal language classes and their relationships.
- To design grammars and recognizers for different formal languages.

CO-1	Remember and understand the basic concepts/Principles of Theory of Computations
CO-2	Analyze the Various Concepts to understand them through case studies
CO-3	Apply the knowledge in understanding practical problems
CO-4	Execute/Create the Project or field assignment as per the knowledge gained in the course

UNIT – I: **(10 hours)**

Introduction: Alphabets, Strings and Languages; Automata and Grammars, Deterministic finite Automata (DFA)-Formal Definition, Simplified notation: State transition graph, Transition table, Language of DFA, Nondeterministic finite Automata (NFA), NFA with epsilon transition, Language of NFA, Equivalence of NFA and DFA, Minimization of Finite Automata.

UNIT – II: **(12 hours)**

Regular Expression (RE): Definition, Operators of regular expression and their precedence, Algebraic laws for Regular expressions, Kleen’s Theorem, Regular expression to FA, DFA to Regular expression, Arden Theorem, Non-Regular Languages, Pumping Lemma for regular Languages. Application of Pumping Lemma, Closure properties of Regular Languages, Decision properties of Regular Languages, FA with output: Moore and Mealy machine, Equivalence of Moore and Mealy Machine, Applications and Limitation of FA.

UNIT – III: **(12 hours)**

Context Free Grammar (CFG) and Context Free Languages (CFL): Definition, Examples, Derivation, Derivation trees, Ambiguity in Grammar, Inherent ambiguity, Ambiguous to Unambiguous CFG, Useless symbols, Simplification of CFGs, Normal forms for CFGs: CNF and GNF, Closure proper ties of CFLs, Decision Properties of CFLs: Emptiness, Finiteness and Membership, Pumping lemma for CFLs.

Push Down Automata (PDA): Description and definition, Instantaneous Description, Language of PDA, Acceptance by Final state, Acceptance by empty stack, Deterministic PDA, Equivalence of PDA and CFG, CFG to PDA and PDA to CFG.

UNIT – IV: **(10 hours)**

Turing machines (TM): Basic model, definition and representation, Instantaneous Description, Language acceptance by TM, Variants of Turing Machine, TM as Computer of Integer functions, Universal TM, Church’s Thesis, Recursive and recursively enumerable languages, Halting problem, Introduction to Undecidability, Undecidable problems about TMs. Post correspondence problem (PCP), Modified PCP, Introduction to recursive function theory.

TEXT BOOKS

1. John E. Hopcroft, Rajeev Motwani and Jeffery D. Ullman, Automata Theory, Languages, and Computation (3rd. Edition), Pearson Education, 2008.

REFERENCE BOOKS:

1. H.R.Lewis and C.H.Papadimitriou, Elements of The theory of Computation, Second Edition, Pearson Education/PHI, 2003
2. Michael Sipser, Introduction to the Theory of Computation, Books/Cole Thomson Learning, 2001.
3. J.E. Hopcroft and JD Ullman, Introduction to Automata Theory, Languages, and Computation, Addison-Wesley, 1979.

SOFTWARE ENGINEERING

Course Code	**CS 523**	**L-P-T-Cr.:**	3	0	0	3	**Semester:**	**II**
Category:	Core							
Prerequisite:	**Knowledge on programming and data structure**							
Learning Objective:	• To understand common cycle process life processes. • To understand the basic concepts in Requirement engineering, software design, coding, testing and maintenance • To learn about the role of project management including scheduling, planning, risk management etc. • To have a basic knowledge about software quality, how to ensure good quality software.							

CO-1	Remember and understand the basic concepts/Principles of Software Engineering
CO-2	Analyze the Various Concepts to understand them through case studies
CO-3	Apply the knowledge in understanding practical problems
CO-4	Execute/Create the Project or field assignment as per the knowledge gained in the course

UNIT – I: (12 hours)

Introduction to software and software engineering: various software process modules, capability, maturity, module and KPAs. Project planning, project introduction, team organization, scheduling and management, constructive cost model. Software measures, indicators and metrics, software risk analysis and management.

UNIT – II: (12 hours)

Software requirement analysis and specifications: applicability to small, medium, and large-scale systems. Software design, technical design, objectives of design, design metrics, modularity, module coupling and cohesion, relation between cohesion and coupling; Design strategies: Bottom-up design, top-down design, functional oriented design, object-oriented design; IEEE recommended practice for software design description.

UNIT – III: (14 hours)

Software testing: testability, testing process, structural testing, unit testing and integrated testing, debugging, testing tools, software maintenance, maintenance process, maintenance cost, reverse engineering and reengineering.

UNIT – IV: (10 hours)

Configuration management & Software Quality: Evolution of software quality, assessing and controlling software quality. Software reliability: Hardware vs Software reliability, Reliability metrics. CASE tools and workbenches.

TEXT BOOKS

1. Pressman R., “Software Engineering”, McGraw-Hill.

REFERENCE BOOKS:

1. Sommerville, I., “Software Engineering”, Pearson Education.
2. Dfleeger, S. L., “Software Engineering”, Pearson Education.
3. Rajib Mall, Software Engineering

DATA STRUCTURES

Course Code	**CS 524**	**L-P-T-Cr.:**	3	0	1	3	**Semester:**	**II**
Category:	Core							

Prerequisite: **Concept of C programming, Basics of Computer Architecture (Primary and Secondary storage structure)**

Learning Objective:

- To get clear understanding about the basic data structures and their operations, the concepts of algorithms, basic search and sort algorithms. Student will also gain adequate knowledge to choose appropriate data structure and algorithm to solve a problem

CO-1	Remember and understand the basic concepts/Principles of Data Structures
CO-2	Analyze the Various Concepts to understand them through case studies
CO-3	Apply the knowledge in understanding practical problems
CO-4	Execute/Create the Project or field assignment as per the knowledge gained in the course

UNIT – I: INTRODUCTION, SORTING, AND SEARCHING TECHNIQUES (10 hours)

Introduction: Basic Terminology, Elementary Data Organization, Structure operations, Algorithm Complexity and Time-Space trade-off.

Arrays: Array Definition, Representation and Analysis, Single and Multidimensional Arrays, address calculation, application of arrays, Character String in C, Character string operation, Array as Parameters, Ordered List, Sparse Matrices and Vectors.

Searching: Sequential search, binary search, comparison and analysis.

Sorting: Insertion Sort, Bubble Sort, Quick Sort, Merge Sort, Heap Sort, Sorting on Different Keys, Practical consideration for Internal Sorting.

UNIT – II: LINEAR DATA STRUCTURES (10 hours)

Stacks: Array Representation and Implementation of stack, Operations on Stacks: Push& Pop, Linked Representation of Stack, Operations Associated with Stacks, Application of stack: Conversion of Infix to Prefix and Postfix Expressions, Evaluation of postfix expression using stack, Applications of recursion in problems like 'Tower of Hanoi.

Queues: Array and linked representation and implementation of queues, Operations on Queue: Create, Add, Delete, Full and Empty, Circular queues, D-queues and Priority Queues.

Linked list: Representation and Implementation of Singly Linked Lists, Two-way Header List, Traversing and Searching of Linked List, Overflow and Underflow, Insertion and deletion to/from Linked Lists, Insertion and deletion Algorithms, doubly linked list, Polynomial representation and addition, Generalized linked list.

UNIT – III: NON-LINEAR DATA STRUCTURES (10 hours)

Trees: Basic terminology, Binary Trees, Binary tree representation, algebraic Expressions, Complete Binary Tree, Extended Binary Trees, Array and Linked Representation of Binary trees, Traversing Binary trees, Threaded Binary trees, Traversing Threaded Binary trees, Huffman algorithm.

Binary Search Trees: Binary Search Tree (BST), Insertion and Deletion in BST, Complexity of Search Algorithm, Path Length, AVL Trees, Red-Black tree, B-trees.

Graphs: Terminology & Representations, Graphs & Multi-graphs, Directed Graphs, Sequential representations of Graphs, Adjacency Matrices, Traversal, Connected Component and Spanning Trees, Minimum Cost Spanning Trees.

UNIT – IV: HASHING AND FILE STRUCTURES (10 hours)

Hashing: Hash Table, Hash Functions, Collision Resolution Strategies, Hash Table Implementation

File Organization and Structures: Physical Storage Media File Organization, Organization of records into Blocks, Sequential Files, Indexing and Hashing, Primary indices, Secondary indices, B^+ Tree index Files, B Tree index Files, Indexing and Hashing Comparisons.

TEXT BOOKS

1. Horowitz and Sahani, "Fundamentals of data Structures", Galgotia.
2. Data Management and File Structures, Mary E.S. Loomis, PHI

REFERENCE BOOKS:

1. Richard F. Gilberg & Behrouz A. Forouzan, Data Structures: A pseudo code approach with C, CENGAGE Learning
2. A. M. Tenenbaum, "Data Structures using C & C++", PHI
3. Adam Drozdek, "Data Structures and Algorithms in C++", Thomson Asia Pvt.Ltd.(Singapore)

OPERATING SYSTEMS

Course Code	**CS 525**	**L-P-T-Cr.:**	3	0	0	3	**Semester:**	**II**
Category:	Core							
Prerequisite:	**Computer Programming and Data Structures** **Computer Organization and Architecture**							

Learning Objective:

- Provide an introduction to operating system concepts (i.e., processes, threads, scheduling,
- synchronization, deadlocks, memory management, file and I/O subsystems and protection)
- Introduce the issues to be considered in the design and development of operating system
- Introduce basic Unix commands, system call interface for process management, inter-process communication and I/O in Unix

CO-1	Remember and understand the basic concepts/Principles of Operating Systems.
CO-2	Analyze the Various Concepts to understand them through case studies
CO-3	Apply the knowledge in understanding practical problems
CO-4	Execute/Create the Project or field assignment as per the knowledge gained in the course

UNIT – I: (12 hours)

Operating System Overview: -Introduction, The Need of Operating Systems, Evolution of Operating Systems, Types of Operating Systems, Simple Batch, Multiprogrammed, Time-shared, Personal Computer, Parallel, Distributed Systems, Real-Time Systems, System components, Operating System services, System Calls, Virtual Machines, System Design and Implementation.

Process Management – Process concepts, Life cycle, PCB, Schedulers, Process Scheduling, Threads, Scheduling Levels, CPU Scheduling: Scheduling-Criteria, Algorithms, Algorithm Evaluation, interprocess communication.

UNIT – II: (12 hours)

Concurrency: Process synchronization, The Critical- Section Problem, Peterson's Solution, synchronization Hardware, Semaphores, Classic problems of synchronization, Monitors

Deadlocks: System Model, Deadlock Characterization, Deadlock Prevention, Deadlock Detection and Avoidance, Recovery from Deadlock.

UNIT – III: (12 hours)

Memory Management: Swapping, Contiguous Memory Allocation, Paging, Structure of the Page Table, Segmentation, Virtual memory, Demand Paging, Page-Replacement, Algorithms, Allocation of frames, thrashing.

File system Interface: Concept of a File, Access Methods, Directory structure, File System Mounting, File sharing, Protection.

File System Implementation: File -system structure, File- system Implementation, Directory Implementation, Allocation methods, Free-Space Management, Efficiency and Performance.

UNIT – IV: (12 hours)

Secondary Storage Structure: Overview of Mass-Storage Structure, Disk Structure, Disk Attachment, Disk Scheduling, Disk Formatting, Swap-Space Management, RAID structure.

Protection: Domain of Protection, Access Control, Access Matrix, Access Control Lists, Capability Lists.

Security: Security Objectives, Security Problems, Intruders, Inside System Attacks, Outside System Attacks, Cryptography as a Security Tool, Intrusion Detection System.

TEXT BOOKS

1. Operating System Concepts- Abraham Silberchatz, Peter B. Galvin, Greg Gagne 7th Edition, John Wiley.
2. Principles of Operating systems- NareshChauhan,Oxford Higher Education.

REFERENCE BOOKS:

1. Operating Systems – Internal and Design Principles Stallings, Fifth Edition–2005, Pearson education.
2. Operating System A Design Approach-Crowley, TMH.
3. Modern Operating Systems, Andrew S Tanenbaum Pearson/PHI.

OBJECT ORIENTED PROGRAMMING USING JAVA LAB

Course Code	**CS 526**	**L-P-T-Cr.:**	0	3	0	2	**Semester:**	**II**
Category:	Laboratory Course							
Prerequisite:	Programming in C lab, Data Structure Lab							

Learning Objective:

- Programming in the Java programming language
- Knowledge of object-oriented paradigm in the Java programming language
- The use of Java in a variety of technologies and on different platforms.

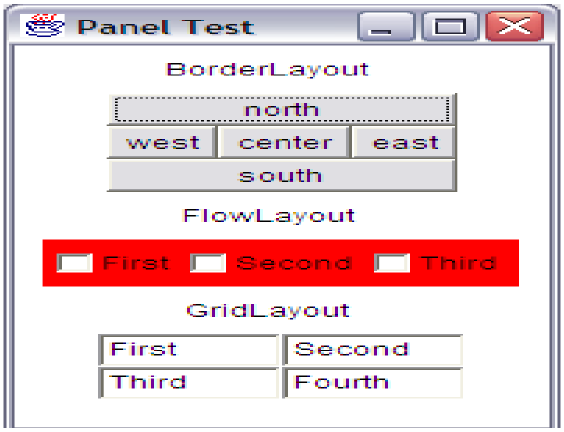
CO-1	Remember and understand the basic concepts/Principles of Object-Oriented Programming using JAVA Lab.
CO-2	Analyze the Various Concepts to understand them through case studies
CO-3	Apply the knowledge in understanding practical problems
CO-4	Execute/Create the Project or field assignment as per the knowledge gained in the course

LIST OF TOPICS FOR WRITING JAVA PROGRAMS:

1. Write a java program to read three numeric values (integer) from user and find the largest number among them.
2. Write a program to print the Fibonacci series up to a given number taken from user through command line.
3. Write a statistical computation program that to find out the maximum, minimum and mean value. Read input through command line.
4. WAP to create a class Rectangle (length, breadth), with zero argument constructor (default value is 5.0), one argument constructor (length = breadth), and two argument constructors, and define the methods area and perimeter of the rectangle. Create different objects with the help of three different constructors and print the area (length x breadth) and perimeter (2 x (length + breadth)) of those objects.
5. Define a class called Room with the following attributes 1. length, 2. breadth, 3. height, 4. floor_area, 5. Wall_area, 6. No._of_fans, 7. No.of_windows, 8. no.of_doors. Define a suitable constructor and a method to display details of a room. Assume that 20% of the total wall area is occupied by doors and windows and calculate accordingly. All data must be taken from user.
6. Define a class point, inherit class line from point, rectangle from line, and cube from rectangle. Write no argument constructor in each class. Write a print statement in these constructors mentioning which class it is. Create an object of the cube class in the main method of a separate class called test and show the output.
7. WAP to create a Person class having name, age and gender as instance variables. Write three constructors for constructor overloading like,
   a) First with no-argument.
   b) Second with three arguments for passing name, age and gender.
   c) Third with object as parameter to create a new copy of an existing Person object.

   Display the properties of Person class object with suitable methods.
8. Create an abstract class Shape with methods calc_area and calc_volume. Derive four classes Sphere(radius), Cone (radius, height) and Cylinder(radius, height), Box(length, breadth, height) from it. Calculate area and volume of all. (Use Method overriding).
9. Define an abstract class “Staff” with members name and address. Define two subclasses of this class – “FullTimeStaff” (department, salary) and “PartTimeStaff” (numberof- hours, rate-per-hour). Define appropriate constructors. Create n objects which could be of either FullTimeStaff or PartTimeStaff class by asking the user’s choice. Display details of all “FullTimeStaff” objects and all “PartTimeStaff” objects.
10. Define an interface “StackOperations” which declares methods for a static stack. Define a class “MyStack” which contains an array and top as data members and implements the above interface. Initialize the stack using a constructor. Write a menu driven program to perform operations on a stack object.
11. Define an interface “QueueOperations” which declares methods for a static queue. Define a class “MyQueue” which contains an array and front and rear as data members and implements the above interface. Initialize the queue using a constructor. Write a menu driven program to perform operations on a queue object.

12. Write a java program to create n objects of the Student class. Assign roll numbers in the ascending order using static method. Accept name and percentage from the user for each object. Define a method “sort Student” which sorts the array on the basis of percentage
13. Write a program to enter the student’s name, Rollno., Marks, in any no. of subjects as command line argument and find the percentage and grade of the student and thrown a NumberFormatException if required.
14. WAP having multiple catch and finally blocks where the catch blocks should handle the exceptions like, ArrayIndexOutOfBoundsException, NumberFormatException and ArithmeticException or any other exception.
15. Write a java program to creates ten threads, each of which do some work(search for the maximum value of a large matrix .Each thread searches one portion of the matrix.) It waits for them all to finish, then gathers the results.
16. Write a java program to show the use of synchronized method ().
17. Write a program to remove common characters from two strings.
18. Write a program to print all the palindrome words of a given string.
19. Input some strings through command line. Half of which will be stored in a String array and rest will be stored in a StringBuffer array. Write a program that will concatenate each element of this array of String objects with each element of StringBuffer objects. And the result will be stored in an array of StringBuffer.
20. Write an applet program to display the following by using different layouts.



DATA STRUCTURES LAB

Course Code	**CS 527**	**L-P-T-Cr.:**	0	3	0	2	**Semester:**	**II**

Category: Laboratory Course

Prerequisite: Programming in C Lab, Prior programming experience

Learning Objective:

- To understand how various data structures work.
- To understand some important applications of various data structures.
- To familiarize how certain applications can benefit from the choice of data structures.
- To understand how the choice of data structures can lead to efficient implementations of algorithms.

CO-1	Remember and understand the basic concepts/Principles of Data Structures Lab
CO-2	Analyze the Various Concepts to understand them through case studies
CO-3	Apply the knowledge in understanding practical problems
CO-4	Execute/Create the Project or field assignment as per the knowledge gained in the course

LIST OF TOPICS FOR WRITING PROGRAMS:

1. Write a program to create an array dynamically, accept its members and sort the array using following sorting algorithm. Also count the total number of swaps.
   a. Bubble sort
   b. Selection sort
   c. Insertion sort
   d. Quick sort
   e. Merge sort
2. Write a function search an element from the array using following searching techniques:
   a. Linear search
   b. Recursive linear search
   c. Binary search
   d. Recursive binary search
   e. Ternary search
3. Write a structure for an integer stack, implement **push, pop, and peek, IsEmpty** and **IsFull** function. Write a main function and call the functions based on user's choice.

```
typedef struct stack
{
        int top;
        int data[max];
}Stack;
```

4. Write a structure for an integer queue, implement **enqueue, dequeue, and traverse, IsEmpty** and **IsFull** function. Write a main function and call the functions based on user's choice.

```
typedefstruct queue
{
       int front,rear;
       int data[max];
}Queue;
```

5. Write a program to implement queue using two stacks. Include mystack.h and do the program.

6. Write a structure for an integer circular queue, implement **enqueue, dequeue, and traverse, IsEmpty** and **IsFull** function. Write a main function and call the functions based on user's choice.

```
typedef struct circularQueue
{
       int front,rear;
       int data[max];
} circularQueue;
```

7. Create a singly linked list of integers, write functions to add elements at different places (beginning, end, at a specified position), delete a node from different positions (beginning, end, at a specified position) and traverse the linked list based on user's choice.

8. Write a program to implement stack using linked list.
9. Write a program to implement Queue using linked list.
10. Create a singly circular linked list of integers, write functions to add elements at different places (beginning, end, at a specified position), delete a node from different positions (beginning, end, at a specified position) and traverse the linked list based on user's choice.
11. Create a doubly linked list of characters, write functions to add elements at different places (beginning, end, at a specified position), delete a node from different positions (beginning, end, at a specified position) and traverse the linked list in both directions based on user's choice.
12. Declare a binary search tree (BST) where information at each node would be a single integer. Write recursive and non-recursive (use mystack.h) functions for
    a. Inserting a key
    b. Deleting a key from the tree.
    c. Searching an element
    d. Inorder , Preorder and Postorder traversal
    e. Finding height of the tree
    f. Count number of nodes
    g. Display leaf nodes
13. Declare an AVL Tree where information at each node would be a single integer. Write recursive functions for
    a. Inserting a key
    b. Deleting a key from the tree.
    c. Searching an element
14. Write a program to implement single threaded binary tree and perform the following functions.
    a. Inserting a key
    b. Deletion of a key
    c. In-order traversal using the thread
    d. Maximum depth of the tree
15. Write a program for Breadth First Traversal of a graph.
16. Write a program for Depth First Traversal of a graph.
17. Write a program to check whether there is a path between two vertices of graph.
18. Given a directed graph. Write a program to find shortest path among all the nodes of a graph using Floyd- Warshall Algorithm.
19. Given an undirected, connected and weighted graph, find **M**inimum **S**panning **T**ree (MST) of the graph using Kruskal's Algorithm.
20. Given an undirected, connected and weighted graph, find **M**inimum **S**panning **T**ree (MST) of the graph using Prim's Algorithm.

Semester – III						
Code	**Course Title**	**Category**	**L**	**P**	**T**	**Credits**
CS 531	Compiler Design	Core	3	0	1	3
CS 532	Design and Analysis of Algorithms	Core	4	0	0	4
CS 533	Computer Graphics	Core	4	0	0	3
CS 534	Web Technology	Core	4	0	0	3
XX XXXX	Elective-I	Prog. Elect.	4	0	0	3
	CS 53E1 Mobile Computing					
	CS 53E2 Information Retrieval Systems					
	CS 53E3 Optimization Techniques					
	CS 53E4 Management Information Systems					
	CS53E5 Computer Based Numerical and Statistical Methods		4	0	0	
CS 535	Web Technology Lab.	Laboratory Course	0	3	0	2
CS 536	Python Programming Lab.	Laboratory Course	0	3	0	2
MOC 537	Big Data Computing (MOOCs-2)	MOOC	3	0	0	3
					Total Credit:	**23**

COMPILER DESIGN

Course Code	**CS 531**	**L-P-T-Cr.:**	3	0	1	3	**Semester:**	**III**
Category:	Core							
Prerequisite:	**Theory of Computation / Automata theory**							

Learning Objective:

- To learn various stages of compilation, design phases of a compiler construction process.
- This course will also introduce open source tool Lex and Yaac.

CO-1	Remember and understand the basic concepts/Principles of Compiler Design
CO-2	Analyze the Various Concepts to understand them through case studies
CO-3	Apply the knowledge in understanding practical problems
CO-4	Execute/Create the Project or field assignment as per the knowledge gained in the course

UNIT – I: **(8 hours)**

Overview of Compilation: Phases of Compilation – Lexical Analysis, Regular Grammar and regular expression for common programming language features, pass and Phases of translation, interpretation, bootstrapping, data structures in compilation – LEX lexical analyzer generator.

UNIT – II: **(14 hours)**

Syntax Analysis: Role of a parser, context free grammars and context free languages, parse trees and derivations, ambiguous grammar.

Top-Down Parsing: Recursive descent parsing, LL (1) grammars, non-recursive predictive parsing, error reporting and recovery.

Bottom Up Parsing: Handle pruning and shift reduces parsing, SLR parsers and construction of SLR parsing tables, LR(1) parsers and construction of LR(1) parsing tables, LALR parsers and construction of efficient LALR parsing tables, parsing using ambiguous grammars, error reporting and recovery, parser generator.

UNIT – III: **(12 hours)**

Syntax Directed Translation: Syntax directed definitions (SDD), inherited and synthesized attributes, dependency graphs, evaluation orders for SDD, semantic rules, application of syntax directed translation.

Symbol Table: Structure and features of symbol tables, symbol attributes and scopes.

Intermediate Code Generation: Introduction, benefits and types of intermediate code generation, three address codes - quadruples and triples, DAG for expressions, types and declarations, translation of expressions, translation of Boolean expressions and control flow statements, back patching, intermediate code generation for procedures.

UNIT – IV: **(13 hours)**

Run Time Environment: storage organizations, static and dynamic storage allocations, stack allocation, Activation of the procedure and the activation record.

Code Generations: Introduction, Major Issues of Code generation, registers allocation, simple code generation using basic blocks.

Elements of Code Optimization: Objective, peephole optimization, redundant and un-reachable codes, concepts of elimination of local common sub-expressions, basics of flow of control optimization.

TEXT BOOKS

1. Principles of Compiler Design, A.V. Aho .J.D.Ullman; Pearson Education.
2. Modern Compiler Implementation in C- Andrew N. Appel, Cambridge University Press.

REFERENCE BOOKS:

1. Lex&Yacc, John R. Levine, Tony Mason, Doug Brown, O’reilly
2. Modern Compiler Design, Dick Grune, Henry E. Bal, Cariel T. H. Jacobs, Wiley dreamtech.
3. Engineering a Compiler, Cooper & Linda, Elsevier.
4. Compiler Construction, Louden, Thomson

DESIGN AND ANALYSIS OF ALGORITHM

Course Code	**CS 532**	**L-P-T-Cr.:**	4	0	0	4	**Semester:**	**III**
Category:	Core							
Prerequisite:	**Data Structure**							

Learning Objective:

- Learn the algorithm analysis techniques.
- Become familiar with the different algorithm design techniques.
- Understand the limitations of Algorithm power

CO-1	Remember and understand the basic concepts/Principles of Design and Analysis of Algorithm
CO-2	Analyze the Various Concepts to understand them through case studies
CO-3	Apply the knowledge in understanding practical problems
CO-4	Execute/Create the Project or field assignment as per the knowledge gained in the course

UNIT – I: **(10 hours)**

Introduction, Definition, Chrematistics of algorithm, Growth of Functions, Asymptotic analysis, Amortized analysis, standard notations and common functions, Recurrences, solution of recurrences by substitution, recursion tree, induction method, and Master methods, Algorithm design techniques, worst case analysis of Merge sort, Quick sort and Binary search, Design & Analysis of Divide and conquer algorithms.

UNIT – II: **(10 hours)**

Heapsort mechanism, Heaps, Building a heap, The heapsort algorithm, Priority Queue, Lower bounds for sorting. Dynamic programming methodology, Elements of dynamic programming, Matrix-chain multiplication, Longest common subsequence, Greedy Algorithms, Elements of Greedy strategy, Assembly-line scheduling, Activity selection Problem, Fractional knapsack problem, Huffman codes).

UNIT – III: **(10 hours)**

Data structure for disjoint sets, Disjoint set operations, Linked list representation, B and B + tree, connected components and bi connected components. Breadth first search and depth-first search, Minimum Spanning Trees, Kruskal algorithm and Prim's algorithms, single- source shortest paths (Bellman-ford algorithm and Dijkstra’s algorithms), All-pairs shortest paths (Floyd – Warshall Algorithm).

UNIT – IV: **(10 hours)**

Back tracking, Branch and Bound, Eight Queen problem, Travelling sales person problem, 0/1 knapsack problem, NP - Completeness (Polynomial time, Polynomial time verification, NP -Completeness and reducibility, NP-Complete problems (without Proofs), Approximation algorithms characteristics, Traveling Salesman Problem.

TEXT BOOKS

1. T.H. Cormen, C.E. Leiserson, R.L. Rivest, C.Stein : Introduction to Algorithms, 2nd Edition, PHI Learning Pvt. Ltd.
2. H. Bhasin: Algorithms, Design and Analysis, First Edition, Oxford Higher Education.

REFERENCE BOOKS:

1. Sanjay Dasgupta, Umesh Vazirani: Algorithms, McGraw-Hill Education.
2. Horowitz & Sahani: Fundamentals of Algorithm, 2nd Edition, Universities Press.
3. Goodrich, Tamassia: Algorithm Design, Wiley India.

COMPUTER GRAPHICS

Course Code	**CS 533**	**L-P-T-Cr.:**	4	0	0	3	**Semester:** **III**
Category:	Core						
Prerequisite:	**Knowledge on C programming and mathematics**						

Learning Objective:

- To identify and understand the core concepts of computer graphics
- To apply graphics programming techniques to design and create computer graphics scenes.
- To learn about the 2D and 3D transformations including translation, scaling, rotation and reflection
- To understand principle of clipping, basic line-clipping algorithms
- To learn about application of curves in computer graphics

CO-1	Remember and understand the basic concepts/Principles of Computer Graphics
CO-2	Analyze the Various Concepts to understand them through case studies
CO-3	Apply the knowledge in understanding practical problems
CO-4	Execute/Create the Project or field assignment as per the knowledge gained in the course

UNIT – I: **(10 hours)**

Graphics Hardware: Display devices, input devices, Raster Graphics.

Line and Circle drawing algorithms: DDA, Bresenham's line drawing algorithm, midpoint circle drawing algorithm.

UNIT – II: **(10 hours)**

Windowing and Clipping: Viewing and Window coordinate System, Viewport, Window, Zoom-in, Zoom-out, Cohen Sutherland, Cyrus beck line clipping algorithms

2D and 3D Geometrical Transformations: Homogeneous Coordinate system, Basic transformations: Translation, Scaling, Rotation and Reflection.

UNIT – III: **(10 hours)**

Viewing Transformation: Parallel Projection: Orthographic, Axonometric, Cavalier and Cabinet. **Perspective Projection:** one point, two-point, three-point perspective projection, vanishing point.

UNIT – IV: **(10 Hours)**

Curve and Surfaces: Properties of curves, Blending functions: Cubic Bezier and B-Spline curves.

Parametric Surfaces: Surface of revolution Sweep surfaces, Fractal curves and surfaces,

Hidden line/surface removal: Object space and Image space methods, Inside- outside test, Back Face detection: Z-buffer, A-Buffer Methods. Introduction to computer animation.

TEXT BOOKS

1. Hearn D. and P. Baker, Computer Graphics C version, Prentice-Hall. (Major Reading)

REFERENCE BOOKS:

1. David F. Rozers, Procedural Elements for Computer Grapihics, TMH.
2. David F. Rozers, Mathematical Elements for Computer graphics, TMH.
3. Foley, J.D. A. Van Dam, Computer Graphics: Principles and Practice, Addison- Wesley.

WEB TECHNOLOGY

Course Code	**CS 534**	**L-P-T-Cr.:**	4	0	0	3	**Semester:**	**III**
Category:	Core							
Prerequisite:	**Fundamentals of Programming and Networking**							

Learning Objective:

- Describe the concepts of WWW including browser and HTTP protocol.
- List the various HTML tags and use them to develop the user-friendly web pages.
- Define the CSS with its types and use them to provide the styles to the web pages at various levels.
- Develop the modern web pages using the HTML and CSS features with different layouts as per need of applications.
- Use the JavaScript to develop the dynamic web pages.
- Use server-side scripting with PHP to generate the web pages dynamically using the database connectivity.
- Develop the modern Web applications using the client and server-side technologies and the web design fundamentals.

CO-1	Remember and understand the basic concepts/Principles of Web Technology
CO-2	Analyze the Various Concepts to understand them through case studies
CO-3	Apply the knowledge in understanding practical problems
CO-4	Execute/Create the Project or field assignment as per the knowledge gained in the course

UNIT – I: **(8 hours)**

Introduction: Concept of WWW, Internet and WWW, HTTP Protocol: Request and Response, Web browser and Web servers, Features of Web 2.0.

Web Design: Concepts of effective web design, Web design issues including Browser, Bandwidth and Cache, Display resolution, Look and Feel of the Website, Page Layout and linking, User centric design, Sitemap, Planning and publishing website, Designing effective navigation

UNIT – II: **(14 hours)**

HTML: Basics of HTML, formatting and fonts, commenting code, color, hyperlink, lists, tables, images, forms, XHTML, Meta tags, Character entities, frames and frame sets, Browser architecture and Web site structure. Overview and features of HTML5.

Style sheets: Need for CSS, introduction to CSS, basic syntax and structure, using CSS, background images, colors and properties, manipulating texts, using fonts, borders and boxes, margins, padding lists, positioning using CSS, CSS2, Overview and features of CSS3

UNIT – III: **(14 hours)**

JavaScript: Client-side scripting with JavaScript, variables, functions, conditions, loops and repetition, Pop up boxes, Advance JavaScript: JavaScript and objects, JavaScript own objects, the DOM and web browser environments, Manipulation using DOM, forms and validations, DHTML: Combining HTML, CSS and JavaScript, Events and buttons.

XML: Introduction to XML, uses of XML, simple XML, XML key components, DTD and Schemas, Using XML with application. Transforming XML using XSL and XSLT

UNIT – IV: **(13 hours)**

PHP: Introduction and basic syntax of PHP, decision and looping with examples, PHP and HTML, Arrays, Functions, Browser control and detection, string, Form processing, Files, Advance Features: Cookies and Sessions, Object Oriented Programming with PHP.

PHP and MySQL: Basic commands with PHP examples, Connection to server, creating database, selecting a database, listing database, listing table names, creating a table, inserting data, altering tables, queries, deleting database, deleting data and tables, PHP myadmin and database bugs.

TEXT BOOKS

1. Developing Web Applications, Ralph Moseley and M. T. Savaliya, Wiley-India.
2. Web Technologies, Black Book, dreamtech Press
3. HTML 5, Black Book, dreamtech Press

REFERENCE BOOKS:

1. Web Design, Joel Sklar, Cengage Learning
2. Developing Web Applications in PHP and AJAX, Harwani, McGrawHill

Elective-I

MOBILE COMPUTING

Course Code	**CS 53E1**	**L-P-T-Cr.:**	4	0	0	3	**Semester:**	**III**
Category:	Program Elective Course							
Prerequisite:	**Data communication and Computer Networks**							
Learning Objective:	• Describe wireless and mobile communications systems and be able to choose an appropriate mobile system from a set of requirements.							

CO-1	Remember and understand the basic concepts/Principles of Mobile Computing
CO-2	Analyze the Various Concepts to understand them through case studies
CO-3	Apply the knowledge in understanding practical problems
CO-4	Execute/Create the Project or field assignment as per the knowledge gained in the course

UNIT – I: **(10 hours)**

Introduction to Mobile Communications and Computing: Mobile Computing (MC): Introduction to MC, novel applications, limitations, and architecture.

GSM: Mobile services, System architecture, Radio interface, Protocols, Localization and calling, Handover, Security, and New data services.

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

UNIT – II: **(10 hours)**

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: **(10 hours)**

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: **(10 hours)**

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

1. Jochen Schiller, “Mobile Communications”, Addison-Wesley.
2. Stojmenovic and Cacute, “Handbook of Wireless Networks and Mobile Computing”

REFERENCE BOOKS:

1. Reza Behravanfar, “Mobile Computing Principles: Designing and Developing Mobile Applications with UML and XML”, ISBN: 0521817331, Cambridge University Press.
2. Adelstein, Frank, Gupta, Sandeep KS, Richard III, Golden, Schwiebert, Loren, “Fundamentals of Mobile and Pervasive Computing”, McGraw-Hill Professional.
3. Hansmann, Merk, Nicklous, Stober, “Principles of Mobile Computing”, Springer.
4. MartynMallick, “Mobile and Wireless Design Essentials”, Wiley DreamTech.

INFORMATION RETRIEVAL SYSTEMS

Course Code	**CS 53E2**	**L-P-T-Cr.:**	4	0	0	3	**Semester:**	**III**
Category:	Program Elective Course							
Prerequisite:	**Data Structure and Algorithm, Linear Algebra, Basics of Web programming**							
Learning Objective:	• This course will cover traditional material, as well as recent advances in Information Retrieval (IR), the study of indexing, processing, and querying textual data. Basic retrieval models, algorithms, and IR system implementations will be covered. It will also cover web search, link analysis.							

CO-1	Remember and understand the basic concepts/Principles of Information Retrieval Systems
CO-2	Analyze the Various Concepts to understand them through case studies
CO-3	Apply the knowledge in understanding practical problems
CO-4	Execute/Create the Project or field assignment as per the knowledge gained in the course

UNIT – I: **(8 hours)**

Introduction to course: Discussion of issues in search, Introduction to Information Retrieval. Inverted indices and Boolean queries. Query optimization. The nature of unstructured and semi-structured text. Course administrivia. The term vocabulary and postings lists.

Text encoding: tokenization, stemming, lemmatization, stop words, phrases. Optimizing indices with skip lists. Proximity and phrase queries. Positional indices.

UNIT – II: **(10 hours)**

Index construction. Postings size estimation, sort-based indexing, dynamic indexing, positional indexes, n-gram indexes, distributed indexing, real-world issues.

Index compression: lexicon compression and postings lists compression. Gap encoding, gamma codes, Zipf's Law, variable-byte encoding. Blocking. Extreme compression.

Dictionaries and tolerant retrieval. Dictionary data structures. Wild-card queries, permuterm indices, n-gram indices. Spelling correction and synonyms: edit distance, soundex, language detection.

Scoring, term weighting, and the vector space model. Parametric or fielded search. Document zones. The vector space retrieval model. TF/IDF weighting. The cosine measures. Scoring documents.

UNIT – III: **(8 hours)**

Computing scores in a complete search system: Components of an IR system. Efficient vector space scoring. Nearest neighbor techniques, reduced dimensionality approximations, random projection.

Results summaries: static and dynamic. Evaluating search engines. User happiness, precision, recall, F-measure. **Creating test collections:** kappa measure, inter judge agreement. Relevance, approximate vector retrieval

Probabilistic IR. Binary Independence Model.

UNIT – IV: **(14 hours)**

Classification: Introduction to text classification. Naive Bayes models. Spam filtering. Probabilistic IR. K Nearest Neighbors, Decision boundaries, Vector space classification using centroids. Support vector machine classifiers. Kernel Function. Evaluation of classification. Micro- and macro-averaging. Learning rankings.

Clustering: Introduction to the problem. Partitioning methods: k-means clustering; Hierarchical clustering. Learning to rank. Latent semantic indexing (LSI). Applications to clustering and to information retrieval.

Web search overview, web structure, the user, paid placement, search engine optimization/spam. Web size measurement. Link analysis, Crawling and web indexes. Near-duplicate detection.

TEXT BOOKS

1. Introduction to Information Retrieval, Christopher D. Manning, PrabhakarRaghavan, HinrichSchütze, Cambridge University Press.

REFERENCE BOOKS:

1. Readings in Information Retrieval, K.Sparck Jones and P. Willet, Morgan Kaufmann.
2. Modern Information Retrieval, Ricardo BaezaYates and BerthierRibeiroNeto, Ricardo Baeza-Yates and BerthierRibeiro-Neto Addison-Wesley.

OPTIMIZATION TECHNIQUES

Course Code	**CS 53E3**	**L-P-T-Cr.:**	4	0	0	3	**Semester:**	**III**

Category: Program Elective Course

Prerequisite: **Basic Knowledge of Linear Algebra and Matrix Theory.**

Learning Objective:

- To introduce the fundamental concepts of Optimization Techniques.
- To make the learners aware of the importance of optimizations in real scenarios and modern application in computer science like AI, Machine learning, sensor network and routing protocol.
- Optimization methods using calculus have several limitations and thus not suitable for many practical applications. Most widely used optimization method is linear programming which is the main objective of this module.
- To provide the concepts of various classical and modern methods of for constrained and unconstrained problems in both single and multivariable.

CO-1	Remember and understand the basic concepts/Principles of Optimization Techniques
CO-2	Analyze the Various Concepts to understand them through case studies
CO-3	Apply the knowledge in understanding practical problems
CO-4	Execute/Create the Project or field assignment as per the knowledge gained in the course

UNIT – I: OPTIMIZATION & SIMPLEX METHOD (10 hours)

Introduction to Optimization, Classification of Optimization Problem, Classical Optimization Technique, Single –Variable Optimization, Multivariable optimization, Direct Solution, Constrained variation, Lagrange multiplier, linear programming, Standard form of linear programming, Simplex Algorithm, Two Phage of simplex Method, Big M-Method.

UNIT – II: DUALITY, ASSIGNMENT & TRANSPIRATION (10 hours)

Formation of dual problems, Important Result in Duality, Dual simplex Method, Mathematical formulation of Assignment problem, Hungarian Method, The travelling Salesman Problem, Transportation Problem, North West Corner Rule, Optimality Test, Difference of Assignment and Transportation problem.

UNIT – III: INTEGER PROGRAMMING & GEOMETRIC PROGRAMMING (10 hours)

Introduction, Importance of Integer Programming Problems, Application, Methods of Integer Programming Problem, Cutting, Search, Cutting Plane, Mixed Integer Programming, Unconstrained Geometric programming, Constrained Minimization, Mixed inequality.

UNIT – IV: NETWORK SCHEDULING & GAME THEORY (10 Hours)

Introduction, Rules of Network Construction, Time Analysis, Critical Path Method (CPM), Program Evaluation and Review Techniques (PERT), Cost Consideration in PERT/CPM, Payoff, Types of Game, The Maxmin – Minimax Principle

TEXT BOOKS

1. Engineering Optimization, Singiresu S. Rao, New Age International Publisher.
2. Operation Research, Kanti Swarup, S Chand & Sons Publisher

REFERENCE BOOKS:

1. Operations Research, Dr. S.D. Sharma
2. Operations Research: An Introduction. Taha, PHI Pvt. Ltd.

MANAGEMENT INFORMATION SYSTEM

Course Code	**CS 53E4**	**L-P-T-Cr.:**	4	0	0	3	**Semester:**	**III**
Category:	Program Elective Course							
Prerequisite:	**None**							
Learning Objective:	• The objective of the course is to develop the basic understanding of the decision support system of the artificial intelligence for business organization. Implication of emerging trends in technology.							

CO-1	Remember and understand the basic concepts/Principles of Management Information System
CO-2	Analyze the Various Concepts to understand them through case studies
CO-3	Apply the knowledge in understanding practical problems
CO-4	Execute/Create the Project or field assignment as per the knowledge gained in the course

UNIT – I: **(10 hours)**

Management Information system: Introduction, objective, definition, benefits, characteristics of MIS, information system level, types of information system, resistance to MIS, implementing MIS, features of MIS, components of MIS.

UNIT – II: **(12 hours)**

Managerial decision making: Decision making process, problem solving techniques, how decisions are being supported – decisions styles group, Simon Model of decision making, features of various CBIS.

Decision support system overview - relevance, scope, characteristic and capabilities, components and classification of DSS Decision support System: Introduction, architecture, components, limitation, development.

UNIT – III: **(10 hours)**

Database management system: Objective, characteristics, components and use of DBMS, types of database, Role of DBA. Model base management system: types of models, certainty, uncertainty, risk

UNIT – IV: **(12 hours)**

Information Security challenges in E-enterprise: Introduction, Security Threats and Vulnerability, Controlling Security Threat and Vulnerability, Management Security Threat in E-business, Disaster Management, MIS and Security Challenges, Software security: threats, method of safety, cryptography, digital signature.

Introduction to Emerging trends technology, Expert System, knowledge management, A.I., data mining, data warehousing

TEXT BOOKS

1. Keen, peter G.W.: Decision Support System an Organisational Perspective Addison-Wesley Pub.
2. Theierauff, Robert J. Decision Support System for effective planning – Prentice Hall – 1982.

REFERENCE BOOKS:

1. Kroger, Donald W., and Hugh J. Watson Computer Based Information System New York, 1984.
2. Davis, Michael W. A management Approach – Macmillan Publishing company, Prentice Hall, New Jersey,1988.
3. Andrew P. Decision support System Engineering, Sage, John Wiley & Sons, New York, 1991.

COMPUTER BASED NUMERICAL AND STATISTICAL METHODS

Course Code	**CS 53E5**	**L-P-T-Cr.:**	4	0	0	3	**Semester:**	**III**

Category: Program Elective Course

Prerequisite: NA

Learning Objective:

- To understand Computation and to make use the computers effectively for solving problems. Designing Flowcharts, algorithms for providing apt computation.
- To familiarize students with statistical and numerical techniques needed in problem-solving and industrial applications.

CO-1	Remember and understand the basic concepts/Principles of Computer Based Numerical Statistical Methods
CO-2	Analyze the Various Concepts to understand them through case studies
CO-3	Apply the knowledge in understanding practical problems
CO-4	Execute/Create the Project or field assignment as per the knowledge gained in the course

UNIT – I: **(10 hours)**

Introduction: Numbers and their accuracy, Computer Arithmetic, Mathematical preliminaries, Errors and their Computation, General error formula, Error in a series approximation

Solution of Algebraic and Transcendental Equation: Bisection Method, Iteration method, Method of false position, Newton-Raphson method, Methods of finding complex roots, Muller's method, Rate of convergence of Iterative methods, Polynomial Equations.

UNIT – II: **(10 hours)**

Interpolation: Finite Differences, Difference tables Polynomial Interpolation**:** Newton's forward and backward formula Central Difference Formulae: Gauss forward and backward formula, Stirling's, Bessel's, Everett's formula. Interpolation with unequal intervals: Langrange's Interpolation, Newton Divided difference formula, Hermite's Interpolation,

Numerical Integration and Differentiation: Introduction, Numerical differentiation Numerical Integration: Trapezoidal rule, Simpson's 1/3 and 3/8 rule, Boole's rule, Waddle's rule.

UNIT – III: **(10 hours)**

Solution of differential Equations: Picard's Method, Euler's Method, Taylor's Method, Runge-Kutta Methods, Predictor Corrector Methods, Automatic Error Monitoring and Stability of solution

UNIT – IV: **(10 hours)**

Statistical Computation: Frequency chart, Curve fitting by method of least squares, fitting of straight lines, polynomials, exponential curves etc., Data fitting with Cubic splines, Regression Analysis, Linear and Nonlinear Regression, Multiple regression, Statistical Quality Control methods.

TEXT BOOKS

1. Rajaraman V, "Computer Oriented Numerical Methods", Pearson Education
2. Gerald & Whealey, "Applied Numerical Analyses", AW

REFERENCE BOOKS:

1. Jain, Iyengar and Jain, "Numerical Methods for Scientific and Engineering Computations", New Age Int.
2. Grewal B S, "Numerical methods in Engineering and Science", Khanna Publishers, Delhi
3. T Veerarajan, T Ramachandran, "Theory and Problems in Numerical Methods, TMH
4. Goyal, M, "Computer Based Numerical and Statistical Techniques", Firewall Media, New Delhi.

WEB TECHNOLOGY LAB

Course Code	**CS 535**	**L-P-T-Cr.:**	0	3	0	2	**Semester:**	**III**
Category:	Laboratory Course							
Prerequisite:	NA							

Learning Objective:

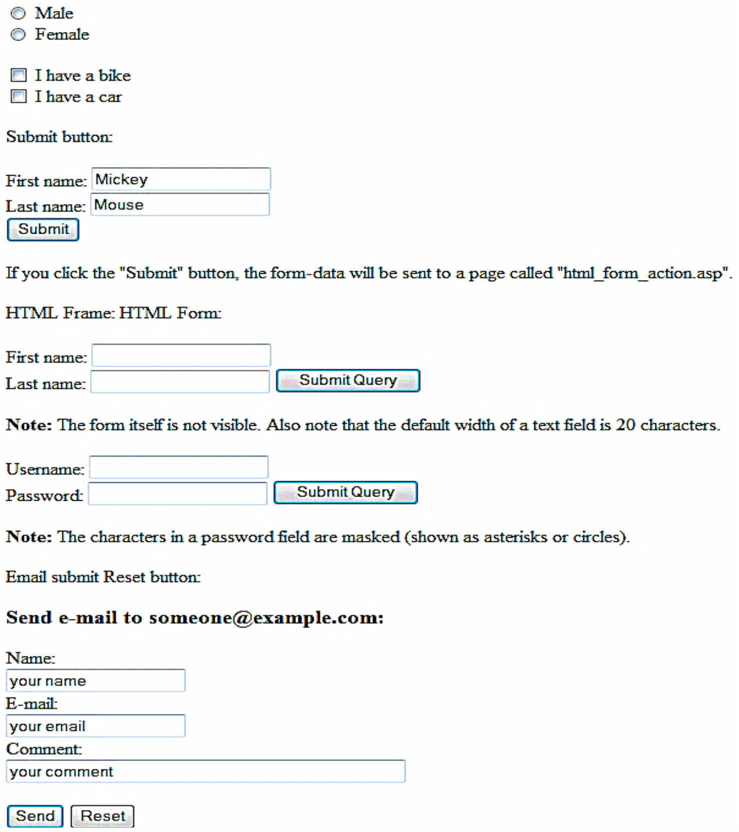
- HTML, the fundamentals of how the Internet and the Web function
- A basic understanding of graphic production with a specific stress on creating graphics for the Web
- A general grounding introduction to more advanced topics such as programming and scripting. This will also expose students to the basic tools and applications used in Web publishing.

CO-1	Remember and understand the basic concepts/Principles of Web Technology Lab
CO-2	Analyze the Various Concepts to understand them through case studies
CO-3	Apply the knowledge in understanding practical problems
CO-4	Execute/Create the Project or field assignment as per the knowledge gained in the course

LIST OF TOPICS:

1. Design the following static web pages required for an online book store web site.
   1) HOME PAGE: The static home page must contain three **frames**.
   2) LOGIN PAGE
   3) CATOLOGUE PAGE: The catalogue page should contain the details of all the books available in the web site in a table.
   4) CART PAGE: The cart page contains the details about the books which are added to the cart.
   5) REGISTRATION PAGE
2. Write JavaScript to validate the fields of the above page. Write JavaScript to validate the fields of the Login page.
3. Design a web page using CSS which includes the following:
   1) Use different font, styles:
   2) Set a background image for both the page and single elements on the page.
   3) Control the repetition of the image with the background-repeat property.
   4) Define styles for links
   5) Working with layers
   6) Add a customized cursor
4. Write an XML file which will display the Book information. Write a Document Type Definition (DTD) to validate the above XML file. Display the XML file in a table. Use XML schemas XSL and CSS for the above purpose.
5. Create a simple visual bean with a area filled with a color. The shape of the area depends on the property shape. If it is set to true then the shape of the area is Square and it is Circle, if it is false. The color of the area should be changed dynamically for every mouse click. The color should also be changed if we change the color in the “property window”.

6. Design the following Web page.



○ Male
○ Female

□ I have a bike
□ I have a car

Submit button:

First name: Mickey
Last name: Mouse
Submit

If you click the "Submit" button, the form-data will be sent to a page called "html_form_action.asp".

HTML Frame: HTML Form:

First name:
Last name: Submit Query

Note: The form itself is not visible. Also note that the default width of a text field is 20 characters.

Username:
Password: Submit Query

Note: The characters in a password field are masked (shown as asterisks or circles).

Email submit Reset button:

Send e-mail to someone@example.com:

Name:
your name
E-mail:
your email
Comment:
your comment

Send Reset

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. Assume four users user1, user2, user3 and user4 having the passwords pwd1, pwd2, pwd3 and pwd4 respectively. Write a servlet 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 and authenticate with the values available in the cookies.
9. Install a database (Mysql). Create a table which should contain at least the following fields: name, password, email-id, phone number Write a PHP program to connect to that database and extract data from the tables and display them. Insert the details of the users who register with the web site, whenever a new user clicks the submit button in the registration page.
10. Write a PHP which insert the details of the 3 or 4 users who register with the web site by using registration form. Authenticate the user when he submits the login form using the user name and password from the database.

PYTHON PROGRAMMING LAB

Course Code	**CS 536**	**L-P-T-Cr.:**	0	3	0	2	**Semester:**	**III**

Category: Laboratory Course

Prerequisite: Programming in C, Object Oriented Programming, Scripting Language

Learning Objective:

- To learn how to use lists, tuples, and dictionaries in Python programs.
- To learn how to identify Python object types.
- To learn how to use indexing and slicing to access data in Python programs.
- To define the structure and components of a Python program.
- To learn how to write loops and decision statements in Python.
- To learn how to write functions and pass arguments in Python.

CO-1	Remember and understand the basic concepts/Principles of Python Lab
CO-2	Analyze the Various Concepts to understand them through case studies
CO-3	Apply the knowledge in understanding practical problems
CO-4	Execute/Create the Project or field assignment as per the knowledge gained in the course

LIST OF TOPICS FOR WRITING PYTHON PROGRAMS:

1. Python Overview and Command line
2. Python Keywords, Variables
3. Python Identifiers, Lines Indention, Comments
4. Python Data types and Operators
5. Python Conditional Statements (if, if…else, if…elif, nested if statement)
6. Python for() loops and functions used in for() loop
7. Python while() loop
8. Python user-defined function
9. File Operation using python
10. Python Modules: creating a user-defined module, installing and importing a predefined module

<table>
<tr><th colspan="8">Semester – IV</th></tr>
<tr><th>Code</th><th colspan="2">Course Title</th><th>Category</th><th>L</th><th>P</th><th>T</th><th>Credits</th></tr>
<tr><td><b>CS 541</b></td><td colspan="2">Data Warehousing and Data Mining</td><td>Core Course</td><td>4</td><td>0</td><td>0</td><td>4</td></tr>
<tr><td><b>CS 542</b></td><td colspan="2">Artificial Intelligence</td><td>Core Course</td><td>4</td><td>0</td><td>0</td><td>4</td></tr>
<tr><td rowspan="6"><b>XX XXXX</b></td><td colspan="2">Elective-II</td><td>Prog. Elect.</td><td rowspan="6">4</td><td rowspan="6">0</td><td rowspan="6">0</td><td rowspan="6">4</td></tr>
<tr><td><b>CS 54E1</b></td><td colspan="2">Wireless Sensor Networks</td></tr>
<tr><td><b>CS 54E2</b></td><td colspan="2">Cloud Computing</td></tr>
<tr><td><b>CS 54E3</b></td><td colspan="2">Machine Learning</td></tr>
<tr><td><b>CS 54E4</b></td><td colspan="2">Introduction to Big Data Analytics</td></tr>
<tr><td><b>CS 54E5</b></td><td colspan="2">Information & Cyber Security</td></tr>
<tr><td><b>CS 543</b></td><td colspan="2">Project</td><td>Project Work</td><td>-</td><td>-</td><td>-</td><td>8</td></tr>
<tr><td><b>CS 544</b></td><td colspan="2">Seminar</td><td>Tech. Seminar</td><td>-</td><td>-</td><td>-</td><td>2</td></tr>
<tr><td colspan="7" align="right"><b>Total Credit:</b></td><td><b>22</b></td></tr>
</table>

DATA WAREHOUSING AND DATA MINING

Course Code	**CS 541**	**L-P-T-Cr.:**	4	0	0	4	**Semester:**	**IV**
Category:	Core							
Prerequisite:	**Data Structure and Algorithm, Linear Algebra, Basics of Web programming**							
Learning Objective:	• This course deals with evolving multidimensional intelligent model from a typical system, representation of multi-dimensional data for a data warehouse, discovering the knowledge imbibed in the high dimensional system, finding the hidden interesting patterns in data, and gives the idea to evaluate various mining techniques on complex data objects.							

CO-1	Remember and understand the basic concepts/Principles of Data Warehousing and Data Mining
CO-2	Analyze the Various Concepts to understand them through case studies
CO-3	Apply the knowledge in understanding practical problems
CO-4	Execute/Create the Project or field assignment as per the knowledge gained in the course

UNIT – I: INTRODUCTION TO DATA WAREHOUSING AND ARCHITECTURE (08 hours)

Evolution of Decision Support Systems: Data warehousing Components –Building a Data warehouse, Data Warehouse and DBMS, Data marts, Metadata, Multidimensional data model, OLAP vs OLTP, OLAP operations, Data cubes, Schemas for Multidimensional Database: Stars, Snowflakes and Fact constellations

UNIT – II: DATA WAREHOUSE PROCESS AND ARCHITECTURE (08 hours)

Types of OLAP servers, 3–Tier data warehouse architecture, distributed and virtual data warehouses. Data warehouse implementation, tuning and testing of data warehouse. Data Staging (ETL) Design and Development, data warehouse visualization, Data Warehouse Deployment, Maintenance, Growth, Business Intelligence **Overview:** Data Warehousing and Business Intelligence Trends - Business Applications

UNIT – III: INTRODUCTION TO DATA MINING AND CLASSIFICATIONS (14 hours)

Data mining: KDD versus datamining, Stages of the Data Mining Process-task primitives, Data Mining Techniques

Data mining knowledge representation: Data mining query languages, Integration of a Data Mining System with a Data Warehouse – Issues, Data preprocessing – Data cleaning, Data transformation, Feature selection, Dimensionality reduction, Discretization and generating concept hierarchies-Mining frequent patterns- association-correlation.

Decision Tree Induction – Bayesian Classification – Rule Based Classification –Classification by Back propagation – Support Vector Machines – Associative Classification – Lazy Learners – Other Classification Methods

UNIT – IV: CLUSTERING, ADVANCES IN DATA MINING (10 hours)

Clustering techniques: Partitioning methods- k-means Hierarchical Methods - distance-based agglomerative and divisible clustering,

Mining complex data objects, Spatial databases, temporal databases, Multimedia databases, Time series and **Sequence data:** Text Mining: Graph mining-web mining-Application and trends in data mining

TEXT BOOKS:

1. Jiawei Han and Micheline Kamber, Data Mining: Concepts and Techniques, Morgan Kaufmann Publishers, third edition 2011, ISBN: 1558604898.
2. Alex Berson and Stephen J. Smith, “Data Warehousing, Data Mining & OLAP”, TataMc Graw Hill Edition, Tenth Reprint 2007.
3. G. K. Gupta, “Introduction to Data Min Data Mining with Case Studies”, Easter Economy Edition, Prentice Hall of India, 2006

REFERENCE BOOKS:

1. Mehmedkantardzic, “Datamining concepts, models, methods, and algorithms”, Wiley Interscience, 2003.
2. Ian Witten, Eibe Frank, Data Mining; Practical Machine Learning Tools and Techniques, third edition, Morgan Kaufmann, 2011.
3. George M Marakas, Modern Data Warehousing, Mining and Visualization, Prentice Hall, 2003

WEB REFERENCES

1. http://www.data-miners.com/

ARTIFICIAL INTELLIGENCE

Course Code **CS 542** **L-P-T-Cr.:** 4 0 0 4 **Semester:** **IV**

Category: Core

Prerequisite: **Linear Algebra, Programming Language**

Learning Objective:

- To learn the difference between optimal reasoning VS human like reasoning.
- To understand the notions of state space representation and heuristic search.
- To learn different knowledge representation techniques.
- To understand the applications of AI: namely Game playing, Theorem Proving. Expert systems, machine learning and Natural language Processing.

CO-1	Remember and understand the basic concepts/Principles of Artificial Intelligence
CO-2	Analyze the Various Concepts to understand them through case studies
CO-3	Apply the knowledge in understanding practical problems
CO-4	Execute/Create the Project or field assignment as per the knowledge gained in the course

UNIT – I: **(10 hours)**

Introduction to Artificial Intelligence: AI Problems, AI Techniques, Problems, Problem Space and Search, Defining the problem as a state space search, Production system, Problem characteristics

Heuristic search Technologies: Generate and Test, Hill Climbing, Best First Search, Problem Reduction, means-end-analysis, optimal and A*, AND-OR Graphs, AO* Algorithms.

UNIT – II: **(12 hours)**

Representation Knowledge using Predicate Logic: representing simple facts in logic, Representing Instance and ISA relationships, Computable functions and Predicates, Resolution, Representing Knowledge using Rules, Forward Vs Backward Reasoning, Matching, Control Knowledge, Weak slot and Filter structures, Semantic nets, Frames.

UNIT – III: **(12 hours)**

Strong slot and Filter structures, Conceptual Dependencies, Scripts. Introduction to Non monotonic reasoning, Logics for Non monotonic reasoning, Implementation: Depth First Search, Dependency-Directed Back Tracking, Justification based Truth Maintenance Logic based Truth Maintenance systems, Statistical Reasoning, Probability and Bayes Theorem, Certainty factors, Rule based Systems, Bayesian Networks, Dempster-Shaffer Theory.

UNIT – IV: **(12 hours)**

Minmax search, alpha-beta cutoffs, Planning system, Goal stack planning, Hierarchical Planning, Natural Language Processing., Syntactic Analysis, Semantic Analysis, Discuses and Pragmatic Processing. Introduction and Fundamentals of Artificial Neural Networks, Biological Prototype, Artificial Neuron, Single Layer Artificial Neural Networks, Multilayer Artificial Neural Networks, Training of Artificial Neural Networks

TEXT BOOKS:

1. Elaine Rich, Kevin Knight and Shivashankar B. Nair, Artificial Intelligence, 3/e, McGraw Hill Education, 2008.
2. Neural Computing: Theory and practice- Waserman.

REFERENCE BOOKS:

1. Artificial Intelligence Structures and Strategies complex problem solving-George F. Lugar Pearson Education
2. Stuart Russell and Peter Norvig, Artificial Intelligence: A Modern Approach, 2/e, Pearson Education, 2010.
3. Dan W. Patterson, Artificial Intelligence and Expert Systems, PHI.
4. Neural Networks: A Comprehensive Foundation 2/e- Symen Pearson Education.

ELECTIVE-II

WIRELESS SENSOR NETWORKS

Course Code	**CS 54E1**	**L-P-T-Cr.:**	4	0	0	4	**Semester:**	**IV**
Category:	Program Elective Course							
Prerequisite:	**Basic Computer Network**							

Learning Objective:

- To obtain a broad understanding about the network architecture of wireless sensor network. Understand all basic characteristics of wireless sensor networks and sensor nodes. The principles of data transmission, clustering algorithm and routing protocols. Design and development of new network architecture and MAC protocols.

CO-1	Remember and understand the basic concepts/Principles of Wireless Sensor Network
CO-2	Analyze the Various Concepts to understand them through case studies
CO-3	Apply the knowledge in understanding practical problems
CO-4	Execute/Create the Project or field assignment as per the knowledge gained in the course

UNIT – I: INTRODUCTION (10 hours)

Networked wireless sensor devices, Applications: Habitat Monitoring, Smart Transportation, Key design challenges. **Network deployment:** Structured versus randomized deployment, Network topology, Connectivity. Introduction to cloud system, Sensor Cloud Systems, Challenges in Sensor Cloud Systems.

UNIT – II: LOCALIZATION AND WIRELESS CHARACTERISTICS (10 hours)

Localization: issues & approaches, Coarse-grained & Fine-grained node localization, Network-wide localization. **Wireless characteristics:** Basics, Wireless link quality, Radio energy considerations, SINR capture model for interference.

UNIT – III: MEDIUM-ACCESS AND SLEEP SCHEDULING (10 hours)

Issues in designing MAC protocol for WSNs, Classification of MAC Protocols, Energy efficiency in MAC protocols, Asynchronous sleep techniques, Sleep-scheduled techniques. Classification of Energy Management Schemes

Sleep-based topology control: Constructing topologies for connectivity, constructing topologies for coverage.

UNIT – IV: ROUTING AND INTEGRATION OF SENSOR & CLOUD SYSTEM (10 hours)

Routing: Metric-based approaches, Routing with diversity, Multi-path routing, Lifetime-maximizing energy-aware routing techniques, Geographic routing. Data-centric routing, Data-gathering with compression, Querying, Data-centric storage and retrieval, The database perspective on sensor networks.

TEXT BOOKS:

1. Wireless Sensor Networks: Technology, Protocols, and Applications: Kazem Sohraby, Daniel Minoli, Taieb Znati , Wiley Inter Science.
2. Networking Wireless Sensors: Bhaskar Krismachari, Cambridge University Press

REFERENCE BOOKS:

1. Wireless Sensor Networks: Architectures and Protocols: Edgar H. Callaway, Jr. Auerbach Publications, CRC Press.
2. Wireless Sensor Networks: Edited by C.S Raghavendra, Krishna M, Sivalingam, TaiebZnati , Springer.
3. Distributed Sensor Networks: A Multiagent Perspective, Victor Lesser, Charles L. Ortiz, and MilindTambe , Kluwer Publications.
4. Wireless Sensor Networks: An Information Processing Approach- by Feng Zhao, Leonidas Guibas , Morgan Kaufmann Series in Networking 2004.

CLOUD COMPUTING

Course Code	**CS 54E2**	**L-P-T-Cr.:**	4	0	0	4	**Semester:**	**IV**
Category:	Program Elective Course							
Prerequisite:	**Basic Computer Network**							
Learning Objective:	Fundamental of cloud Computing, Management of cloud services, virtualization, storage and optimization.							

CO-1	Remember and understand the basic concepts/Principles of Cloud Computing
CO-2	Analyze the Various Concepts to understand them through case studies
CO-3	Apply the knowledge in understanding practical problems
CO-4	Execute/Create the Project or field assignment as per the knowledge gained in the course

UNIT – I: **(10 hours)**

Networked wireless sensor devices, Applications: Habitat Monitoring, Smart Transportation, Key design challenges. **Network deployment:** Structured versus randomized deployment, Network topology, Connectivity. Introduction to cloud system, Sensor Cloud Systems, Challenges in Sensor Cloud Systems.

UNIT – II: **(08 hours)**

Reliability, availability and security of services deployed from the cloud: Performance and scalability of services, tools and technologies used to manage cloud services deployment; Cloud Economics: Cloud Computing infrastructures available for implementing cloud-based services.

UNIT – III: **(12 hours)**

Cloud infrastructures: public, private, hybrid. Service provider interfaces; Saas, Paas, Iaas. VDC environments; concept, planning and design, business continuity and disaster recovery principles. Managing VDC and cloud environments and infrastructures.

Storage strategy and governance: security and regulations. Designing secure solutions; the considerations and implementations involved. Securing storage in virtualized and cloud environments.

UNIT – IV: **(10 hours)**

Architecture of storage: analysis and planning. Storage network design considerations; NAS and FC SANs, hybrid storage networking technologies (iSCSI, FCIP, FCoE), design for storage virtualization in cloud computing, host system design considerations. Global storage management locations, scalability, operational efficiency. **Global storage distribution:** terabytes to petabytes and greater.

TEXT BOOKS:

1. Greg Schulz, “Cloud and Virtual Data Storage Networking”, Auerbach Publications [ISBN: 978-1439851739], 2011.
2. Gautam Shroff, “Enterprise Cloud Computing Technology Architecture Applications”, Cambridge University Press; 1 edition, [ISBN: 978-0521137355], 2010.

REFERENCE BOOKS:

1. Toby Velte, Anthony Velte, Robert Elsenpeter, “Cloud Computing, A Practical Approach” McGraw-Hill Osborne Media; 1 edition [ISBN: 0071626948], 2009.
2. Dimitris N. Chorafas, “Cloud Computing Strategies” CRC Press; 1 edition [ISBN: 1439834539],2010.
3. EMC, “Information Storage and Management” Wiley; 2 editions [ISBN: 9780470294215],2012.

MACHINE LEARNING

Course Code	**CS 54E3**	**L-P-T-Cr.:**	4	0	0	4	**Semester:**	**IV**
Category:	Program Elective Course							
Prerequisite:	**Fundamental of computer science and mathematics**							

Learning Objective:

- To introduce concepts of learning.
- To know decision tree learning and various learning methods.

CO-1	Remember and understand the basic concepts/Principles of Machine Learning
CO-2	Analyze the Various Concepts to understand them through case studies
CO-3	Apply the knowledge in understanding practical problems
CO-4	Execute/Create the Project or field assignment as per the knowledge gained in the course

UNIT – I: **(12 hours)**

General Introduction: Learning Problems, Choosing Training experience/Target Function, Representation of the target function, issues in machine learning.

Concept Learning: Concept learning task-Inductive Learning, Concept Learning as search, FIND-S algorithm, version spaces, The List then Eliminate algorithm, Representation of version spaces, The Candidate Elimination algorithm, Inductive bias.

UNIT – II: **(12 hours)**

Decision Tree Learning: Decision tree representation, ID3 Learning algorithm, Entropy, Information gain, over fitting, reduced error pruning, Rule-post pruning.

Bayesian Learning: Bayes' Theorem and concept Learning, Bayes optimal classifier, Bayesian Belief Network.

UNIT – III: **(12 hours)**

Instance based Learning: Introduction, k-Nearest Neighbors Learning algorithm, distance weighted nearest neighbors learning algorithm, case-based reasoning, lazy learner and eager learner.

Learning Set of Rules: Sequential covering algorithm, First Order Inductive Learning (FOIL), Induction as inverted deduction, Inverting resolution (First order resolution), Generalization, theta-subsumption and entailment, PROGOL.

UNIT – IV: **(12 Hours)**

Analytical Learning: Inductive vs Analytical Learning, Prolog-EBG, Combining inductive and analytical learning

TEXT BOOKS:

1. Tom M. Mitchell, Machine Learning, Mac Graw Hill

REFERENCE BOOKS:

1. Christopher M. Bishop, Machine Learning and Pattern Recognition, Springer

INTRODUCTION TO BIG DATA ANALYTICS

Course Code	**CS 54E4**	**L-P-T-Cr.:**	4	0	0	4	**Semester:**	**IV**
Category:	Program Elective Course							
Prerequisite:	**Basic Computer Network, Cloud Computing and Database system.**							

Learning Objective:

- Fundamentals of Big data
- Fundamental of MapReduce
- Information Management and Data Privacy and Ethics

CO-1	Remember and understand the basic concepts/Principles of Big data and Analytics
CO-2	Analyze the Various Concepts to understand them through case studies
CO-3	Apply the knowledge in understanding practical problems
CO-4	Execute/Create the Project or field assignment as per the knowledge gained in the course

UNIT – I: **(10 hours)**

Introduction: Big data and its importance, a flood of mythic "startup" proportions, big data is more than merely big why now? a convergence of key trends, a wider variety of data, the expanding universe of unstructured data, **Industry examples of big data:** Digital marketing and the online world, the right approach, cross channel lifecycle marketing.

UNIT – II: **(14 hours)**

Big Data Technology: The elephant in the room: Hadoop's parallel world, old vs. new approaches. Data discovery: Work the way people's minds work, open-source technology for big data analytics, the cloud and big data, predictive analytics moves into the limelight, a brief history of Hadoop, apache Hadoop and the Hadoop ecosystem.

MapReduce: Analyzing the data with Hadoop, map and reduce, java MapReduce, scaling out, data flow, combiner functions, running a distributed MapReduce job, Hadoop streaming, the Hadoop distributed file system, the design of HDFS, HDFS concepts, blocks, name nodes and data nodes, HDFS federation, HDFS high, availability, the command, line interface, basic file system operations, Hadoop file systems.

UNIT – III: **(12 hours)**

Information Management: The big data foundation, big data computing platforms, big data computation, more on big data storage, big data computational limitations, big data emerging technologies.

Business analytics: The last mile in data analysis, geospatial intelligence will make your life better, consumption of analytics, from creation to consumption.

Visualizing: How to make it consumable? organizations are using data visualization as a way to take immediate action.

UNIT – IV: **(12 hours)**

Data Privacy and Ethics: The privacy landscape, the great data grab isn't new, preferences, personalization, and relationships, rights and responsibility, playing in a global sandbox, conscientious and conscious responsibility, privacy may be the wrong focus can data be anonymized? Balancing for counter intelligence.

TEXT BOOKS:

1. Michael Minelli, Michele Chambers, Big Data,Big Analytics, Wiley Publications, 2013
2. 2. Tom White, Hadoop: The Definitive Guide, 3/e, O'Reilly Publications, 2012.

REFERENCE BOOKS:

1. Bill Franks Taming, The Big Data Tidal Wave, 1/e, Wiley, 2012.
2. Frank J. Ohlhorst, Big Data Analytics, 1/e, Wiley, 2012

INFORMATION & CYBER SECURITY

Course Code **CS 54E5** **L-P-T-Cr.:** 4 0 0 4 **Semester:** **IV**

Category: Program Elective Course

Prerequisite: **Computer Network**

Learning Objective:

- Explain the objectives of information security & importance and application of each of confidentiality, integrity, authentication and availability
- Understand various cryptographic algorithms.
- Understand the basic categories of threats to computers and networks
- Describe the enhancements made to lPv4 by IPSec
- Understand Intrusions and intrusion detection
- Discuss the fundamental ideas of public-key cryptography.
- Generate and distribute a PGP key pair and use the PGP package to send an encrypted e-mail message & Discuss Web security and Firewalls

CO-1	Remember and understand the basic concepts/Principles of Information & Cyber Security
CO-2	Analyze the Various Concepts to understand them through case studies
CO-3	Apply the knowledge in understanding practical problems
CO-4	Execute/Create the Project or field assignment as per the knowledge gained in the course

UNIT – I: **(10 hours)**

Attacks on Computers and Computer Security: Introduction, The need for security, Security approaches, Principles of security, Types of Security attacks, Security services, Security Mechanisms, A model for Network Security Cryptography: Concepts and Techniques: Introduction, plain text and cipher text, substitution techniques, transposition techniques, encryption and decryption, symmetric and asymmetric key cryptography, stenography, key range and key size, possible types of attacks.

UNIT – II: **(14 hours)**

Symmetric key Ciphers: Block Cipher principles & Algorithms (DES, AES, Blowfish), Differential and Linear Cryptanalysis, Block cipher modes of operation, Stream ciphers, RC4, Location and placement of encryption function, Key distribution Asymmetric key Ciphers: Principles of public key cryptosystems1 Algorithms (RSA, Diffie-Hellman, ECC), Key Distribution.

UNIT – III: **(12 hours)**

Message Authentication Algorithms and Hash Functions: Authentication requirements, Functions, Message authentication codes, Hash Functions, Secure hash algorithm, Whirlpool, HMAC, CMAC, Digital signatures, knapsack algorithm Authentication Applications: Kerberos, X.509 Authentication Service, Public — Key Infrastructure, Biometric Authentication.

UNIT – IV: **(12 hours)**

E-Mail Security: Pretty Good Privacy, S/MIME

IP Security: lP Security overview, IP Security architecture, Authentication Header, encapsulating security payload, Combining security associations, key management.

Web Security: Web security considerations, Secure Socket Layer and Transport Layer Security, Secure electronic transaction.

Intruders, Virus and Firewalls: Intruders, Intrusion detection, password management, Virus and related threats, Countermeasures, Firewall design principles, Types of firewalls.

TEXT BOOKS:

1. Cryptography and Network Security: William Stallings, Pearson Education,4t" Edition.
2. Cryptography and Network Security: Atul Kahate, McGraw Hill, 2nd Edition.

REFERENCE BOOKS:

1. Cryptography and Network Security: C K Shyamala, N Harini, Dr T R Padmanabhan, Wiley India, 1' Edition.
2. Cryptography and Network Security: Forouzan Mukhopadhyay, McGraw Hill, 2 Edition.
3. Information Security, Principles and Practice: Mark Stamp, Wiley India.
4. Principles of Computer Security: WM. Arthur Conklin, Greg White, TMH.
5. Introduction to Network Security: Neal Krawetz, CENGAGE Learning.