M.Sc. PROGRAMME IN COMPUTER SCIENCE

(Effective from Session 2026-27)

(Batch: 2026-2028)



SAMBALPUR UNIVERSITY

JYOTI-VIHAR, BURLA, SAMBALPUR, ODISHA-768019

SAMBALPUR UNIVERSITY

SEMESTER-WISE COURSE STRUCTURE FOR THE TWO YEARS P.G PROGRAMME IN UNIVERSITY P.G DEPARTMENT AND COLLEGES UNDER SAMBALPUR UNIVERSITY

TO BE EFFECTIVE FROM
2025-26 BATCH: 2025-27
(Ref: letter No: 4873/Acd.-I Dated 21.08.2023)

For (Science/ Humanities/Social Sciences/ Commerce)				
Semester	**Core Course Credit**	**Additional Course**	**Additional Course Credit**	**Total Credit**
First	20	**AECC I**: Environmental Studies and Disaster management	2	22
Second	20	Inter Dept. Course (IDC) or open elective	3	23
Third	20	**AECC II**: Entrepreneurship Development	2	22
Fourth (including project of 4 credit)	20	MOOCs one paper	3	23
TOTAL	**80**		**10**	**90**
	Total credit for 2 years course = 90 Credits			
	Furthermore, following non - credit course will be taken by the students			
1. Yuva Sanskar		2. N.C.C/N.S. S/Sports/Performing Arts/Yoga (Of which one has to be opted)		

SAMBALPUR UNIVERSITY

JYOTI VIHAR, BURLA-768019

Two-Year M.Sc. Degree Course in

Computer Science

M.Sc. Computer Science

(Effective from Academic year 2026-2028)

Semester Structure

First Semester - August to December.

Second Semester - January to June.

Third Semester - July to December.

Fourth Semester - January to June.

Course Structure of M.Sc. Computer Science Programme

Semester - I	Semester - II	Semester - III	Semester - IV
Data Structure Using C	Python Programming	Design and Analysis of Algorithms	Industrial Project Problem Identification & Literature Review
Object Oriented Programming using C++	Database Management System	Elective – 1 A. Internet of Things B. Machine Learning	Industrial Project Model Development & Experimentation
Operating System	Data Communication & Computer Networking	Elective – 2 A. Digital Image Processing B. Information & Cyber Security	Project Report Evaluation
Computer Organization and Architecture	Discrete Mathematics	Software Engineering	Project Viva Voce
Lab on Data Structure	Lab on Python Programming	Lab on Design & Analysis of Algorithms	Mid-Semester Seminar I
Lab on C++	Lab on Database Management System	Lab on Elective-1/Elective-2	Mid-Semester Seminar II
Env. Studies & Disaster Management	Inter Dept. Course (IDC) or Open Elective	Entrepreneurship Development	MOOC/Alternative Course
Yuva Sanskar	NCC/NSS/Sports/Performing Arts/Yoga (of which one has to be opted)		

Semester - I

Course Code	Course Title	No. of Credits	Hours/Week
CS-411	Data Structure Using C	4	4
CS-412	Object Oriented Programming using C++	4	4
CS-413	Operating System	4	4
CS-414	Computer Organization and Architecture	4	4
CS-415	Lab on Data Structure	2	2
CS-416	Lab on C++	2	2
CS-417	Env. Studies & Disaster Management	2	2
	Yuva Sanskar	0	0
Total Credits	**22**	-	-

Semester – II

Course Code	Course Title	No. of Credits	Hours/Week
CS-421	Python Programming	4	4
CS-422	Database Management System	4	4
CS-423	Data Communication & Computer Networking	4	4
CS-424	Discrete Mathematics	4	4
CS-425	Lab on Python Programming	2	2
CS-426	Lab on Database Management System	2	2
CS-427	Inter Dept. Course (IDC) or Open Elective	3	3
	NCC/NSS/Sports/Performing Arts/Yoga (of which one has to be opted)	0	0
Total Credits	**23**	-	-

Semester – III

Course Code	Course Title	No. of Credits	Hours/Week
CS-511	Design and Analysis of Algorithms	4	4
CS-512	Elective – 1 A. Internet of Things B. Machine Learning	4	4
CS-513	Elective – 2 A. Digital Image Processing B. Information & Cyber Security	4	4
CS-514	Software Engineering	4	4
CS-515	Lab on Design & Analysis of Algorithms	2	2
CS-516	Lab on Elective-1/Elective-2	2	2
CS-517	Entrepreneurship Development	2	2
Total Credits	**22**	-	-

Semester – IV

Course Code	Course Title	No. of Credits	Hours/Week
CS-521	Industrial Project Problem Identification & Literature Review	4	4
CS-522	Industrial Project Model Development & Experimentation	4	4
CS-523	Project Report Evaluation	4	4
CS-524	Project Viva Voce	4	4
CS-525	Mid-Semester Seminar I	2	2
CS-526	Mid-Semester Seminar II	2	2
CS-527	MOOCs one paper	3	3
Total Credits	**23**	-	-

Total Credits of the Course

Semesters	Sem - I	Sem - II	Sem - III	Sem - IV	TOTAL
Credits	22	23	22	23	**90**

NOTE:

- Furthermore, the following non-credit course will be taken by the students
  1. Yuva Sanskar in 1st Semester
  2. NSS/ NCC/ Sports/ Performing Arts/ Yoga (any one) in 2nd Semester.
- Students have to opt for an Inter-Department Course (IDC) offered by other departments of the university in the 2nd Semester.

- Students will apply in prescribed form their preference for NCC/NSS/sports/Performing Arts/Yoga at the beginning of the session, i.e., in the 1st Semester.

- The students will take one MOOC course other than the core courses present in the MSc Computer Science syllabus. The student may opt, according to their preference, in consultation with the HOD from the 1st Semester, and submit documents in support of undertaking the MOOCs course to the Department. Furthermore, the student must submit the completion certificate of the opted MOOC course before filling out the exam form in the 4th Semester.

FIRST SEMESTER
Course Title: DATA STRUCTURE USING C
Course Code: CS-411
No. of Credits - 4

Course Outcomes

CO1	Apply C programming constructs (variables, operators, loops, and conditional statements) to write structured code.
CO2	Develop modular programs using arrays, strings, functions, pointers, and structures.
CO3	Implement and analyze basic data structures including stacks, queues, linked lists, trees, and graphs.
CO4	Solve problems using algorithmic approaches such as expression conversion/evaluation, tree/graph traversals, and topological sorting.

Unit -I
Introduction: Overview of C, Constants, Variables and Data Types, Operators and Expressions: types of operators, associativity, precedence, Decision Making and Control Structure: if, if..else, if..else ladder, switch statement, for loop, while loop, do..while, break, continue.

Unit -II
Arrays: Definition, 1-D, 2-D arrays, initialization and access of elements, strings, functions, pointers, Structure,

Unit – III
Introduction to data structures: storage structure for arrays, sparse matrices, Stacks, and Queues: representation and application. Linked lists: Single-linked lists, linked list representation of stacks and Queues. Operations on polynomials, double-linked lists, and circular lists.

Unit – IV
Infix to Postfix conversion, Postfix expression evaluation. Trees: Tree terminology, Binary tree, Binary search tree, General tree, Complete Binary Tree representation, Tree traversals, operations on Binary tree-expression Manipulation. Graphs: Graph terminology, Representation of graphs, path matrix, BFS (breadth first search), DFS (depth first search), topological sorting

Text Books:

1. E. Balaguruswamy, "Programming in ANSI C", 8th Edition, 2019, McGraw Hill Education
2. Gilberg and Forouzan: "Data Structure- A Pseudo code approach with C" by Thomson publication
3. "Data structure in C" by Tanenbaum, PHI publication / Pearson publication.

Reference Books:

1. B. Kernighan & Dennis Ritchie, "The C Programming Language"
2. "Fundamentals of data structure in C" Horowitz, Sahani & Freed, Computer Science Press.
3. "Fundamental of Data Structure" (Schaums Series) Tata-McGraw-Hill.

FIRST SEMESTER
Course Title: OBJECT ORIENTED PROGRAMMING USING C++
Course Code: CS-412
No. of Credits - 4

Course Outcomes

CO1	Apply fundamental programming concepts and control structures using C++.
CO2	Design and implement programs using arrays, structures, and object-oriented constructs like classes and objects.
CO3	Utilize advanced features such as constructors, destructors, function overloading, and operator overloading effectively.

CO4	Implement inheritance, polymorphism, virtual functions, and exception handling to build modular and maintainable applications.

UNIT-I

Object-Oriented programming paradigm, Basic data types, Tokens, Keywords, Identifiers, Variables, Operators: Arithmetic, Relational, Logical, Assignment, Ternary, Bitwise, Unary Operators, Expressions and statements, Input and Output in C++, manipulators with parameters, Flow of control - if, if-else, while, do-while, for loop, Switch, break and continue.

UNIT-II

Single and multidimensional arrays. Character array, string variables, reading multiple lines, arrays of strings, specifying the structure, accessing structure members, array of structures. Classes and objects, Class declaration, Data member and Member functions, private and public members, scope resolution operator

UNIT-III

Inline Functions, passing objects as arguments, returning objects, Function overloading, Friend function, constructors, destructors, overloaded constructors, Types of Constructors, operator overloading: Unary Operator, Binary Operator

UNIT-IV

Inheritance: Derived Class and Base Class, accessing base class members, the protected access specifier, abstract base class, single, multilevel, multiple inheritance, and ambiguity in multiple inheritance. Polymorphism, pointers, Virtual base class, Virtual functions & dynamic binding, Exception handling.

Books Recommended:

1. E. Balguruswamy,"Object-Oriented programming with C++",TMH, 5th Edition, 2011
2. R.Lafore,"Object-oriented programming in TURBOC++", Galgotia, 1st Edition, 1997
3. Y.P.Kanetkar,"Let us C++", BPB publication , 2nd Edition, 2015
4. Stanley B. Lippman, Josée Lajoie, "C++ Primer", Pearson Education, 4th Edition

FIRST SEMESTER

Course Title: OPERATING SYSTEM

Course Code: CS-413

No. of Credits - 4

Course Outcomes

CO-1	Describe the functions, types, and services of an operating system.
CO-2	Analyze and implement process scheduling and deadlock handling strategies.
CO-3	Understand memory management schemes, including paging, segmentation, and virtual memory.
CO-4	Explain file system structures and evaluate disk scheduling algorithms.

UNIT-I

Evolution of Operating Systems: Types of operating systems, Operating system services. The process management: Process concept, Process Control Block, Short-term, medium-term, and long-term Schedulers, Context Switch, Operation on Processes.

UNIT-II

CPU Scheduling: CPU Scheduler, preemptive Scheduling, Scheduling criteria, Scheduling Algorithms- FCFS, SJF, Priority, and round robin scheduling. Deadlocks: Necessary conditions for deadlock, Resource Allocation graph, Methods of handling Deadlocks, Deadlock prevention & avoidance, Deadlock Detection and deadlock Recovery

UNIT-III

Memory Management: Logical versus physical Address, Swapping, Single partition allocation, multiple partition allocation, paging, segmentation. Demand paging, Virtual memory, page fault, Page replacement algorithms- FIFO, Optimal, LRU.

UNIT-IV

File Systems: general model of a file system, Disk structure, disk scheduling: FCFS scheduling, SSTF scheduling, SCAN scheduling, C-SCAN scheduling, LOOK scheduling.

Books Recommended:

1. Silberschatz& P.B. Galvin, "Operating Systems Concepts", Addison - Wesley, 9th Edition, 2012
2. Tanenbaum," Modern Operating System", Pearson Education, 3rd Edition, 2015
3. William Stallings ,"Operating Systems: Internals and Design Principles", 6th Edition
4. Dhananjay Dhamdhere, "Operating System a Concept Based Approach", 3rd Edition, 2017

FIRST SEMESTER
Course Title: COMPUTER ORGANIZATION AND ARCHITECTURE
Course Code: CS-414
No. of Credits - 4

Course Outcomes

CO-1	Explain the functional organization of a computer and apply Boolean algebra for logic design.
CO-2	Design and analyze combinational and sequential circuits including adders, multiplexers, and flip-flops.
CO-3	Understand instruction formats, addressing modes, and compare RISC and CISC architectures.
CO-4	Describe memory hierarchy, I/O systems, and access techniques like cache memory and DMA.

UNIT-I:

Basic organization of the computer and block-level description of the functional units as related to the execution of a program. Digital logic gates, Boolean algebra, Boolean Function, and simplification, Simplification of Boolean function using K-Map. Canonical form of Boolean functions: SOP and POS.

UNIT-II:

Combinational and Sequential Circuits- Adders, Subtractors, Carry-Look-Ahead (CLA) adder, encoders and decoders, multiplexer and demultiplexer. Flip Flops- RS, JK, D, T, and Master Slave Flip Flops. Registers and counters.

UNIT-III:

CPU Organization: Instruction codes, Computer Instructions, Instruction Cycles, Execution of Instructions, Instruction Formats (Zero, One and Two address instructions), Addressing Modes, Discussions about RISC versus CISC architectures.

UNIT-IV:

Memory and IO access: Memory maps, Read Write operation, Programmed IO and interrupt driven IO, DMA controller and data transfer. Memory organization: static and dynamic memory, Memory Hierarchy, cache memory and its access techniques; Virtual memory.

Books Recommended:

1. M. Morris Mano, "Computer System Architecture", PHI, 3rd Edition
2. William Stallings , "Computer Organization and Architecture", Pearson / PHI, 9th Edition, 2013

3. B.RAM , "Fundamentals of Microprocessor and Microcontrollers", Dhanpat Rai Publication, 2010 Edition

FIRST SEMESTER
Course Title: LAB ON DATA STRUCTURE
Course Code: CS-415
(Based on Paper CS-411: Data Structure Using C)
No. of Credits - 2

Course Outcomes

CO1	Implement and manipulate **basic data structures** such as arrays, stacks, and queues using C.
CO2	Write modular programs to perform operations on **linked lists**, including insertion, deletion, and traversal.
CO3	Construct and traverse **trees and graphs**, and implement algorithms such as DFS, BFS, and expression evaluation.
CO4	Analyze and apply appropriate data structures for **problem-solving** in programming tasks.

FIRST SEMESTER
Course Title: LAB ON C++
Course Code: CS-416
(Based on Paper CS-412: Object Oriented Programming using C++)
No. of Credits – 2

Course Outcomes

CO-1	Write and execute basic C++ programs using variables, control structures, and I/O operations.
CO-2	Implement OOP features including **classes, objects, constructors, destructors, and function overloading**.
CO-3	Apply **operator overloading, inheritance, and polymorphism** in practical coding scenarios.
CO-4	Develop modular and robust C++ applications using **virtual functions, dynamic binding, and exception handling**.

FIRST SEMESTER
Course Title: ENV. STUDIES & DISASTER MANAGEMENT
Course Code: CS-417
No. of Credits - 2

SECOND SEMESTER
Course Title: PYTHON PROGRAMMING
Course Code: CS-421
No. of Credits - 4

Course Outcomes

CO-1	Understand Python's syntax, variable assignments, and write basic programs in script and interactive mode.
CO-2	Work with standard Python data types and apply operators and built-in functions for problem-solving.
CO-3	Write structured code using **control flow**, **loops**, and **functions**, including lambda and default parameters.
CO-4	Implement **object-oriented programming concepts**, manage exceptions, and perform basic **file operations**.

Unit-I

Introduction: Installation, First Python Program: Interactive Mode Programming, Script Mode Programming; Identifiers, Reserved Words, Lines and Indentation, Multi-Line Statements, Quotation & Comments; Assigning Values to Variables, Multiple Assignment.

Unit-II

Standard Data Types: Numbers, Strings, Lists, Tuples, Dictionary, Set; Basic Operators: Arithmetic, Comparison, Assignment, Bitwise; Python Numbers & Mathematical functions.

Unit-III

Python statements and Loops: if, if-else, While, for loops, break, continue, pass; Functions: Definition, call, positional and keyword parameter. Default parameters, variable number of arguments, lambda function.

Unit-IV

Object Oriented Programming: classes and objects – Inheritance, Polymorphism; Error handling & Exceptions - try, except and raise, File Processing: reading and writing files.

Books Recommended:

1. T. Budd, Exploring Python, TMH, 1st Ed, 2011
2. Allen Downey, Jeffrey Elkner, Chris Meyers , How to think like a computer scientist : learning with Python , Freely available online.2012

SECOND SEMESTER
Course Title: DATABASE MANAGEMENT SYSTEM
Course Code: CS-422
No. of Credits - 4

Course Outcomes

CO1	Explain database fundamentals, DBMS architecture, data models, and ER diagrams.
CO2	Formulate complex queries using **relational algebra, calculus, and SQL**.
CO3	Apply **normalization techniques** to design well-structured relational databases.
CO4	Understand **transaction management, concurrency control**, and distributed database concepts.

UNIT-I

Database: Definition, Database Management, Structure, Limitations of traditional file processing systems, Advantages and disadvantages of DBMS, Users of DBMS. Components of DBMS, Data Independence and 3-tier architecture and View of Data. Data Model: Hierarchical, Network, Relational, Entity Relationship model: Concepts of entity, entity set, attributes, E-R diagram.

UNIT-II

Relational Query Languages, Relational Algebra and operations, Tuple and Domain Relational Calculus, Functional Dependency, Super key, Candidate Key, Primary Key, Alternate and foreign keys. Strong and weak entities. Integrity constraints. SQL- Languages: DDL, DML and DCL.

UNIT-III

Relational Database Design: Normal Forms (1NF, 2NF, 3NF, BCNF), Decomposition, Dependency Preservation and Lossless Join. Codd's rules. Object Oriented Databases Need for OODBMS, Object structure: Class, polymorphism, encapsulation, inheritance.

UNIT-IV

The distributed databases -Motivation for Distributed Database, Distributed Database concepts. Transaction Management: Transaction Management and Concurrency Control Transaction: Properties (ACID), states, Commit, Rollback Concurrency: Control, Lost update problems, Locks, two phase locking, serialization.

Books Recommended:

1. Silberschatz, Korth, Sudarshan, "Database System Concepts", McGraw Hill, 4th Edition 2002
2. Elmasari, Navathe," Fundamentals of Database Systems", Pearson, 7th Edition, 2016
3. Ramakrishnan, "Database Management Systems", Mcgraw Higher Ed, 3rd Edition, 2014
4. Atul Kahate,"Introduction to Database Management Systems", Pearson Education, 1st Edition,2004

SECOND SEMESTER
Course Title: DATA COMMUNICATION & NETWORKING
Course Code: CS-423
No. of Credits - 4

Course Outcomes

CO1	Understand networking fundamentals, types, topologies, and physical layer concepts.
CO2	Explain and apply **digital and analog transmission techniques**, line coding, modulation, and multiplexing.
CO3	Implement and analyze **data link layer protocols** and **LAN technologies**.
CO4	Describe the working of **network, transport, and application layer protocols** including IP, TCP, and DNS.

UNIT-I:

Overview of Data Communications and Networking. Networking - Needs and Advantages, Network Types- Client, Server and Peers. Network Topology-Bus, Ring, Star and Mesh Topologies. Mode of data communication. Physical Layer: Analog and Digital, data and signals, Signal properties, Data Rate Limits, Transmission Impairment.

UNIT-II

Digital Transmission: Line coding, Pulse Code Modulation(PCM), Transmission mode. Analog Transmission: Modulation of Digital Data (ASK, FSK, PSK, QAM), Modulation of Analog signals (AM, FM, PM). Multiplexing: FDM, WDM and TDM, Transmission Media: Guided Media, Unguided media (wireless)

UNIT-III

Data Link Layer: Flow and error Control, Stop-and-wait ARQ. Go-Back-N -ARQ, Selective Repeat ARQ, Point-to-Point Access: Point-to Point Protocol, Multiple-Access: Random Access, Controlled Access, Channelization. Local Area Network: Ethernet, Token bus (IEEE- 802.4), Token ring (IEEE-802.5).

UNIT-IV

Network Layer: Internetworking, IPv4 addressing and Subnetting, IPv6 Address, Internet Protocol (IP), Transport Layer: Process to process Delivery, Client Server Paradigm, Port Number, UDP, TCP congestion control. Application Layer: Domain Name System (DNS), Electronic Mail, and File transfer (FTP), Telnet, HTTP and WWW,

Books Recommended:

1. B.A. Forouzan,"Data Communication and Networking", TMH, 4th Edition, 2006
2. A.S. Tannenbaum, "Computer Networks", Pearson, 5th Edition, 2012
3. William Stallings , "Data and Computer Communications", Pearson, 8th Edition, 2009

SECOND SEMESTER
Course Title: DISCRETE MATHEMATICS
Course Code: CS-424
No. of Credits - 4

Course Outcomes

CO1	Apply concepts of **logic, sets, relations, lattices**, and Boolean algebra in computational problems.
CO2	Solve **counting problems, recurrence relations**, and use generating functions effectively.
CO3	Understand and apply **group theory, ring theory**, and related algebraic structures.
CO4	Analyze and apply **graph and tree concepts** and compute basic **statistical measures**.

UNIT - I

Fundamentals of logic, Prepositional equivalences, Predicates and Quantifiers, Mathematical Induction. Sets, Set operations, Properties of binary relations, Equivalence relations and partitions, Partial ordering relations and lattices, Properties of lattices, Distributive and Complemented lattices, Boolean algebra,

UNIT - II

The basics of counting, Permutations and Combinations, Recurrence relations, Solving Recurrence relations, Generating functions

UNIT - III

Groups, Subgroups, Cosets and Lagrange's Theorem, Codes and Group codes, Homomorphism and Normal subgroups, Isomorphism, Ring, Integral Domains and Fields.

UNIT-IV

Introduction to graphs, Graph terminology, Representing graphs and Graph isomorphism, Euler and Hamilton paths, Introduction to trees, Applications of trees.

Frequency Distribution, Measures of Central Tendencies, Dispersion, Skewness, Kurtosis, Mathematical Expectation.

Books Recommended:

1. Kenneth H. Rosen, "Discrete Mathematics & Its Application", TMH, 7th Edition, 2011
2. C. L. Liu, "Elements of Discrete Mathematics", TMH, 2nd Edition, 2000
3. BernardiKolman, Robert C. Busby, Sharon Ross, "Discrete Mathematical Structure", PHI, 6th Edition, 2008
4. S.P.Gupta,"Statistical Methods", S.Chand& Sons, 2011 Edition

SECOND SEMESTER
Course Title: LAB ON PYTHON PROGRAMMING
Course Code: CS-425
(Based on Paper- CS-421: PYTHON PROGRAMMING)
No. of Credits - 2

Course Outcomes

CO1	Write and execute Python programs using variables, expressions, and basic I/O operations.
CO2	Implement decision-making, looping constructs, and define user-defined functions.
CO3	Apply object-oriented principles, handle exceptions, and perform file operations.
CO4	Develop modular programs that solve practical problems using Python scripting techniques

SECOND SEMESTER
Course Title: LAB ON DBMS
Course Code: CS-426
(Based on Paper- CS-S-422: DATABASE MANAGEMENT SYSTEM)
No. of Credits - 2

Course Outcomes

CO1	Use SQL to create tables, define constraints, and manipulate data.
CO2	Write queries involving **joins, subqueries, aggregation**, and views.
CO3	Apply **normalization principles** and enforce **integrity constraints**.
CO4	Demonstrate transaction management, including commit, rollback, and concurrency controls.

SECOND SEMESTER
Course Title: Inter-Department Course (IDC)
Course Code: CS-427
No. of Credits - 3
Appendix - B

Students have to opt for an Inter-Department Course (IDC) offered by other departments of the university.

THIRD SEMESTER
Course Title: DESIGN AND ANALYSIS OF ALGORITHMS
Course Code: CS-511
No. of Credits - 4

Course Outcomes

CO1	Analyze the time and space complexity of algorithms using **asymptotic notations** and recurrence relations.
CO2	Apply and evaluate **sorting, searching**, and **divide-and-conquer algorithms** including Merge Sort, Quick Sort, and Heap Sort.
CO3	Design and implement algorithms using **greedy and dynamic programming** approaches for problems like shortest paths, knapsack, and LCS.
CO4	Understand and analyze **NP-complete problems**, approximation algorithms, and basic principles of **computability**.

UNIT-I

Algorithms and Complexity: Introduction to Algorithm, Asymptotic Notations and Basic Efficiency Classes (Big O, θ, Ω, ω, little o) in analysis of algorithms. Growth of functions, Recurrences: Recursive algorithms, Substitution method, Recurrence Tree method, Master method.

UNIT-II

Sorting and Searching Techniques: Bubble Sort, Insertion Sort, Sequential Search, Binary Search, Depth First Search and Breadth First Search, Divide and Conquer Paradigm: problem solving, Algorithm design and Complexity of Merge Sort, Quick Sort. Heap Sort: Heaps, Maintaining Heap property, Building a heap, Heap Sort algorithm, Priority Queues.

UNIT-III

Greedy Techniques: Prim's Algorithm, Kruskal's Algorithm, Dijkstra's and Bellman Ford Algorithm, Huffman Trees, Fractional Knapsack problem. Dynamic Programming Paradigm : Floyd-Warshall Algorithm, Matrix Chain Multiplication Problem, Longest Common Subsequence Problem, 0/1 Knapsack Problem.

UNIT-IV

Travelling Salesman Problem and its State Space Search Tree. Introduction to Computability: Complexity Classes, P, NP, NP-Hard, NP-Completeness and Reducibility, Approximation Algorithms: Vertex Cover Problem.

Books Recommended

1. Horowitz E. &Sahni S and S.Rajasekaran,"Fundamentals of Computer Algorithms", Universities Press, 2nd Edition, 2008
2. Aho, Hopcroft & Ullman, "The Design and Analysis of Computer Algorithms", Pearson, 1st Edition, 1974
3. T.H.Coremen, C.E Leiserson, R.L.Rivest and C.Stein," Introduction to Algorithms", PHI, 3rd Edition, 2009.

THIRD SEMESTER
Course Title: ELECTIVE - 1
Course Code: CS-512
No. of Credits - 4
Appendix - C

THIRD SEMESTER
Course Title: ELECTIVE - 2
Course Code: CS-513
No. of Credits - 4
Appendix - D

THIRD SEMESTER

Course Title: SOFTWARE ENGINEERING
Course Code: CS-514
No. of Credits - 4

Course Outcomes

CO1	Understand and apply **software life cycle models, project planning**, and estimation techniques like COCOMO.
CO2	Analyze and design software using **DFDs, UML diagrams**, and object-oriented principles.
CO3	Demonstrate knowledge of **coding practices, software testing techniques**, and maintenance processes.
CO4	Evaluate **software quality standards (SEI CMM, ISO 9000)** and understand the role of CASE tools and software reuse.

UNIT-I

Introduction: Software Life cycle Models, project planning, metrics (LOC, Functional Point), estimation techniques: empirical and heuristic (COCOMO), Scheduling: Work Breakdown Structure, Critical Path Method, PERT chart, Risk management, Requirements analysis and specification: Requirement gathering and analysis, Software Requirement Specification.

UNIT-III

Software Design: Overview of Design, Cohesion and coupling, Function oriented software design: Data Flow Diagrams, Object-oriented Design: Object modeling using UML, use case diagram, class diagram, Interaction diagrams: activity diagram, Package, Component and Deployment diagram, State chart Diagram.

UNIT-IV

Coding and Testing: Coding, Code review, software documentation, testing, unit testing, black box and white box testing, integration and system testing. Maintenance: Characteristics, maintenance tasks, types of maintenance, software maintenance process models, Estimation of maintenance cost.

UNIT-IV

Software quality: SEI CMM and ISO-9000. Software reliability and fault-tolerance. Computer-aided software engineering (CASE): Characteristics of CASE Tools, Architecture of CASE environment, Software reuse.

Books Recommended:

1. Rajib Mall," Fundamentals of Software Engineering", PHI, 4th edition.
2. R.S. Pressman, "Software Engineering Practitioner's Approach", TMH, 7th Edition, 2010
3. Pankaj Jalote, "A Concise Introduction to Software Engineering", Springe

THIRD SEMESTER
Course Title: LAB ON DESIGN AND ANALYSIS OF ALGORITHMS
(Based on Paper- CS-511: DESIGN AND ANALYSIS OF ALGORITHM)
Course Code: CS-515
No. of Credits - 2

Course Outcomes

CO1	Implement and evaluate **recursive and divide-and-conquer algorithms** such as Merge Sort and Quick Sort.
CO2	Apply **greedy and dynamic programming techniques** to solve optimization problems like shortest paths, knapsack, and LCS.
CO3	Design and simulate **graph algorithms** such as DFS, BFS, Prim's, and Kruskal's.
CO4	Analyze the complexity of algorithms and explore **NP-completeness and approximation solutions** in practice.

THIRD SEMESTER

Course Title: LAB ON ELECTIVE-1/ELECTIVE-2
(Based on Paper Elective -I / Elective -II)
Course Code: CS-516
No. of Credits - 2

THIRD SEMESTER
Course Title: ENTREPRENEURSHIP DEVELOPMENT
Course Code: CS-517
No. of Credits - 2

FOURTH SEMESTER
Course Title: INDUSTRIAL PROJECT PROBLEM IDENTIFICATION & LITERATURE REVIEW
Course Code: CS-521
No. of Credits - 4

Students will undertake an internship in a software industry/organization to identify a real-world software problem based on organizational requirements. Activities include requirement analysis, interaction with stakeholders, and study of existing software solutions within the organization. A comprehensive literature review needs to be conducted to analyze current technologies, methodologies, and systems, leading to the identification of research or implementation gaps.

Course Outcomes

CO1	Identify and formulate a real-world software industry problem with defined requirements.
CO2	Perform a literature survey on existing software systems, frameworks, and technologies.
CO3	Analyze current solutions to identify limitations and research gaps.
CO4	Prepare a structured technical review report following standard documentation practices.

FOURTH SEMESTER
Course Title: INDUSTRIAL PROJECT MODEL DEVELOPMENT & EXPERIMENTATION
Course Code: CS-522
No. of Credits - 4

During the internship in a software industry/organization, students will design and develop a software-based solution addressing the identified problem. Work includes system architecture design, coding, database integration, and the use of relevant frameworks and tools. Testing, debugging, and performance evaluation are to be carried out using real or organizational data to ensure practical applicability and efficiency of the solution.

Course Outcomes

CO1	Design and develop a software solution based on the identified requirements.
CO2	Implement appropriate algorithms, data structures, and software frameworks.
CO3	Perform testing, debugging, and performance evaluation of the software system.
CO4	Analyze results and optimize the system for efficiency, scalability, and usability.

FOURTH SEMESTER
Course Title: PROJECT REPORT EVALUATION
Course Code: CS-523
No. of Credits - 4

Students will prepare a comprehensive project report based on work carried out during the internship in a software industry/organization. The report will include document requirement analysis, system design, implementation details, testing procedures, and results. The document should emphasize clear technical documentation, inclusion of diagrams and performance analysis, and adherence to academic and professional standards.

Course Outcomes

CO1	Prepare a detailed software project report covering all development phases.
CO2	Present system design, architecture, and implementation details clearly.
CO3	Document testing results, performance metrics, and analysis effectively.
CO4	Apply proper citation, version control, and ethical documentation practices.

FOURTH SEMESTER
Course Title: PROJECT VIVA VOCE
Course Code: CS-524
No. of Credits - 4

Evaluation will be based on the student's internship project work completed in a software industry/organization. Students will present and defend their software solution, explaining system design, implementation, tools used, and outcomes achieved. Assessment focuses on technical understanding, problem-solving ability, and effectiveness in communicating project work.

Course Outcomes

CO1	Demonstrate in-depth knowledge of the developed software system.
CO2	Explain system architecture, design choices, and implementation strategies.
CO3	Defend technical decisions and respond to queries with clarity.
CO4	Exhibit effective communication and professional presentation skills.

FOURTH SEMESTER
Course Title: MID-SEMESTER SEMINAR I
Course Code: CS-525
No. of Credits -2

Students will present initial progress of their internship project undertaken in a software industry/organization, including problem identification, requirement analysis, literature review, and proposed system design. Feedback from faculty and peers will help refine the approach and improve the development plan.

Course Outcomes

CO1	Present software problem statement and system requirements clearly.
CO2	Explain proposed architecture and technology stack effectively.
CO3	Incorporate feedback to improve design and development plan.
CO4	Develop technical communication and presentation skills.

FOURTH SEMESTER
Course Title: MID-SEMESTER SEMINAR II
Course Code: CS-526
No. of Credits -2

Students will present progress of their internship project carried out in a software industry/organization, including implementation status, module development, and preliminary results. Demonstration of developed features and discussion of challenges faced during development will be included, along with feedback for further improvement.

Course Outcomes

CO1	Present progress in software development and module implementation.
CO2	Demonstrate working components and system functionalities.
CO3	Analyze challenges in coding, integration, and testing with solutions.
CO4	Communicate technical progress and results effectively.

FOURTH SEMESTER
Course Title: MOOCs
Course Code: CS-527
No. of Credits - 3

Students have to opt for a MOOC course offered on the SWAYAM/ NPTEL platform.

APPENDIX - B
SECOND SEMESTER
Course Title: COMPUTER FUNDAMENTALS
Inter Dept. Course (IDC) or Open Elective
Course Code: IDC(CSA)-429
No. of Credits -3

Course Outcomes

CO1	Understand the **basic components of a computer system**, data representation, and types of software.
CO2	Identify and describe the function of **I/O devices, memory units**, and operating system services.
CO3	Explain the **organization of a computer system**, including CPU, memory, buses, and storage.
CO4	Demonstrate a foundational understanding of **networking, internet services, and basic network security**.

Unit-I

Introduction: Introduction to computer systems, uses, and types.

Data Representation: Number systems and character representation, binary arithmetic.

Software: Introduction, types of software, utility programs

Unit-II

Input and output devices (with connections and practical demo): keyboard, mouse, joystick, scanner, OCR, OMR, bar code reader, web camera, monitor, printer

Computer Memory: Primary, secondary, auxiliary memory, RAM, ROM, cache memory, hard disks, optical disks.

Fundamentals of Operating Systems: Types of OS, Services, process concepts.

Unit-III

Computer Organization and Architecture Fundamentals: CPU, registers, system bus, main memory unit, cache memory.

Computer Networking Fundamentals: Introduction, Types of Networks, Internet, Email, Client-Server, Web Services.

Network Security Fundamentals: The need for security, Types of Security attacks, Security services, Security Mechanisms, A model for Network Security Cryptography.

Recommended Books:

1. A. Goel, Computer Fundamentals, Pearson Education, 2010.
2. P. Aksoy, L. DeNardis, Introduction to Information Technology, Cengage Learning, 2006
3. P. K.Sinha, P. Sinha, Fundamentals of Computers, BPB Publishers, 2007

APPENDIX - C
Elective – I
Course Title: INTERNET OF THINGS
Course Code: CS-512 (A)

No. of Credits -4

Course Outcomes

CO1	Students should be able to design a portable IoT using Arduino/ equivalent boards and relevant protocols.
CO2	Students should be able to develop web services to access/control IoT devices.
CO3	Students should be able to deploy an IoT application and connect to the cloud.
CO4	Students should be able to analyse applications of IoT in real-time scenarios.

UNIT-I

Introduction to Internet of Things: Sensing, Actuation, Basic components of IoT, Applications, Service Oriented Architecture, Basics of associated technologies with IoT (Cloud Computing, WSN, IoV, M2M, CPS, IoE), Challenges in IoT,

UNIT-II

Connectivity: IPv6, RPL ,
Data Protocol: MQTT, CoAP, AMQP
Communication Protocols: IEEE 802.15.4, ZigBee, 6LowPAN, Bluetooth, NFC, RFID

UNIT-III

Implementation of IoT: Introduction to Arduino Programming, Integration of Sensors and Actuators with Arduino, Arduino Libraries, Basics of Embedded C programming for Arduino, Interfacing LED and push button, Basic Networking with ESP8266 Wi-Fi module, Various Wi-Fi library, Introduction to Raspberry Pi, Implementation of IoT with Raspberry Pi

UNIT-IV

Cloud Computing: Recent trends in Computing, NIST visual model, Characteristics, components, service model (SaaS, PaaS, IaaS), Public cloud, private cloud and hybrid clouds, Service management and security, Cloud simulators, Open-source clouds, commercial clouds, IOT Cloud platforms, ThingSpeak API, Interfacing ESP8266 with Web services

Books Recommended:

1. "The Internet of Things: Enabling Technologies, Platforms, and Use Cases", by Pethuru Raj and Anupama C. Raman (CRC Press)
2. "Internet of Things: A Hands-on Approach", by Arshdeep Bahga and Vijay Madisetti (Universities Press)

Elective – I
Course Title: MACHINE LEARNING
Course Code: CS-512 (B)

No. of Credits -4

Course Outcomes

CO1	Develop a solid understanding of the fundamentals of machine learning, including its types and applications.
CO2	Apply various machine learning techniques such as linear regression, logistic regression, and decision trees to train models and make predictions
CO3	Gain proficiency in using support vector machines (SVM) for classification and regression tasks.
CO4	Explore the fundamentals of deep learning and artificial neural networks, including their architecture and activation functions.

Unit-I

The Fundamentals of Machine Learning Understanding Machine Learning,Need and Relevance of Machine Learning, Types of Machine Learning, Supervised Learning, Unsupervised Learning & Reinforcement Learning. Challenges of Machine Learning, Testing and Validation. Classification, MNIST Dataset, Performance Measures, Confusion Matrix, Precision and Recall, Precision/Recall Tradeoff, The ROC Curve, Multiclass Classification, Error Analysis.

Unit-II

Training Models Linear Regression, Gradient Descent, Batch Gradient Descent, Stochastic Gradient Descent, Mini-batch Gradient Descent, Polynomial Regression, Learning Curves, The Bias/Variance Tradeoff, Ridge Regression, Lasso Regression, Early Stopping, Logistic Regression, Decision Boundaries, Softmax Regression, Cross Entropy.

Unit-III

Linear SVM Classification, Soft Margin Classification, Nonlinear SVM Classification, Polynomial Kernel, Gaussian RBF Kernel, SVM Regression, Decision Trees, Training and Visualizing a Decision Tree, Making Predictions, The CART Training Algorithm, Gini Impurity vs Entropy, Regularization Hyperparameters.

Unit-IV

What is Deep Learning? Need Deep Learning? Introduction to Artificial Neural Network (ANN), Core components of neural networks, Multi-Layer Perceptron (MLP), Activation functions, Sigmoid, Rectified Linear Unit (ReLU), Introduction to Tensors and Operations, Tensorflow framework

APPENDIX-D
Elective – II
Course Title: DIGITAL IMAGE PROCESSING
Course Code: CS-513 (A)
No. of Credits -4

Course Outcomes

CO1	Remember and understand the basic concepts/Principles of DIGITAL IMAGE PROCESSING
CO2	Analyze the Various Concepts to understand them through case studies
CO3	Apply the knowledge in understanding practical problems
CO4	Execute/Create the Project or field assignment as per the knowledge gained in the course

UNIT-I

What is Digital Image Processing, Fundamental Steps in Digital Image Processing, Elements of Visual Perception, Image Sampling and Quantization, Some basic relationships between Pixels, Image Enhancement: Gray Level Transformation: Image Negatives, Log Transformations, Histogram Processing: Histogram Equalization, Basics of Spatial Filtering

UNIT-II

Image Transforms; Fourier Transform and their properties, Smoothing Frequency-Domain Filtering: Ideal, Butterworth, Gaussian Low pass Filters, Sharpening Frequency Domain Filtering: Ideal, Butterworth, Gaussian High pass Filters.

UNIT-III

Image Restoration: A model of the Image Degradation/Restoration Process, Noise Models. Restoration in the Presence of Noise Only-spatial Filtering. Estimating the Degradation Function. Inverse Filtering. Minimum Mean Square Error (Wiener) Filtering. Morphological Image Processing: Preliminaries. Dilation and Erosion. Opening and Closing, The Hit-or- Miss Transformation.

UNIT-IV

Image Compression: Fundamentals, Image Compression Models, Elements of Information Theory, Error-Free Compression: Variable length coding, LZW coding, Image Segmentation: Detection of Discontinuity, Edge linking and Boundary Detection, Region based Segmentation.

Books Recommended:

1. R.C.Gonzalez&R.E.Wood, "Digital Image Processing", Addison Wesley.
2. B.Channda&D.Dutta,"Digital Image Processing and Analysis", PHI, 2nd Edition, 2011
3. A.K.Jain, "Fundamentals of Digital Image Processing",Pearson Education, 1st Edition, 2015

Elective – II
Course Title: INFORMATION & CYBER SECURITY
Course Code: CS-513 (B)
No. of Credits -4

Course Outcomes

CO1	Remember and understand the basic concepts/Principles of INFORMATION & CYBER SECURITY.
CO2	Analyze various concepts to gain a deeper understanding through case studies.
CO3	Apply the knowledge to understand practical problems.
CO4	Execute/Create the Project or field assignment as per the knowledge gained in the course.

UNIT – I

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

Symmetric key Ciphers: Block Cipher principles & Algorithms (DES, AES), 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 cryptosystems Algorithms (RSA, Diffie-Hellman, ECC), Key Distribution.

UNIT – III

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

UNIT – IV

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.