BACHELOR
OF
COMPUTER APPLICATION

Specialization

In

Artificial Intelligence & Machine Learning

Programme Code: BCA AI ML
Duration: 3 Years

EFFECTIVE FROM SESSION: 2025-2026

CBCS

Department of Computer Science & Information Technology
Faculty of Engineering & Technology

CHHATRAPATI SHIVAJI MAHARAJ UNIVERSITY
PANVEL, NAVI MUMBAI

	BCA- AI ML 1 Year - Semester I										
Type	**Course Code**	**Course Title**	**Hours/ Week**			**Theory Marks**		**Practical Marks**		**Total Marks**	**Credit**
			L	**T**	**P**	**IA**	**ESE**	**IA**	**ESE**		
DC	CSAB1010	C Programming	3	-	-	30	70	-	-	100	3
DC	CSAB1020	Computer Fundamental & Office Automation	3	1	-	30	70	-	-	100	4
DC	CSAB1030	Introduction to Artificial Intelligence	3	1	-	30	70	-	-	100	4
OE	***	Elective 1	3	-	-	30	70	-	-	100	3
AECC	ENGG1000	English Communication Skills	3	1	-	30	70	-	-	100	4
BS	MTHG1000	Engineering Mathematics - I	3	1	-	30	70	-	-	100	4
DC	CSAB1011	C Programming Lab	-	-	4	-	-	15	35	50	2
DC	CSAB1021	Office Automation Lab	-	-	4	-	-	15	35	50	2
SEC		Soft Skills									
		TOTAL	**18**	**4**	**8**	**180**	**420**	**30**	**70**	**700**	**26**

Elective-1 MGTG1000/ BBAB2030
1. Principle Of Management - I
2. Ethics & Corporate Social Responsibility

Course Title	C Programming	Course Code:CSAB1010	
Teaching Scheme		**Evaluation Scheme**	
Lectures	**3 Hrs Week**	**Internal Assessment Test**	**30 Marks**
Total Credits	**3**	**End-Semester Examination**	**70 Marks**

Course Objectives:

1. It aims to train the student to the basic concepts of the C-programming language and provide exposure to problem-solving through programming.
2. To provide the knowledge of control structures in c programming language.
3. Use and implement data structures like arrays and pointers to obtain solutions.
4. Implement different programming constructs such as structures and decomposition of problems into functions
5. Define and use of strings and file functions in C programming language.

Detailed Syllabus:

Unit-I

Introduction to 'C' Language

History, Structures of 'C' Programming, Function as building blocks, Language Fundamentals: Character set, C Tokens, Keywords, Identifiers, Variables, Constant, Data Types, Comments, Operators, Built-in Operators and function, Console based I/O and related built-in I/O function, Concept of header files, Preprocessor directives.

Unit-II

Control structures

Decision making structures- If, If-else, Nested If -else, Switch. Loop Control structures- While, Do-while, for, nested for loop, other statements- break, continue, goto, exit.

Unit-III

Arrays and Pointers

Arrays: Definition, declaration and initialization of one-dimensional array, accessing array elements, displaying array elements, Two-Dimensional array-declaration and initialization, accessing and displaying, memory representation of array-row major, Column major, Multidimensional array.

Pointers: Definition and declaration, Initialization, indirection operator, address of operator, pointer arithmetic, dynamic memory allocation, arrays and pointers, function and pointers

Unit-IV

Structures and Functions

Structures: Definition and declaration, Variables initialization, accessing fields and structure operations, Nested structures, Union-Definition and declaration, Differentiate between Union and structure.

Functions: Basic types of function, Declaration and definition, Function call, Types of function, Parameter passing- Call by value, Call by reference, Scope of variables, Storage classes, Recursion.

Unit-V

Strings and File Handling

Strings: Definition, declaration and initialization of strings, standard library functions: -strlen (), strcpy( ), strcat( ), strcmp( ), Implementation without using standard library functions.

File handling: Definition of Files, Opening modes of files, Standard function: -fopen( ), fclose( ), feof( ), fseek( ), rewind( ), Using text files:- fgetc( ), fputc( ), fprintf( ), fscanf**( ).**

Text /Reference Books:

1. Let us C-Yashwant Kanetkar.
2. Programming in C- Balguruswamy
3. The C Programming Lang., Pearson Ecl – Dennis Ritchie
4. Structured programming approach using C-Forouzah &Ceilberg Thomson learning publication.
5. Pointers in C – Yashwant Kanetkar
6. How to solve it by Computer – R. G. Dromy
7. Introduction to algorithms – Cormen, Leiserson, Rivest, Stein

Course Outcomes:

After completion of this subject students will be able to

1. CO1: Acquire very basic knowledge about c programming.
2. CO2: Apply appropriate control structures to solve problems.
3. CO3: Students get detailed information about Arrays & Pointers and its application in C programming.
4. CO4: Create Structures and user defined function to solve problems in C programming.
5. CO5: Apply functions towards performing operations on Files.

Course Title	Computer Fundamental & Office Automation	**Course Code:CSAB1020**	
Teaching Scheme		**Evaluation Scheme**	
Lectures / Tutorial	**3/1 Hrs Week**	**Internal Assessment Test**	**30 Marks**
Total Credits	**4**	**End-Semester Examination**	**70 Marks**

Course Objectives:

1. To provide the foundational concepts of computer hardware, software, programming languages, operating systems, peripherals, etc. along with how to get the most value and impact from computer technology.
2. To make understand the beginners as well as advanced learners the concept of step wise problem solving.
3. To make students at a moderate level of expertise in the knowledge of computer basics from where a student can take himself to the next level.
4. To help students create the worksheets and perfom basic calculations
5. To understand how to use software packages in day-to-day activities and learn the essential and use of operating systems.

Detailed Syllabus:

Unit-I

Introduction of computer, Characteristics of Computers, Block diagram of computer, Block diagram of computer, Types of computer and features, Types of programming languages, Data Organization, Drives, Files, Directories, Types of Memory, Secondary storage devices, I/O devices, Number Systems.

Unit-II

Algorithm and Flowcharts
Algorithm - Definition of Algorithm, Characteristics of Algorithm, Advantages and disadvantages, Examples. Flowchart- Definition, Define symbols of flowchart, Advantages and disadvantages, Examples.

Unit-III

Windows Operating Environment
Features of MS –Windows, Control Panel, Taskbar, Desktop, Windows Application, Icons, Windows Accessories, Notepad, Paintbrush. Editors and Word Processors: Basic Concepts, MS-Word, Introduction to desktop publishing.

Unit-IV

Spreadsheets and Database packages
Purpose, usage, commands, MS-Excel, Creation of files in MS-Access, Switching between applications, MS -PowerPoint.

Unit-V

DOS: Introduction to Operating System, DOS - History Files and Directories, Internal and External Commands, Batch Files, Types of O.S.
Linux: File system, Linux Commands, Permission, I/O redirection, VI Editor.

Text /Reference Books:

1. Fundamentals of computers - V. Rajaraman - Prentice- Hall of India
2. Microsoft Office 2007 Bible - John Walkenbach,Herb Tyson,Faithe Wempen,cary N.Prague, Michael R.groh, Peter G.Aitken, and Lisa a.Bucki -Wiley India pvt.ltd.
3. The complete reference Linux - Richard Retersen - Tata McGraw - Hill Edition
4. A Conceptual Guide to OpenOffice.org 3 - R. Gabriel Gurley- CreateSpace Independent Publishing Platform, 2008
5. Introduction to Information Technology - Alexis Leon, Mathews Leon, and Leena Leon, Vijay Nicole Imprints Pvt. Ltd., 2013.
6. Computer Fundamentals - P. K. Sinha Publisher: BPB Publications

Course Outcomes:

After completion of this subject students will be able to

1. CO1: Remember the basic knowledge about computers.
2. CO2: Understand & Apply knowledge of flowcharts with algorithm concepts.
3. CO3: Utilize Windows operating environment with additional knowledge of word processors.
4. CO4: Understand the basic database applications & create spreadsheets which will be used in daily work life.
5. CO5: Execute the Linux operating system commands.

Course Title	Introduction to Artificial Intelligence	**Course Code:CSAB1030**	
Teaching Scheme		**Evaluation Scheme**	
Lectures/ Tutorial	**3 / 1 Hrs Week**	**Internal Assessment Test**	**30 Marks**
Total Credits	**4**	**End-Semester Examination**	**70 Marks**

Course Objectives:

1. Understand the fundamental concepts of Artificial Intelligence & non-AI solution.
2. Develop understanding of how informed and uninformed search algorithms play vital role in problem solving.
3. Apply knowledge of reasoning and knowledge representation for solving real world problems.
4. Develop understanding of reasoning and learning in AI.
5. Explore learning probabilistic reasoning in AI.

Detailed Syllabus:

Unit-I

Introduction to Artificial Intelligence: Definition, history, and evolution of AI, Characteristics and goals of AI ,Applications of AI in real-world domains (healthcare, robotics, finance, education),Foundations of AI: logic, reasoning, knowledge, learning, Intelligent agents: structure, types, and environments.

Unit-II

Problem Solving and Search Techniques: Problem formulation in AI,Search strategies:Uninformed search: Breadth-First Search (BFS), Depth-First Search (DFS) &Informed search: Greedy search, Game playing: Minimax algorithm and Alpha-Beta pruning, Constraint Satisfaction Problems (CSP)

Unit-III

Knowledge Representation and Reasoning: Introduction to knowledge representation (KR),Logical representations: Propositional and First-Order Logic, Semantic networks, frames, ontologies, Inference techniques: forward chaining, backward chaining, resolution, Rule-based systems and knowledge-based agents

Unit-IV

Planning and Learning in AI:Components of planning systems,Types of planning: state-space planning, goal stack planning,Introduction to machine learning in AI context,Learning types: Supervised, Unsupervised,Applications of learning in AI systems (adaptive behavior, personalization)

Unit-V

AI Applications and Ethics: Real-world applications of AI: natural language processing, computer vision, robotics, AI in smart systems: recommendation engines, chatbots, virtual assistants, Limitations and challenges in AI implementation, Ethical issues: data privacy, transparency, job displacement, Future of AI and responsible AI development

Assignments

Assignment 1: Image Classification using Google Teachable Machine

Aim: To create a simple image recognition model using a no-code tool.

Tool: Google Teachable Machine

Task:

- Train an image classifier using webcam or uploaded images
- Create two or more categories (e.g., thumbs up / thumbs down)
- Export and test the model with real-time input

Learning Outcome: Understand supervised learning and AI classification basics.

Reference : **https://www.youtube.com/watch?v=jhGm4KDafKU**

Assignment 2: Maze Solver using MIT Scratch (Visual Search Simulation)

Aim: To visually simulate search-based problem solving using a block-based platform.

Tool: MIT Scratch

Task:

- Create a simple maze in Scratch
- Implement logic using blocks for navigating from start to finish
- Simulate BFS/DFS using conditional steps and loops

Learning Outcome: Learn how AI uses search to solve problems

Reference : **https://www.youtube.com/watch?v=22Dpi5e9uz8**

Assignment 3: Rule-Based Expert System using CLIPS

Aim: To create a basic expert system for decision-making using rule-based logic.

Tool: CLIPS

Task:

- Define facts and rules (e.g., suggest clothing based on weather conditions)
- Use the inference engine to deduce and display decisions

Learning Outcome: Learn rule-based reasoning and knowledge representation.

Reference:

https://www.youtube.com/watch?v=i7Z76TgN9W0&list=PL84b1TR9XxFki1gO30JL8Mm_iVZygQKe5&index=5

Text /Reference Books:

1. Rich **E** & Knight K, **Artificial Intelligence,** Tata McGraw Hill, Edition 3
2. Russell S & Norvig P, *Artificial Intelligence: A Modern Approach*, Prentice Hall.
3. Dan W. Patterson, Artificial Intelligence & Expert Systems, Pearson Education with Prentice Hall India. Indian Edition.
4. Artificial Intelligence and Intelligent Systems by Padhy, Oxforfd University Press,
5. John Paul Mueller, Luca Massaron, "Artificial Intelligence for dummies", 2nd edition, Wiley, 2021.
6. Daeyeol Lee, "Birth of Intelligence: From RNA to Artificial Intelligence", 1st edition, Oxford University Press, 2020.

Course Outcomes:

CO-1: Understand and relate the goals of Artificial Intelligence and AI & non-AI solution.

CO-2: Analyze and illustrate how informed and uninformed search algorithms play vital role in problem solving.

CO-3: Acquire knowledge of reasoning and knowledge representation for solving real world problems

CO-4: Acquire knowledge about various method using forward, backward reasoning, and reinforcement, ANN-based Learning.

CO-5: Explain learning probabilistic reasoning in AI using Bayesian networks simple decision making.

Elective 1			
Course Title	Principle Of Management - I	**Course Code:MGTG1000**	
Teaching Scheme		**Evaluation Scheme**	
Lectures	**3 Hrs Week**	**Internal Assessment Test**	**30 Marks**
Total Credits	**3**	**End-Semester Examination**	**70 Marks**

Course Objectives:

1. This subject is primarily meant for all those students of Management who aspire to enter the corporate world.
2. To make a handy reference guide for professionals who have started their career and want to become successful managers in future.

Detailed Syllabus:

Unit-I

Overview of Management
Definition, Management, Role of managers, Evolution of Management thought, Organization and the environmental factors, Trends and Challenges of Management in Global Scenario.

Unit-II

Planning
Nature and purpose of planning, Planning process, Types of plans, Objectives, managing by objective (MBO) Strategies, Types of strategies, Policies, Decision Making, Types of decision, Decision Making Process, Rational Decision Making

Unit-III

Organizing
Nature and purpose of organizing, Organization structure, Formal and informal groups in organization, Line and Staff authority, departmentation, Span of control, Centralization and Decentralization, Delegation of authority, Staffing, Selection and Recruitment, Orientation, Career Development, Career stages, Training, Performance Appraisal

Unit-IV

Directing
Creativity and Innovation, Motivation and Satisfaction, Motivation Theories, Leadership Styles, Leadership theories, Communication, Barriers to effective communication, Organization Culture, Elements and types of culture, Managing cultural diversity.

Unit-V

Controlling

Process of controlling, Types of control, Budgetary and non-budgetary control Q techniques , Managing Productivity , Cost Control , Purchase Control , Maintenance Control , Quality Control , Planning operations.

Text /Reference Books:

1. Stephen P. Robbins and Mary Coulter, 'Management', Prentice Hall of India, 8th edition. 2. Charles W L Hill, Steven L McShane, 'Principles of Management', Mcgraw Hill Education, Special Indian Edition, 2007.
2. Hellriegel, Slocum & Jackson, ' Management - A Competency Based Approach', Thomson South Western, 10th edition, 2007.
3. Harold Koontz, Heinz Weihrich and Mark V Cannice, 'Management - A global
4. Stephen A. Robbins & David A. Decenzo & Mary Coulter, "Fundamentals of Management" 7th Edition, Pearson Education, 2011.
5. Robert Kreitner & Mamata Mohapatra, "Management", Biztantra, 2008.
6. Harold Koontz & Heinz Weihrich "Essentials of management" Tata Mc Graw Hill, 1998.
7. Tripathy PC & Reddy PN, "Principles of Management", Tata McGraw Hill, 1999.

Course Outcomes:

CO1: Understand Management Fundamentals

CO2: Developing Analytical Skills to planning process

CO3: Enhancing organizing skills and acquire leadership Abilities

CO4: Promote Ethical and Social Responsibility

CO5: Prepare for professional roles for controlling in organization.

Course Title	Ethics & Corporate Social Responsibility	Course Code:BBAB2030	
Teaching Scheme		**Evaluation Scheme**	
Lectures	**3 Hrs Week**	**Internal Assessment Test**	**30 Marks**
Total Credits	**3**	**End-Semester Examination**	**70 Marks**

Course Objectives:

1. Gain familiarity with fundamental concepts of ethics, corporate governance, and CSR, including sustainability, stakeholder theory, and social justice.
2. Examine the impact of global business practices on ethics and CSR, including cultural differences and regulatory environments.

Detailed Syllabus:

Unit-I

Introduction: Values-Concept, types and formation of values, Values of Indian Managers, Ethics-development of ethics, ethical decision making and decision-making process, relevance of ethics and values in business.

Unit-II

Management of Ethics: Management process and ethics, managerial performance, ethical issues, ethos of Vedanta in management, Code of Ethics, Benefits of Ethical codes, AIMA Code of conduct for professional managers.

Unit-III

Corporate Governance: Concept, need to improve corporate governance standards, Features of good governance, Role played by regulators to improve corporate governance, accounting standards and corporate governance, corporate disclosure, insider trading.

Unit-IV

Corporate Social Responsibility & Consumer Protection: Corporate responsibility of business: employees, consumers and comm. Moduley, Corporate Governance, Code of Corporate Governance, Consumerism, unethical issues, in sales, marketing and technology.

Unit-V

Understanding Success: Definitions of success, Principles for competitive success, prerequisites to create blueprint for success. Successful stories of business gurus.

Text /Reference Books:

1. Bhanumurthy K V: Ethics and Social Responsibility of Business, Pearson Education India.
2. Kaur, Tripat; Values & Ethics in Management, Galgotia Publishers. ● Manuel G Velasquez : Business ethics- concepts and cases Pearson.
3. Kaur, Tripat; Values & Ethics in Management, Galgotia Publishers.
4. Chakraborty, S.K.; Human values for Managers
5. Dr. F.C. Sharma Business Values & Ethics, Shree Mahavir Book Depot (Publisher)

Course Outcomes:

CO1 Define types of ethics.
CO2 Understand the concept of Business Ethics.
CO3 Use different concepts of ethics.
CO4 Analyse reasons to follow workplace Ethics.
CO5 Recommend Ethics in Advertising & Marketing.

Course Title	English Communication Skills	**Course Code:ENGG1000**	
Teaching Scheme		**Evaluation Scheme**	
Lectures / Tutorial	**3 Hrs Week**	**Internal Assessment Test**	**30 Marks**
Total Credits	**4**	**End-Semester Examination**	**70 Marks**

Course Objectives:

1. To acquaint the students with appropriate language skills with the purpose of improving the existing ones-LSRW.
2. To make the learners understand the importance and effective use of non-verbal communication.
3. To make the learner proficient in public speaking and presentation skills.
4. To guide and teach the students to utilize the principles of professional business and technical writing for effective communication in the global world.

Detailed Syllabus:

Unit-I

Communication and Communication Process Introduction to Communication, Forms and functions of Communication, Barriers to Communication ((linguistic and semantic, psychological, physical, mechanical, cultural), and overcoming them, Types of communication: verbal and non-verbal communication. Reading: Introduction to Reading, Barriers to Reading, Types of Reading: Skimming, Scanning, Fast Reading, Strategies for Reading, Comprehension. Listening: Importance of Listening, Types of Listening, and Barriers to Listening.

Unit-II

Writing Skills, Reading Skills & Listening Skills

Features of Good Language, Technical Style of writing, Writing Emails and it's etiquettes, Technical Reports: Report Writing: Types, Format and Structure of reports.

Unit-III

Letter Writing

Types of letters: Job application letter, complaint letter, enquiry letter, reply to enquiry, sales letter, Essential and non-essential parts of letters, formats of letters.

Unit-IV

Grammar

Types of sentences, Antonyms and Synonyms, Use of Auxiliaries and Modal Auxiliaries, Synonyms and Antonyms, Pairs of confused words, Common Errors in sentences

Unit-V

Soft Skills Body language, Teamwork and skills, Decision making ability, Negotiation skills and Interview skills.

Dialogues Writing and Speaking

Greeting someone and responding to greet, thanking someone and responding to thanks, making inquiry and responding to enquiry on telephone, making request and responding to request.

Text /Reference Books:

1. Communication in Organizations by Dalmar Fisher, Jaico Publishing House
2. Communication Skills by Meenakshi Raman & Sangeeta Sharma,
3. Oxford University Press
4. Business Correspondence & Report-writing by R.C. Sharma& Krishna Mohan, Tata McGraw- Hill Education
5. Effective Technical Communication by Ashraf Rizvi, Tata McGraw-Hill.
6. Technical Writing & Professional Communication for non-native speakers of English by Thomas N. Huckin & Leslie A. Olsen, McGraw –Hill.
7. Mastering Communication by Nicky Stanton, Palgrave Master Series

Course Outcomes:

After completion of this subject students will be able to

CO1: Understand and evaluate information they listen to and express their ideas with greater clarity.

CO2: Comprehend language effectively along the various channels of communication in a business organization.

CO3: Apply the Communication through result-oriented writing and reading techniques both within and outside the organization.

CO4: Speak convincingly before an audience with the help of an expanded vocabulary and enhanced digital content.

CO5: Develop soft skills, dialog writing and speaking skills to understand technical description and instructions.

Course Title	Engineering Mathematics - I	**Course Code:MTHG1000**	
Teaching Scheme		**Evaluation Scheme**	
Lectures / Tutorial	**3 / 1 Hrs Week**	**Internal Assessment Test**	**30 Marks**
Total Credits	**4**	**End-Semester Examination**	**70 Marks**

Course Objectives:

1. To make the students familiarize with concepts and techniques in Calculus, Complex number and Matrices.
2. The aim is to equip them with the techniques to understand advanced level mathematics and its applications that would enhance analytical thinking power, useful in their disciplines

Detailed Syllabus:

Unit-I

Review on matrices: Definition of matrix, types of matrices, Algebra of matrices, Adjoint of matrix, inverse of matrix, Unitary & Orthogonal matrices, Echelon form, Rank of a matrix, Normal form, PAQ normal form. System of homogeneous & non homogeneous equations, Conditions of their consistency & Inconsistency & solutions. Solution of system of linear algebraic equations, by (1) Gauss Elimination Method (2) Guass Jordan Method (3) Jacobi iteration (4) Gauss Seidel Method

Unit-II

Complex Numbers

Definition of Complex number, Algebra of complex number, Representation of complex number on complex plane, D'Moivre's Theorem., Powers and roots of Exponential & Trigonometric functions., Expansion of $sin^n\theta$, $cos^n\theta$ in terms of sines and cosines of multiples of θ and Expansion of sinnθ, cosnθ in powers of sinθ, cosθ . Circular functions of complex numbers and Hyperbolic functions, Inverse Circular and Inverse Hyperbolic functions, Logarithmic functions. Separation of real and Imaginary parts of all types of Functions

Unit-III

Numerical Integration

Numerical Integration‑Different types of operators such as shift, forward, backward difference and their relation. Interpolation, Newton Interpolation, Integration by (a) Trapezoidal (b) Simpson's 1/3rd (c) Simpson's 3/8th rule.

Unit-IV

Partial Differential Equation

Partial derivatives of first and higher order, total differentials, differentiation of composite and implicit functions, Euler's Theorem on Homogeneous functions with two and three independent variables (with proof). Deductions from Euler's Theorem

Unit-V

Application of Partial Differentiation, Indeterminate forms and curve fitting
Maxima and Minima of a function of two independent variables, Indeterminate forms, L-Hospital rule, Fitting of curves by least square method for line, parabola and exponential.

Text /Reference Books:

1. A text book of Applied Mathematics, P.N.Wartikar and J.N.Wartikar,Vol – I and –II by Pune Vidyarthi Graha.
2. Higher Engineering Mathematics, Dr.B.S.Grewal, Khanna Publication
3. Advanced Engineering Mathematics, Erwin Kreyszig, Wiley EasternLimited,9thEd.
4. Matrices by Shanti Narayan.
5. Numerical by S.S.Sastry, Prentice Hall

Course Outcomes:

After completion of this subject students will be able to

CO1: Recall the principles of basic operations of matrices.

CO2: Illustrate the concepts of complex numbers to engineering problems.

CO3: Solve the problems using different integration operators.

CO4: Apply the basic principles of partial differentiation.

CO5: Apply concepts of partial differentiation (maxima and minima), expansion of functions as an application of successive differentiation.

Course Title	C Programmin g Lab	**Course Code:CSAB1011**	
Teaching Scheme		**Evaluation Scheme**	
Lectures	**4 Hrs/Week**	**Internal Assessment Test**	**15 Marks**
Total Credit	**2**	**End-Semester Examination**	**35 Marks**

Course Objectives:

1. To understand the fundamentals of programming concepts and structured programming techniques using the C language.
2. To develop the ability to write efficient, modular, and reusable code using functions, loops, conditionals, and arrays in C.
3. To introduce the concepts of pointers, memory management, and dynamic allocation for efficient programming.
4. To enable students to implement data structures such as strings, structures, and unions using C language features.
5. To prepare students for problem-solving and logical thinking through practical implementation of algorithms and real-world programming exercises.

Tools: Turbo C++ / VS Code Editor / Dev C

1. Write a C program to display “This is my first C Program.
2. Write a C program to add two numbers and display its sum.
3. Write a C program to multiply two numbers (4 and 5) and display its product.
4. Write a C program to calculate area and circumference of a circle.
5. Write a C program to perform addition, subtraction, division and multiplication of two numbers. Write a Program to Check Whether a Number is Prime or not.
6. Write a program to find the largest and smallest among three entered numbers and also display whether the identified largest/smallest number is even or odd.
7. Write a program to check whether the entered year is a leap year or not (a year is leap if it is divisible by 4 and divisible by 100 or 400.)
8. Write a program to find the factorial of a number.
9. Write a program to check if the number is Armstrong or not. (Hint: A number is Armstrong if the sum of cubes of individual digits of a number is equal to the number itself).
10. Write a program to print day names using switch cases.
11. Write a program to determine whether the input character is capital or small letter, digits or special

symbol.

12. Write a program to reverse a given integer.
13. Write a program to print numbers in reverse order with a difference of 2.
14. Write a program to print the sum of digits of a number using a for loop.
15. Write a program to check whether a number is Palindrome or not.
16. Write a program to generate Fibonacci series.
17. Write a Program to Search an element in an array.

18. Write a Program to find the largest and smallest element in Array.
19. Write a program to add, subtract, multiply and divide two integers using user defined type function with return type.
20. Write a program to calculate the sum of the first 20 natural numbers using a recursive function.
21. Write a program to swap values of two variables using a pointer.
22. Write a program to add two numbers using pointers.
23. C Program to list all files and sub-directories in a directory.
24. C Program to count the number of lines in a file.
25. C Program to print contents of file.
26. C Program to copy contents of one file to another file.

Course Outcomes:

After completion of this subject students will be able to

CO1: Evaluate different approaches and algorithms to solve problems efficiently.

CO2: Create solutions using structured programming techniques to address various programming challenges.

CO3: Develop programs to perform tasks like identifying the largest and smallest numbers, checking for palindromes, generating Fibonacci series, etc.

CO4: Apply problem-solving skills and critical thinking to develop robust and efficient solutions.

CO5: Apply higher-order thinking skills to optimize code, improve algorithm efficiency, and handle edge cases effectively.

Course Title	Office Automation Lab	**Course Code:CSAB1021**	
Teaching Scheme		**Evaluation Scheme**	
Lectures	**4 Hrs/Week**	**Internal Assessment Test**	**15 Marks**
Total Credit	**2**	**End-Semester Examination**	**35 Marks**

Course Objectives:

1. To introduce the basic concepts of computer hardware, software, and operating systems and enable students to operate a computer system effectively.
2. To develop proficiency in using essential applications such as word processors, spreadsheets, and presentation tools through hands-on practical sessions.
3. To familiarize students with file management and internet browsing, including safe and efficient use of email and search engines.
4. To build foundational skills in basic programming or scripting (e.g., using Scratch, Python, or shell commands) as part of computer problem-solving.
5. To promote digital literacy and responsible computing practices, including cyber hygiene, data backup, and understanding of basic security features.

Suggested list of experiments:

MS-WORD

1. Text Manipulation: Write a paragraph about your institution and change the font size and type, Spell check, Aligning and justification of Text
2. Bio data: Prepare a Bio-data.
3. Find and Replace: Write a paragraph about yourself and do the following. Find and Replace - Use Numbering Bullets, Footer and Headers.
4. Tables and manipulation: Creation, Insertion, Deletion (Columns and Rows). Create a mark sheet.
5. Mail Merge: Prepare an invitation to invite your friends to your birthday party. Prepare at least five letters.

MS-EXCEL

1. Data Sorting-Ascending and Descending (both numbers and alphabets)
2. Mark list preparation for a student
3. Individual Pay Bill preparation.
4. Invoice Report preparation. 5. Drawing Graphs. Take your own table.

MS-POWERPOINT

1. Create a slideshow presentation for a seminar.
2. Preparation of Organization Charts
3. Create a slideshow presentation to display percentage of marks in each semester for all students

A. Use a bar chart (X-axis: Semester, Y-axis: % marks).
B. Use different presentation templates and different transition effects for each slide.

MS-ACCESS

1. Create a new database, save it on the desktop and name it "School Database" 2. Create a Table in the School Database with the following:

Field Name	Data Type	Field Size
ID	Number	10
Name	Text	15
Surname	Text	15
Telephone	Number	Number Long Integer
Date of Birth	Date/Time	Medium Date
Stipend	Currency	Currency
Foreigner	Yes/No	Yes/No

Make the "ID Number" Field as the Primary Key& Save the table as "Student's Table".

3. Open the "Students Table" and enter 5 complete records, Sort the table in ascending order by surname. Delete the last Record you have entered.
4. Create a Report: Use the Report Wizard to create a report having the following requirements:

a. Select the LastName field from the Author table.
- Select the Title and Price fields from the Book table.
- Select the PubName field from the Publisher table.
- View the data by Publisher.
- Add a grouping level using Last Name.
- Sort the report by the Title field in ascending order.
- Choose Stepped layout and Portrait orientation.
- Type Book List as the report's title.

b. Switch to Layout view and adjust each column's width as necessary.

Course Outcomes:

After completion of this subject students will be able to

CO1: Apply database design principles to create tables, define fields, and set relationships.

CO2: Analyze data requirements and design database structures to efficiently store and retrieve information.

CO3: Create and populate tables with relevant data, applying sorting and filtering techniques as needed.

CO4: Demonstrate proficiency in creating reports using Report Wizard, incorporating grouping, sorting, and formatting options.

CO5: Apply critical thinking to ensure database efficiency, data accuracy, and report readability.

	BCA- AI ML 1 Year - Semester II										
Type	**Course Code**	**Course Title**	**Hours/ Week**			**Theory Marks**		**Practical Marks**		**Total Marks**	**Credit**
			L	**T**	**P**	**IA**	**ESE**	**IA**	**ESE**		
DC	CSAB2010	Data Structure using C	3	1	-	30	70	-	-	100	4
DC	CSAB2020	Database Management System (DBMS)	3	1	-	30	70	-	-	100	4
DC	CSAB2040	Introduction to R programming	3	1	-	30	70	-	-	100	4
AECC	EVSG2000	Environmental Studies (EVS)	3		-	30	70	-	-	100	3
DC	CSAB2030	Software Engineering	3		-	30	70	-	-	100	3
BS	MTHG2010	Element of Statistics	3	1		30	70			100	4
DC	CSAB2011	Data Structure using C Lab	-	-	4	-	-	15	35	50	2
DC	CSAB2021	Database Management System Lab	-	-	4	-	-	15	35	50	2
SEC		Aptitude Building									
		TOTAL	**18**	**4**	**8**	**180**	**420**	**30**	**70**	**700**	**26**

Course Title	Data Structure using C	Course Code:CSAB2010	
Teaching Scheme		**Evaluation Scheme**	
Lectures / Tutorial	**3 / 1 Hrs Week**	**Internal Assessment Test**	**30 Marks**
Total Credits	**4**	**End-Semester Examination**	**70 Marks**

Course Objectives:

1. Understand and remember algorithms and its analysis procedure.
2. Introduce the concept of data structures through ADT including List, Stack, and Queues.
3. To design and implement various data structure algorithms.
4. To introduce various techniques for representation of the data in the real world.
5. To develop applications using data structure algorithms.
6. Compute the complexity of various algorithms.

Detailed Syllabus:

Unit-I

Introduction

Introduction: Basic Terminology, Elementary Data Organization, built in Data Types in C. Algorithm, Efficiency of an Algorithm, Time and Space Complexity, Asymptotic notations: Big Oh, Big Theta and Big Omega, Time-Space trade-off. Abstract Data Types (ADT).

Unit-II

Arrays and Linked List

Arrays: Definition, Single and Multidimensional Arrays, Representation of Arrays: Row Major Order, and Column Major Order, Derivation of Index Formulae for 1-D,2-D,3-D and Application of arrays, Sparse Matrices and their representations. Linked lists: Array Implementation and Pointer Implementation of Singly Linked Lists, Doubly

Linked List, Circularly Linked List, Operations on a Linked List. Insertion, Deletion, Traversal, Polynomial Representation and Addition Subtraction & Multiplications of Single variable & Two variables Polynomial.

Unit-III

Searching and Sorting

Searching: Concept of Searching, Sequential search, Index Sequential Search, Binary Search. Concept of Hashing & Collision resolution Techniques used in Hashing. Sorting: Insertion Sort, Selection, Bubble Sort, Quick Sort, Merge Sort, Heap Sort and Radix Sort.

Unit-IV

Stacks and Queues

Stacks: Abstract Data Type, Primitive Stack operations: Push & Pop, Array and Linked Implementation of Stack in C, Application of stack: Prefix and Postfix Expressions, Evaluation of postfix expression, Queues: Operations on Queue: Create, Add, Delete, Full and Empty, Circular queues, Array and linked implementation of queues in C, Dequeue and Priority Queue.

Unit-V

Graphs

Graphs: Terminology used with Graph, Data Structure for Graph Representations: Adjacency Matrices, Adjacency List, Adjacency. Graph Traversal: Depth First Search and Breadth First Search, Connected Component, Spanning Trees, Minimum Cost Spanning Trees: Prims and Kruskal algorithm. Transitive Closure and Shortest Path algorithm: Warshal Algorithm and Dijikstra Algorithm.

Text /Reference Books:

1. Aaron M. Tanenbaum, Yedidyah Langsam and Moshe J. Augenstein, "Data Structures Using C and C++",
2. Horowitz and Sahani, "Fundamentals of Data Structures", Galgotia Publications Pvt Ltd Delhi India.
3. Lipschutz, "Data Structures" Schaum's Outline Series, Tata McGraw-hill Education (India) Pvt. Ltd.
4. Thareja, "Data Structure Using C" Oxford Higher Education.
5. AK Sharma, "Data Structure Using C", Pearson Education India.
6. Rajesh K. Shukla, "Data Structure Using C and C++" Wiley Dreamtech Publication.

Course Outcomes:

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

CO1: Understand the role and applications of data structure in real life.

CO2: Demonstrate proficiency in core programming language concepts.

CO 3: Understand & apply basic data structure operations like searching, insertion, and deletion, traversing mechanism etc. on various problem domains.

CO 4: Design and implement advanced data structures, including non-linear ones, and assess their efficiency in solving complex problems.

CO5: Analyze the efficiency of algorithm.

Course Title	Database Management System (DBMS)	**Course Code:CSAB2020**	
Teaching Scheme		**Evaluation Scheme**	
Lectures / Tutorial	**3 / 1 Hrs Week**	**Internal Assessment Test**	**30 Marks**
Total Credits	**4**	**End-Semester Examination**	**70 Marks**

Course Objectives:

1. To make the students learn and practice data modeling using the entity-relationship and developing database designs.
2. To make students understand the use of Structured Query Language (SQL) and learn SQL syntax.
3. To illustrate students how to apply normalization techniques to normalize the database
4. To make students understand the needs of database processing and learn techniques for controlling the consequences of concurrent data access.

Detailed Syllabus:

Unit-I

Introduction to DBMS: Introduction, Characteristics of databases, File system v/s Database system, Users of Database system Data Independence, DBMS system architecture, Database Administrator

Unit-II

Entity–Relationship Data Model

The Entity-Relationship (ER) Model: Entity types: Weak and strong entity sets, Entity sets, Types of Attributes, Keys, Relationship constraints: Cardinality and Participation, Extended Entity-Relationship (EER) Model: Generalization, Specialization and Aggregation

Unit-III

Relational Model and Relational Algebra

Introduction to the Relational Model, relational schema and concept of keys, Mapping the ER and EER Model to the Relational Model Relational Algebra – unary and set operations, Relational Algebra Queries.

Unit-IV

Structured Query Language (SQL)

Overview of SQL, Data Definition Commands, Data Manipulation commands, Data Control commands, Transaction Control Commands. Set and string operations, aggregate function - group by, having, Views in SQL, joins, Nested and complex queries, Integrity constraints: - key constraints, Domain Constraints, Referential integrity, check constraints Triggers

Unit-V

Relational–Database Design and Transaction Management

Pitfalls in Relational-Database designs, Concept of normalization, Function Dependencies, First Normal Form, 2nd , 3rd , BCNF, multi valued dependencies , 4NF. Transaction concept, Transaction states, ACID properties, Concurrent Executions, Serializability – Conflict and View.

Text /Reference Books:

1. G. K. Gupta ―Database Management Systems‖, McGraw – Hill.
2. Korth, Slberchatz, Sudarshan, ―Database System Concepts‖, 6th Edition, McGraw – Hill
3. Elmasri and Navathe, ―Fundamentals of Database Systems‖, 5th Edition, Pearson education.
4. Peter Rob and Carlos Coronel, ―Database Systems Design, Implementation and Management‖, Thomson Learning, 5th Edition.

Course Outcomes:

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

CO1: Understand the fundamentals of a database systems.

CO 2: Design and draw ER and EER diagrams for real life problems.

CO 3: Understand relational model basics, schema with keys, ER/EER mapping to relations, and apply relational algebra for queries.

CO 4: Design and querying databases using SQL.

CO 5: Analyze and apply concepts of normalization to relational database design.

Course Title	Introduction to R programming	**Course Code:** CSAB2040	
Teaching Scheme		**Evaluation Scheme**	
Lectures / Tutorial	**3 / 1 Hrs Week**	**Internal Assessment Test**	**30 Marks**
Total Credits	**4**	**End-Semester Examination**	**70 Marks**

Course Objectives:

1. To introduce the basics of the R programming language and development environment.
2. To equip students with the ability to write logical and modular code using R constructs.
3. To train students in reading, cleaning, and manipulating data using R packages.
4. To develop skills in data visualization to interpret and present data effectively.
5. To provide a foundation in statistical techniques and their implementation using R.

Detailed Syllabus:

Unit-I

Introduction to R Programming Environment:R programming language and its environment. Students will learn how to install and set up R and RStudio, use the R console and scripts, and explore basic syntax and operations. Data types such as vectors, lists, matrices, data frames, and factors. Students will also gain hands-on experience using variables, expressions, and operators in R to build a solid programming foundation.

Unit-II

Control Structures and Functions: The control flow in R using conditional statements (`if`, `else`, `switch`) and loops (`for`, `while`, `repeat`, `break`, `next`). Functions in R, including the creation and use of user-defined and built-in functions.Importance of modular programming, scope of variables, and techniques to make their code reusable and efficient.

Unit-III

Data Handling and Manipulation: read, write, and manipulate various data formats like CSV, Excel, and text files using packages such as `readr`, `readxl`, and `dplyr`. This unit covers data cleaning, filtering, sorting, grouping, and reshaping using `tidyr`. string operations and regular expressions to handle textual data. The emphasis is on real-world data preprocessing skills for analysis.

Unit-IV

Data Visualization: Graphical representation of data using R. Students will learn to use both base R plotting functions and advanced visualization tools provided by the `ggplot2` package. Key plots such as bar charts, histograms, box plots, scatter plots, and line graphs will be covered.explores plot customization, including labels, legends, themes, and colors to enhance the visual appeal and readability of graphs.

Unit-V

Introduction to Statistical Analysis

basic statistical methods using R.descriptive statistics such as mean, median, mode, variance, and standard deviation, along with probability distributions like normal, binomial, and Poisson. inferential statistics including t-tests, chi-square tests, correlation, and simple regression. Tools from `stats` and `psych` packages will be used to perform analysis.

Assignments

Assignment 1: Getting Started with R Environment

Aim:To understand the R programming environment, data types, and basic operations.

Practical Tasks:

1. Install R and RStudio on your system.
2. Perform basic arithmetic operations: addition, subtraction, multiplication, division.
3. Create variables of different types: numeric, character, logical.
4. Use built-in functions like `sqrt()`, `abs()`, `round()`, and `sum()`.

Expected Output:
Basic outputs in the R console using arithmetic and built-in functions.

Assignment 2: Vectors, Lists, and Data Frames

Aim:To learn how to create and manipulate vectors, lists, and data frames.

Practical Tasks:

1. Create a numeric vector of student marks and perform basic operations (mean, max, min).
2. Create a list containing student details (name, roll number, grades).
3. Create a data frame with at least 5 rows and 3 columns (e.g., Name, Subject, Marks).
4. Use `str()`, `summary()`, and `class()` functions to inspect the data structures.

Expected Output:
Structured outputs showing data frame rows, list contents, and vector operations.

Assignment 3: Control Structures and Loops

Aim:To implement decision-making and loops in R.

Practical Tasks:

1. Write a program to check if a number is even or odd using if-else.
2. Use a for loop to print numbers from 1 to 10.
3. Create a loop to calculate the factorial of a given number.
4. Use a while loop to find the sum of numbers until the value exceeds 100

Expected Output:
Text output in the console showing loop execution and condition checks.

Assignment 4: Reading, Writing & Manipulating Data

Aim: To read external data into R, manipulate it, and write back to a file.

Practical Tasks:

1. Read a CSV file using `read.csv()` (can use sample dataset).
2. Display first few rows using `head()` and structure using `str()`.
3. Use `subset()`, `filter()`, or `dplyr::select()` to extract specific columns/rows.
4. Write the modified data to a new CSV file using `write.csv()`.

Expected Output:
Modified data in R console and a saved CSV file on disk.

Assignment 5: Data Visualization using Base R and ggplot2

Aim:To create basic data visualizations using R plotting tools.

Practical Tasks:

1. Create a bar chart of student marks using `barplot()`.
2. Plot a histogram for randomly generated data using `hist()`.
3. Use `plot()` to create a scatter plot of height vs. weight.
4. Install and use `ggplot2` to recreate one of the charts with improved design.

Expected Output:
Graphs displayed in the RStudio plotting pane with axes, labels, and titles.

Text /Reference Books:

1. The Art of R Programming – Norman Matloff, No Starch Press, 2011
2. R for Data Science – Hadley Wickham & Garrett Grolemund, O'Reilly, 2017
3. Hands-On Programming with R – Garrett Grolemund, O'Reilly, 2014
4. R Cookbook – Paul Teetor, O'Reilly, 2011
5. Practical Data Science with R – Nina Zumel & John Mount, Manning, 2014

Course Outcomes:

At the end of this course students will able to
CO1:Understand and apply fundamental R programming concepts and data structures.
CO2:Use control structures and functions to solve computational problems.
CO3:Perform efficient data manipulation and preprocessing using tidyverse tools.
CO4:Create and customize a variety of data visualizations to communicate insights.
CO5:Conduct basic statistical analysis and interpret results using R.

Course Title	Environmental Studies (EVS)	**Course Code:**EVSG2000	
Teaching Scheme		**Evaluation Scheme**	
Lectures	**3 Hrs Week**	**Internal Assessment Test**	**30 Marks**
Total Credits	**3**	**End-Semester Examination**	**70 Marks**

Course Objectives:

1. Recognize the interconnectedness of multiple factors in environmental challenges
2. Engage constructively with diverse forms of knowledge and experience
3. Identify the multiple scales, actors, and stakes of an issue
4. Recognize and apply methodological approaches of the social sciences, natural sciences, and humanities
5. Identify assumptions inherent in arguments and perspectives

Detailed Syllabus:

Unit-I

Multidisciplinary nature of environmental studies

Multidisciplinary nature of environmental studies Definition, scope and importance Need for public awareness.

Unit-II

Natural Resources

Renewable and non-renewable resources: Natural resources and associated problems, Forest resources: Use and over-exploitation, deforestation, case studies. Timber extraction, mining, dams and their effects on forest and tribal people. Water resources: Use and over-utilization of surface and ground water, floods, drought, conflicts over water, dams-benefits and problems. Mineral resources: Use and exploitation, environmental effects of extracting and using mineral resources, case studies. Food resources: World food problems, changes caused by agriculture and overgrazing, effects of modern agriculture, fertilizer- pesticide problems, water logging, salinity, case studies. Energy resources: Growing energy needs, renewable and non-renewable energy sources, use of alternate energy sources. Case studies. Land resources: Land as a resource, land degradation, man induced landslides, soil erosion and desertification, Role of an individual in conservation of natural resources, Equitable use of resources for sustainable lifestyles.

Unit-III

Ecosystems

Concept of an ecosystem, Structure and function of an ecosystem, Producers, consumers and decomposers, Energy flows in the ecosystem, Ecological succession, Food chains, food webs and ecological pyramids. Introduction, types, characteristic features, structure and function of the following ecosystem: - a. Forest ecosystem b. Grassland ecosystem c. Desert ecosystem d. Aquatic ecosystems (ponds, streams, lakes, rivers, oceans, estuaries).

Unit-IV

Biodiversity and its conservation

Introduction – Definition: genetic, species and ecosystem diversity. Biogeographical classification of India • Value of biodiversity: consumptive use, productive use, social, ethical, aesthetic and option values • Biodiversity at global, National and local levels. • India as a mega-diversity nation V • Hot-sports of biodiversity. • Threats to biodiversity: habitat loss, poaching of wildlife, man-wildlife conflicts. • Endangered and endemic species of India • Conservation of biodiversity: In-situ and Ex-situ conservation of biodiversity.

Unit-V

Environmental Pollution Definition

Cause, effects and control measures of: - a. Air pollution b. Water pollution c. Soil pollution d. Marine pollution e. Noise pollution f. Thermal pollution g. nuclear hazards • Solid waste Management: Causes, effects and control measures of urban and industrial wastes. • Role of an individual in prevention of pollution. • Pollution case studies. • Disaster management: floods, earthquake, cyclone and landslides.

Text /Reference Books:

1. Agarwal, K.C. 2001 Environmental Biology, Nidi Publ. Ltd. Bikaner.
2. Bharucha Erach, The Biodiversity of India, Mapin Publishing Pvt. Ltd., Ahmedabad – 380 013, India, Email: mapin@icenet.net (R)
3. Brunner R.C., 1989, Hazardous Waste Incineration, McGraw Hill Inc. 480p
4. Clark R.S., Marine Pollution, Clanderson Press Oxford (TB)
5. Cunningham, W.P. Cooper, T.H. Gorhani, E & Hepworth, M.T. 2001, Environmental Encyclopaedia, Jaico Publ. House, Mumabai, 1196p 6. De A.K., Environmental Chemistry, Wiley Eastern Ltd.
7. Down to Earth, Centre for Science and Environment (R)
8. Gleick, H.P. 1993. Water in crisis, Pacific Institute for Studies in Dev., Environment & Security. Stockholm Env. Institute Oxford Univ. Press. 473p
9. Hawkins R.E., Encyclopaedia of Indian Natural History, Bombay Natural History Society, Bombay (R)
10. Heywood, V.H & Waston, R.T. 1995. Global Biodiversity Assessment. Cambridge Univ. Press 1140p.
11. Jadhav, H & Bhosale, V.M. 1995. Environmental Protection and Laws. Himalaya Pub. House, Delhi 284 p.

Course Outcomes:

CO1: Understand key concepts in the life and physical sciences, and will apply them to environmental issues.

CO2: Understand and apply the scientific process,

CO3: Understand, evaluate and synthesize information from the scientific literature.

CO4: Analyze data using appropriate statistical methods, and will be able to evaluate the use of statistics by others in a variety of contexts.

CO5: Apply knowledge of the sciences within an interdisciplinary context in solving environmental issues

Course Title	Software Engineering	**Course Code:CSAB2030**	
Teaching Scheme		**Evaluation Scheme**	
Lectures	**3 Hrs Week**	**Internal Assessment Test**	**30 Marks**
Total Credits	**3**	**End-Semester Examination**	**70 Marks**

Course Objectives:

1. To make the students familiarize with system analysis and design is the backbone of Application software development.
2. To illustrate students the steps in analysis and design of the system.
3. It includes the knowledge of preparing projects systematically.
4. It is important to know about various aspects of the system analysis and design so that the students will be able to understand the responsibilities while designing and implementing the project.

Detailed Syllabus:

Unit-I

Introduction to Software Engineering, System Concepts: Types of systems: (open, closed, static and dynamic systems). Introduction, Programmes v/s Software Products Emergence of Software Engineering- Early Computer Programming, High-level Language Programming, Control flow-based Design, Data Structure Oriented Design, Object Oriented Design

Unit-II

Software Life Cycle Models, Requirement of Life Cycle Model, Classic Waterfall Model, Prototyping Model, Evolutionary Model, Spiral Model, introduction to agile methodology. Comparison of different Life Cycle Models

Unit-III

Software Planning, Responsibilities of Software Project Manager - Metrics for Project Size Estimation-LOC (Lines of Code), Function Point Metric - Project estimation Techniques- Using COCOMO Model.

Unit-IV

Requirement Analysis and Specification, Requirement gathering and Analysis, Software Requirement Specifications (SRS), Characteristics of good SRS

Unit-V

Software Testing, Concept of Testing, Testing type cycle (V-Model), Verification v/s Validations, Unit Testing, Black Box Testing, White Box Testing, Integration testing, System testing, Configuration management, Overview of test cases.

Text /Reference Books:

1. Software Engineering by Rajib Mall, PHI Publishers, New Delhi
2. An Integrated Approach to Software Engineering by Pankaj Jalote, Narosa Publishing House Pvt Ltd, Darya Ganj, New Delhi 110002
3. Software Engineering, Sangeeta Sabharwal, New Age International, Delhi
4. Software Engineering by KK Aggarwal and Yogesh Singh
5. Software Engineering – A Practitioner's Approach by RS Pressman, Tata McGraw Hill Publishers, New Delhi

Course Outcomes:

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

CO1: Understand the problem and corresponding requirement for development of software.

CO2: Demonstrate the various phases of the system development life cycle.

CO3: Identify the expected benefits and scope of the projects.

CO4: Prepare and develop data flow diagrams and decision tables.

CO5: Apply different testing techniques on software.

Course Title	Element of Statistics	**Course Code:**MTHG2010	
Teaching Scheme		**Evaluation Scheme**	
Lectures / Tutorial	**3 / 1 Hrs Week**	**Internal Assessment Test**	**30 Marks**
Total Credits	**4**	**End-Semester Examination**	**70 Marks**

Course Objectives:

1. To demonstrate knowledge of probability and the standard statistical distributions.
2. To demonstrate knowledge of fixed-sample and large-sample statistical properties of point and interval estimators.
3. To demonstrate knowledge of the properties of parametric, semi-parametric and nonparametric testing procedures.
4. To demonstrate the ability to perform complex data management and analysis.

Detailed Syllabus:

Unit-I

Introduction to Statistics:
Introduction, Definitions of Statistics, Importance of Statistics, Scope of Statistics

Unit-II

Data:
Introduction: Nature of Subject, Language of Statistics: Population, Variables, Size of Population, Discrete and Continuous Variables, Classification of data: Classification by attributes, Classification of variables, Graphical representation of data: Histogram, Frequency polygon, Ogive curves, Diagrammatic representation of data: Simple bar diagram, Subdivided bar diagram, Pie diagram.

Unit-III

Measures of Central Tendency:
Introduction, Arithmetic mean: Properties of arithmetic mean, Merits and Demerits of mean, Median: Merits and Demerits of median, Mode: Merits and Demerits of mode.

Unit-IV

Measures of Dispersion:
Introduction, Range, Mean Deviation, Variance, Standard Deviation, Absolute and Relative Measure of Dispersion, Coefficient of Variation

Unit-V

Correlation and Linear Regression:
Correlation: Introduction, Positive & Negative Correlation Covariance, Coefficient of Correlation: Properties of Correlation Coefficient, Interpretation of the value of Correlation Coefficient, Computing Correlation Coefficient for Ungrouped Data.
Linear Regression: Introduction, Line of Regression, Equation of Line of Regression by the Method of Least Squares, Interpretation of Coefficient of Regression, Properties of Coefficient of Regression.

Text /Reference Books:

1. “Probability and Statistics” by M H DeGroot.
2. "Introduction to the Practice of Statistics” by David S Moore and George P McCabe
3. Mathematical Statistics by Ray, Sharma and Choudhary
4. Business Statistics By V. K. Kapoor, pub: S. Chand
5. Schaum's outline of theory and problems of statistics Murray R. Spiegel, Larry J. Stephens

Course Outcomes:

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

CO1: Understand the key terminology, concepts, tools and techniques used in statistical analysis.

CO2: Analyze the effectiveness of different graphical representations in conveying information about the dataset.

CO3: Apply different measures of central tendency to analyze and interpret real-life data.

CO4: Discuss critically the uses and limitations of statistical analysis.

CO5: Analyze the properties of correlation coefficient and regression coefficient.

Course Title	Data Structure using C Lab	Course Code:CSAB2011	
Teaching Scheme		**Evaluation Scheme**	
Lectures	**4 Hrs/Week**	**Internal Assessment Test**	**30 Marks**
Total Credits	**2**	**End-Semester Examination**	**70 Marks**

Course Objectives:

1. To understand the basic concepts of data structures such as arrays, linked lists, stacks, queues, trees, and graphs.
2. To develop skills in writing and debugging C programs that utilize various data structures to solve real-world problems efficiently.
3. To demonstrate the ability to apply dynamic memory allocation
4. To enable students to analyze and compare the performance of different data structures

Detailed Syllabus:

1. Implementation of Insertion Sorting Algorithm.
2. Implementation of Merge Sorting Algorithm.
3. Implementation of Selection sorting algorithm.
4. Implementation of Sequential Searching Algorithm.
5. Implementation of Binary search.
6. Implementation of Stack using Array.
7. Implementation of Queue using Array.
8. Implementation of Circular Queue using Array.
9. Implementation of Linked List: create, insert, delete, update.
10. Implementation of Queue using Linked List.
11. Implementation of Circular Queue using Linked List.
12. Implementation of graph representation using linked list and matrix.

Course Outcomes:

After completion of this subject students will be able to

CO1: Demonstrate proficiency in implementing algorithms such as insertion sort, merge sort, selection sort.

CO2: Analyze problem requirements and design algorithmic solutions to efficiently manage data and perform operations like sorting, searching, and traversal.

CO3: Create and implement algorithms using appropriate data structures such as arrays, linked lists, and matrices.

CO4: Apply critical thinking to evaluate algorithm efficiency, optimize code, and handle edge cases effectively.

CO5: Develop algorithms for complex data structures like graphs, considering factors such as connectivity, traversal, and representation.

Course Title	Database Management System Lab	Course Code: CSAB2021	
Teaching Scheme		**Evaluation Scheme**	
Lectures	**4 Hrs/Week**	**Internal Assessment Test**	**15 Marks**
Total Credits	**2**	**End-Semester Examination**	**35 Marks**

Course Objectives:

1. To impart hands-on experience with database design and implementation using SQL and relational database management systems like MySQL or Oracle.
2. To enable students to create and manipulate database schemas through DDL, DML, DCL, and TCL commands for managing data effectively.
3. To develop skills in writing complex queries, including nested queries, joins, and set operations to retrieve meaningful insights from data.
4. To introduce normalization techniques and integrity constraints to ensure data consistency, efficiency, and reduce redundancy.
5. To familiarize students with PL/SQL programming constructs such as cursors, triggers, stored procedures, and functions to build robust database applications.

Detailed Syllabus:

1. Creating a Table with constraints.
2. Writing SQL statements Using ORACLE 1. Writing basic SQL SELECT statements.
    2. Restricting and sorting data.
    3. Displaying data from multiple tables.
    4. Aggregating data using group functions.
    5. Creating and managing tables.
3. Normalization in ORACLE.
4. Creating a cursor in oracle.
5. Creating procedure and functions in oracle.
6. Creating packages and triggers in oracle.
7. Design a Database and create required tables. For e.g. Bank, College Database 8. Apply the constraints like Primary Key, Foreign key, NOT NULL to the tables.
9. Write a sql statement for implementing ALTER, UPDATE and DELETE
10. Write the queries to implement the joins
11. Write the query for implementing the following functions: MAX (), MIN (), AVG (), COUNT ()
12. Write the query to implement the concept of Integrity constraints
13. Write the query to create the views
14. Perform the queries for triggers
15. Perform the following operation for demonstrating the insertion, updation and deletion using the referential integrity constraints
16. Write the query for creating the users and their role.

	BCA- AI ML 2 Year - Semester III										
Type	**Course Code**	**Course Title**	**Hours/ Week**			**Theory Marks**		**Practical Marks**		**Total Marks**	**Credit**
			L	**T**	**P**	**IA**	**ESE**	**IA**	**ESE**		
DC	CSAB3010	Computer Organization & Architecture	3	1	-	30	70	-	-	100	4
DC	CSAB3020	Object Oriented Programming with C++	3	1	-	30	70	-	-	100	4
DC	CSAB3030	Operating System	3	1	-	30	70	-	-	100	4
DC	CSAB3040	Software Testing and Quality Assurance	3	1	-	30	70	-	-	100	4
DC	CSAB3050	Machine Learning Fundamentals	3	1	-	30	70	-	-	100	4
OE	****	Elective 2	3		-	30	70	-	-	100	3
DC	CSAB3021	Object Oriented Programming with C++ Lab	-	-	4	-	-	15	35	50	2
DC	CSAB3031	Operating System Lab	-	-	4	-	-	15	35	50	2
SEC		Quantitative Aptitude & Logical Reasoning-1									
		TOTAL	**18**	**5**	**8**	**180**	**420**	**30**	**70**	**700**	**27**

Elective-2
MGTG3100/CSEB3040
Financial Accounting & Management / Management Information System

Course Title	Computer Organization & Architecture	Course Code:CSAB3010	
Teaching Scheme		**Evaluation Scheme**	
Lectures / Tutorial	**3 / 1 Hrs Week**	**Internal Assessment Test**	**30 Marks**
Total Credits	**4**	**End-Semester Examination**	**70 Marks**

Course Objectives:

1. To make students conceptualize the basics of organizational and architectural issues of a digital computer.
2. To make students learn the various methods for data representation.
3. To analyze the processor and memory design of a digital computer.
4. To understand various data transfer techniques in digital computers to learners.
5. To make the students analyze processor performance improvement using instruction level parallelism.

Detailed Syllabus:

Unit-I

Digital Logic Circuits: Basic Logic Functions, Synthesis of Logic Functions Using AND, OR, and NOT Gates, Minimization of Logic Expression, Synthesis with NAND and NOR Gates, Flip-Flops, Registers and Shift Registers, Counters, Decoders, Multiplexers, Programmable Logic Devices (PLDs), Basic Structure of Computers: Functional units, Basic operational concepts, Bus structures, Software, Performance, multiprocessors and multi computers

Unit-II

Data Representation: Data types, Complements, Other binary codes, Error Detection codes, Register and Micro operations: Register Transfer language, Register Transfer Bus and memory transfers, Arithmetic Micro operations, logic micro-operations, Shift micro-operations, Arithmetic logic shift unit

Unit-III

Processing Unit: Instruction Codes, Computer Registers, Computer Instructions, Instruction Cycle, Memory Reference Instructions, Hardwired Control, Micro Programmed Control, register organization, Stack organization, Instruction formats Addressing modes, Data Transfer and manipulations, RISC, CISC, Computer Arithmetic: Addition, subtraction, multiplication and division operations, Floating point Arithmetic operations

Unit-IV

Input-Output Organization: Peripheral Devices, Input Output Interface, Asynchronous data transfer, Modes of Transfer, Priority Interrupt, Direct memory Access, Input-Output Processor, Serial communication, Parallel and Vector Processing: Parallel processing, Pipelining, Arithmetic pipeline, Instruction pipeline, RISC Pipeline, Vector Processing, Array Processors

Unit-V

Memory Organization: Memory Hierarchy, Main memory, Auxiliary memory, Associative memory Virtual memory, Cache memory, Memory management hardware

Text /Reference Books:

1. John L. Hennessy and David A. Patterson, “Computer Architecture a quantitative approach”, 4th Edition Elsevier, ISBN:10:0123704901
2. William Stallings, “Computer Organization and Architecture”, 6th Edition, Pearson/PHI, ISBN:10:0- 13-609704-9
3. Donald e Givone, “Digital Principles and Design”, TMH.
4. A.Anandkumar, “Fundamentals of digital circuits”, 4th edition, PHI.

Course Outcomes:

CO1: Create logic circuit designs.

CO2: Understand the data types, codes and register, ALU and shift microoperations.

CO3: Understand the intricacies of processing unit.

CO4: Explain concept of data transfer between io devices and memory unit and the types data processing.

CO5: Analize Memory Hierarchy to describe the memory hierarchy, including registers, cache memory, main memory (RAM), and secondary storage (e.g., hard disk drives, solid-state drives).

Course Title	Object Oriented Programming with C++	**Course Code:** CSAB3020	
Teaching Scheme		**Evaluation Scheme**	
Lectures / Tutorial	**3 /1 Hrs Week**	**Internal Assessment Test**	**30 Marks**
Total Credits	**4**	**End-Semester Examination**	**70 Marks**

Course Objectives:

1. To make students understand Object Oriented Programming concepts using the C++ language.
2. To illustrate students the concepts of classes and objects.
3. To explain the concept of c++ principles of data abstraction, inheritance to the learner.
4. To introduce the principles of virtual functions and polymorphism.
5. To make the students familiarize with the concept of exception handling.

Detailed Syllabus:

Unit-I

Object-Oriented Thinking: Different paradigms for problem solving, need for OOP paradigm, differences between OOP and Procedure oriented programming, Overview of OOP concepts-Abstraction, Encapsulation, Inheritance and Polymorphism.

Unit-II

C++ Basics: Structure of a C++ program, Data types, Declaration of variables, Expressions, Operators, Operator Precedence, Evaluation of expressions, Type conversions, Pointers, Arrays, Pointers and Arrays, Strings, Structures, References. Flow control statement- if, switch, while, for, do, break, continue, goto statements. Functions - Scope of variables, Parameter passing, Default arguments, inline functions, Recursive functions, Pointers to functions. Dynamic memory allocation and de-allocation operators-new and delete, Preprocessor directives.

Unit-III

C++ Classes and Data Abstraction: Class definition, Class structure, Class objects, Class scope, this pointer, Friends to a class, Static class members, Constant member functions, Constructors and Destructors, Dynamic creation and destruction of objects, Data abstraction, ADT and information hiding.

Unit-IV

Inheritance: Defining a class hierarchy, Different forms of inheritance, Defining the Base and Derived classes, Access to the base class members, Base and Derived class construction, Destructors, Virtual base class.

Unit-V

Virtual Functions and Polymorphism: Static and Dynamic binding, virtual functions, Dynamic binding through virtual functions, Virtual function call mechanism, Pure virtual functions, Abstract classes, Implications of polymorphic use of classes, Virtual destructors.

Text /Reference Books:

1. Object Oriented Programming in C++ -Robert Lafore, edition, Galgotia publications
2. The Complete Reference C++, Herbert Schildt, 4th Edition, TMH
3. Y. Kanetkar, 'Let's C++', BPB publications
4. E Balagurusamy, 'Object oriented programming with C++', 4th edition, TMH
5. Sourav Sahay , 'Object-Oriented Programming with C++' , Oxford University Press, 2006.

Course Outcomes:

At the end of this course students will able to

CO1: Evaluate the difference between object-oriented programming & procedure-oriented programming

CO2: Design C++ program to solve problems using basic C++ constructs.

CO3: Create programs using Classes and Objects in C++.

CO4**:** Understand inheritance in object-oriented programming

CO5: Understand the concept of virtual functions, polymorphism in object-oriented programming, including static polymorphism (compile-time polymorphism) and dynamic polymorphism (runtime polymorphism)

Course Title	Operating System	Course Code: CSAB3030	
Teaching Scheme		**Evaluation Scheme**	
Lectures / Tutorial	**3/1 Hrs Week**	**Internal Assessment Test**	**30 Marks**
Total Credits	**4**	**End-Semester Examination**	**70 Marks**

Course Objectives:

1. To make students learn the evolution of OS.
2. To make the learners introduce basic concepts of Operating System, its functions and services.
3. To familiarize the students with various views and management policies adopted by O.S. as pertaining with processes, Deadlock, memory, File and I/O operations.
4. To make students understand functionality of various OS like Unix, Linux and Windows XP as Pertaining to resource management.
5. To provide the knowledge of basic concepts towards process synchronization and related Issues to the students.

Detailed Syllabus:

Unit-I

Introduction:
Operating system and functions, Classification of Operating systems- Batch, Interactive, Time-sharing, Real-Time System, Multiprocessor Systems, Multiuser Systems, Multiprocessing Systems, Multithreaded Systems, Operating System Structure- Layered structure, System Components, Operating System services, Monolithic and Microkernel Systems.

Unit-II

Concurrent Processes:
Process Concept, Principle of Concurrency, Producer / Consumer Problem, Mutual Exclusion, Critical Section Problem, Dekker's solution, Peterson's solution, Semaphores, Test and Set operation; Classical Problem in Concurrency- Dining Philosopher Problem, Inter Process Communication models.

Unit-III

CPU Scheduling:
Scheduling Concepts, Performance Criteria, Process States, Process Transition Diagram, Schedulers, Process Control Block (PCB), Process address space, Process identification information, Threads and their management, Scheduling Algorithms, Multiprocessor Scheduling.
Deadlock: System model, Deadlock characterization, Prevention, Avoidance and detection, Recovery from deadlock

Unit-IV

Basic bare machine, Resident monitor, Multiprogramming with fixed partitions, Multiprogramming with variable partitions, Protection schemes, Paging, Segmentation, Paged segmentation, Virtual memory concepts, Demand paging, Performance of demand paging, Page replacement algorithms, Thrashing, Cache memory organization, Locality of reference.

Unit-V

I/O Management and Disk Scheduling: I/O devices and I/O subsystems, I/O buffering, Disk storage and disk scheduling, RAID. File System: File concept, File organization and access mechanism, File directories, and File sharing, file system implementation issues, File system protection and security

Text /Reference Books:

1. Silberschatz, Galvin and Gagne, "Operating Systems Concepts", Wiley
2. Sibsankar Halder and Alex A Aravind, "Operating Systems", Pearson Education
3. Harvey M Dietel, " An Introduction to Operating System", Pearson Education
4. D M Dhamdhere, "Operating Systems : A Concept based Approach", 2nd Edition, TMH
5. William Stallings, "Operating Systems: Internals and Design Principles", 6th Edition, Pearson Education
6. A.S Tanenbaum "Modern Operating Systems" Pearson Education

Course Outcomes:

At the end of this course students will able to

CO1: Understand the role of operating system as System software

CO2: Analyze the various algorithms for management of memory and apply various concepts related with Deadlock to solve problems related with resources allocation

CO3: Discuss various algorithms of CPU scheduling and Deadlock system model.

CO4**:** Understand the concepts of memory management, paging, virtual memory.

CO5: Understand the I/O Management and Disk Scheduling.

Course Title	Software Testing and Quality Assurance	**Course Code:** CSAB3040	
Teaching Scheme		**Evaluation Scheme**	
Lectures / Tutorial	**3/1 Hrs Week**	**Internal Assessment Test**	**30 Marks**
Total Credits	**4**	**End-Semester Examination**	**70 Marks**

Course Objectives:

1. To provide learners knowledge in Software Testing techniques.
2. To make students understand how testing methods can be used as effective tools in providing quality assurance concerning software.
3. To make students understand the concept of software metrics and management process.
4. To provide skills to design test case plans for testing software to the students.

Detailed Syllabus:

Unit-I

Software Testing and Introduction to quality:
Introduction, Nature of errors, an example for Testing, Definition of Quality, QA, QC, QM and SQA, Software Development Life Cycle, Software Quality Factors Verification and Validation: Definition of V&V, Different types of V & V Mechanisms, Concepts of Software Reviews, Inspection and Walkthrough.

Unit-II

Software Testing Techniques:
Testing Fundamentals, Test Case Design, White Box Testing and its types, Black Box Testing and its types.

Unit-III

Software Testing Strategies:
Strategic Approach to Software Testing, Unit Testing, Integration Testing, Validation Testing, System Testing

Unit-IV

Metrics:
Concept and Developing Metrics, Different types of Metrics, Complexity metrics
Defect Management: Definition of Defects, Defect Management Process, Defect Reporting, Metrics Related to Defects, Using Defects for Process Improvement.

Unit-V

Software Quality Assurance:
Quality Concepts, Quality Movement, Background Issues, SQA activities, Software Reviews, Formal Technical Reviews, Formal approaches to SQA, Statistical Quality Assurance, Software Reliability, The ISO 9000 Quality Standards, SQA Plan, Six sigma, Informal Reviews 15L

Text /Reference Books:

1. Software Engineering for Students, A Programming Approach, Douglas Bell, 4 th Edition,, Pearson Education, 2005
2. Software Engineering – A Practitioner's Approach, Roger S. Pressman, 5 th Edition, Tata McGraw Hill, 2001
3. Quality Management, Donna C. S. Summers, 5 th Edition, Prentice-Hall, 2010. 4. Total Quality Management, Dale H. Besterfield, 3 rd Edition, Prentice Hall, 2003

Course Outcomes:

CO1: Discuss the concept of Software testing and software quality factors.
CO2: Understand various software testing methods and strategies.
CO3: Create & Design test cases for different software techniques.
CO4: Understand a variety of software metrics and identify defect management
CO5: Understand the software quality concepts.

Course Title	Machine Learning Fundamentals	**Course Code:** CSAB3050	
Teaching Scheme		**Evaluation Scheme**	
Lectures / Tutorial	**3 / 1 Hrs Week**	**Internal Assessment Test**	**30 Marks**
Total Credits	**4**	**End-Semester Examination**	**70 Marks**

Course Objectives:

1. To introduce the fundamental concepts and techniques of machine learning, including supervised, unsupervised, and reinforcement learning.
2. To provide a strong mathematical foundation for understanding algorithms such as linear regression, decision trees, support vector machines, and neural networks.
3. To enable students to design, implement, and evaluate machine learning models using tools like Python, scikit-learn, TensorFlow, or R.
4. To develop the ability to preprocess and analyze data effectively, including feature selection, dimensionality reduction, and data transformation.
5. To apply machine learning algorithms to solve real-world problems in domains such as healthcare, finance, marketing, and cybersecurity.

Detailed Syllabus:

Unit-I

Introduction to Machine Learning

Definition of Machine Learning: What is Machine Learning? Overview of AI, ML, and Deep Learning.

Types of Machine Learning: Supervised Learning, Unsupervised Learning, Reinforcement Learning.

The Machine Learning Process: Data Collection, Data Preprocessing, Model Building, Model Evaluation, Model Deployment.Applications of Machine Learning: Real-world examples of ML applications in healthcare, finance, retail, and more. Tools and Libraries: Python (NumPy, Pandas), Scikit-learn, Matplotlib, Seaborn, TensorFlow/Keras, Jupyter Notebooks.

Unit-II

Data Preprocessing and Exploration

Data Cleaning: Handling missing values (imputation, removal), removing duplicates, outlier detection and treatment.Feature Engineering: Feature selection, feature extraction, one-hot encoding, label encoding, handling categorical data.Scaling Data: Standardization, normalization, Min-Max scaling.Data Splitting: Train-test split, validation sets, K-Fold Cross-Validation. Exploratory Data Analysis (EDA): Visualizing datasets using Matplotlib and Seaborn, understanding data distributions, identifying relationships between variables.

Unit-III

Supervised Learning Algorithms

Linear Regression: Simple Linear Regression, Multiple Linear Regression, performance evaluation (MSE, RMSE, R^2 score). Logistic Regression: Binary and Multi-Class Classification, performance metrics (accuracy, confusion matrix, precision, recall, F1-score). Decision Trees: Understanding decision trees,

splitting criteria (Gini Index, Entropy), pruning techniques, overfitting and underfitting.K-Nearest Neighbors (k-NN): Nearest neighbor classifier, distance metrics, parameter tuning, model evaluation.

Unit-IV

Unsupervised Learning Algorithms

Clustering:K-Means Clustering: Clustering similar data points, Elbow Method, K-Means++ initialization. Hierarchical Clustering: Dendrograms, Agglomerative clustering. Association Rule Mining: Apriori Algorithm, applications in Market Basket Analysis, support, confidence, and lift metrics.

Unit-V

Introduction to Deep Learning and Neural Networks Neural Networks: Basic building blocks, neurons, activation functions (Sigmoid, ReLU, Tanh). Deep Learning Frameworks: Introduction to TensorFlow/Keras. Training Neural Networks: Forward propagation, loss functions (MSE, Cross-Entropy), backpropagation, gradient descent. Simple Neural Networks for Image Classification: Building and training a simple neural network for MNIST dataset.

Text /Reference Books:

1. **"Machine Learning"** *Author:* Tom M. Mitchell *Publisher:* McGraw-Hill Education
   Description: A classic introductory text that covers key ML algorithms with theoretical foundations.

2. **"Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow" (2nd or 3rd Edition)** *Author:* Aurélien Géron *Publisher:* O'Reilly Media *Description:* Practical guide to building ML systems using Python libraries.

3. **"Introduction to Machine Learning with Python"**
   Authors: Andreas C. Müller, Sarah Guido *Publisher:* O'Reilly Media *Description:* Beginner-friendly guide focused on using Scikit-learn for ML.

4. **"Pattern Recognition and Machine Learning"**
   Author: Christopher M. Bishop *Publisher:* Springer *Description:* A mathematically rigorous text, ideal for understanding probabilistic ML models. actical implementations using Python.

Assignments

Assignment 1: Classifying Weather Data using Decision Trees

Tool: Weka
Aim: To perform basic classification using decision trees (J48).
Steps:

- Load `weather.nominal.arff` dataset in Weka.
- Run the J48 algorithm (decision tree).
- Visualize the decision tree and interpret output.

Learning Outcome: Understand the concept of supervised learning and decision trees.

Assignment 2: Predicting Student Grades using Regression

Tool: Orange (drag-drop interface)
Aim: To apply linear regression to predict student performance.
Steps:

- Import a CSV file (e.g., hours studied vs grades).
- Use the "Linear Regression" widget.
- Train and evaluate the model, observe prediction.

Learning Outcome: Understand regression models and prediction evaluation.

Assignment 3:Train an Image Classification Model

Tool: Google Teachable Machine
Aim: To train a basic image classifier using webcam input.
Steps:

- Create two or more image categories (e.g., smile, no smile).
- Train the model and test it live.
- Export and analyze accuracy.

Learning Outcome: Understand training, validation, and overfitting.

Course Outcomes:

1. CO1 - Understand Machine Learning Fundamentals: Gain a deep understanding of the principles, techniques, and algorithms used in machine learning.
2. CO2 - Develop Skills to Implement Machine Learning Models: Learn to implement machine learning algorithms such as linear regression, decision trees, k-NN, SVM, and clustering algorithms using Python and relevant libraries.
3. CO3 -Master Data Preprocessing Techniques: Preprocess and clean data effectively by handling missing values, outliers, and performing data normalization.
4. CO4 -Evaluate and Optimize Machine Learning Models: Apply various model evaluation techniques, including performance metrics and cross-validation.
5. CO5 -Build and Deploy Machine Learning Models: Gain hands-on experience in deploying machine learning models for practical, real-world applications using frameworks.

Elective-II

Course Title	Financial Accounting & Management	**Course Code:MGTG3100**	
Teaching Scheme		**Evaluation Scheme**	
Lectures / Tutorial	**3 Hrs Week**	**Internal Assessment Test**	**30 Marks**
Total Credits	**3**	**End-Semester Examination**	**70 Marks**

Course Objectives:

The objectives of this course are

1. To equip the students with the skill of preparing accounts and financial statements of various types of business units other than corporate undertakings.

Detailed Syllabus:

Unit-I

Financial Accounting

An Introduction: Introduction, Meaning of Accountancy, book-keeping and Accounting, Accounting Process, Objectives for accounting, Differences between book-keeping and accounting Users of accounting information, Limitations of Accounting, Basic terminologies.

Unit-II

Accounting Concepts

Principles, Bases and Policies: Introduction , Accounting Concepts, Principles, Policies and Standards, Types of accounting concepts - Business Separate entity concept - Going concern concept - Money measurement concept - Periodicity concept - Accrual concept, Accounting Principles - Principle of Income recognition - Principle of expense - Principle of matching cost and revenue - Principle of Historical costs - Principle of full disclosure - Double aspect principle - Modifying Principle - Principle of materiality - Principle of consistency - Principle of conservatism or prudence, Accounting Policies - Changes in Accounting Policies - Disclosure in case of changes in Accounting Policies, Accounting Standards - Scope and functions of Accounting Standards Board - International Financial Reporting System.

Unit-III

Double Entry Accounting

Introduction, meaning of double entry accounting, Classification of accounts under Traditional approach, Classification of accounts under Accounting Equation approach, Comparison of traditional approach with Modern approach equal approach, Accounting Trail, Transactions and events, Meaning and roles of debit and credit, Accounting equation.

Unit-IV

Secondary Books

Introduction, Secondary books, Purchases Book/Purchases Day book - Cash discount, Trade discount - Difference between cash discount and trade discount, Sales Book or Sales Day book - Purchase

Returns Book - Sales Returns Book, bills receivable book - Bills payable book - Cash book, posting to Ledger Accounts Posting to Ledger.

Unit-V

Trial Balance

Introduction, Meaning, Objectives of preparing a trial balance, Methods of preparing a trial balance, Preparation of Trial balance, Adjusting Entries, Errors and their rectification, Errors disclosed by Trial Balance, Errors not disclosed by Trial Balance, steps to locate the errors.

Final Accounts

Introduction, Adjustments before preparing final accounts, Depreciation, Bad Debts and accounting treatment of bad debts, Provision for doubtful debts, Reserves for Discount on Debtors, Reserve for Discount on Creditors, Closing Stock, Trading Account, Profit and Loss Account, Balance Sheet.

Text /Reference Books:

1. Principles of Accounting: J.R. Batlibuoy
2. Advance Accounting and Auditing: Shukla and Grewal
3. “Financial Accounting” by T S Reddy and A Murthy
4. “Financial Accounting” by Goyal V K

Course Outcomes:

At the end of this course students will able to

CO1: Define the concept of bookkeeping and accounting.

CO2: Understand the general purposes and functions of accounting.

CO3: Explain the differences between management and financial accounting.

CO4: Describe the main elements of financial accounting information – assets, liabilities, revenue and expenses.

CO5: Create the main financial statements and their purposes.

Course Title	Management Information System	Course Code:CSEB3040	
Teaching Scheme		**Evaluation Scheme**	
Lectures / Tutorial	**3 Hrs Week**	**Internal Assessment Test**	**30 Marks**
Total Credits	**3**	**End-Semester Examination**	**70 Marks**

Course Objectives:

1. Provide students with comprehensive knowledge and technical skills needed to successfully participate in and support the increasingly applied role of information technology in corporate decision making,
2. Enable graduates to conceptualize and manage the specification, design and implementation of applied information systems,
3. Provide the knowledge of contemporary issues related to the field of managing information systems
4. Develop knowledge and skills required to work effectively in a profession,
5. Enhance self-confidence, ability to make proper decisions and effective communication, and Pursue lifelong learning and continuing education.

Detailed Syllabus:

Unit-I

Organizations and Computing: Introduction, Modern Organization-IT enabled- Networked-Dispersed-Knowledge Organization, Information Systems in Organizations- what are information systems?, Brief history of computing- ENIAC: Way to commercial computers- Advent of artificial intelligence- advent of personal computing-Free Software Movement- Advent of Internet, The role of internet- Internet and Web: they are different-the internet changes everything

Unit-II

Managing Information Systems in Organizations: Introduction, Managing in the Internet Era, Managing Information Systems in Organization the IT interaction model, Challenges for the manager-what information to build? -how much to spend on information systems? -what level of capabilities should be created with information systems? -how centralized should the services be? -what security levels are required? - Technology road map for the organization.

Unit-III

Data and Information: Introduction, data and information- measuring data, information as a resource, information in organizational functions, types of information technology, types of information systems-transaction processing systems management information systems.

Unit-IV

Business Process Integration with IT: Introduction, Business Process Integration- Business processes-example of a complex process, Motivation for Enterprise Systems, Enterprise Resource Planning systems- finance and accounting module-human resource management module-manufacturing and operations module- sales and marketing module.

Unit-V

Managing Data Resources: Introduction , The Need for Data Management- History of data use, Challenges of Data Management- data independence- reduced data redundancy- data consistency- data access- data administration- managing concurrency-managing security- recovery from crashes-application development, Database Concepts- fields, records and files- basic architecture, Data Warehouses- data mining uses.

Text /Reference Books:

1. Management Information Systems, A O'Brien.
2. Management Information System, W S Jawadekar.
3. Management Information Systems, Laaudon and Ludon.
4. Management Information Systems , Robert Schultheis and Mary Summer

Course Outcomes:

CO1: Apply modern tools, techniques, and technology in a functional and productive manner in their professional activities,

CO2: Analyze, design, construct, implement and maintain, usable, reliable, and cost-effective Information Systems (IS) that support operational, managerial, and strategic activities of organizations,

CO3: Analyze, design, manipulate, and implement relational databases on which most IS are built upon,

CO4: Plan, coordinate, monitor, and control IS development projects,

CO5: Study and evaluate existing manual and automated business processes, and identify opportunities for reengineering and/or automation,

Course Title	Object Oriented Programming with C++Lab	Course Code:CSAB3021	
Teaching Scheme		**Evaluation Scheme**	
Lectures / Tutorial	**4 Hrs Week**	**Internal Assessment Test**	**15 Marks**
Total Credits	**2**	**End-Semester Examination**	**35 Marks**

Course Objectives:

1. **To provide hands-on experience in writing and executing C++ programs** using basic constructs like variables, operators, control structures, and functions.
2. **To introduce object-oriented programming (OOP) concepts** such as classes, objects, encapsulation, inheritance, and polymorphism through practical implementation.
3. **To enable students to develop modular and reusable code** using functions, templates, and operator overloading in C++.
4. **To train students in handling data using arrays, pointers, structures, and file operations** for problem-solving.
5. **To develop debugging, logical reasoning, and problem-solving skills** by implementing real-time applications using C++.

List of Experiments:

1. WAP to swap two numbers.
2. WAP to display the grade of students using switch cases.
3. WAP to check whether the given string is palindrome or not.
4. WAP to find LCM and HCF of two numbers using a while loop.
5. WAP to print following alphabet pattern

```
A
A B
A B C
A B C D
```

6. WAP to calculate the sum of 10 elements of an array.
7. WAP to implement linear search.
8. WAP to implement bubble sort.
9. WAP to perform addition of two matrices.
10. WAP to implement an array of structures. Declare structure as Student with variables as rollno, name, and branch. Create an array of structure of size 10 and display information of all students.
11. WAP to declare a class Student having data members roll_no and name. Accept and display this data for 3 objects.
12. WAP to declare a class Time having data members as hrs, min and sec. Write a constructor and destructor for class Time and display for 3 objects. (Default constructor, parameterized constructor, copy constructor).
13. Create two classes as AB and XY. Declare one variable of each class. Swap the contents of variables of two classes using the friend function.

14. WAP to implement single inheritance.

    Parent class- Employee, data members- emp_id, name. Member Functions- void get1(), void put1(). Child class-Fitness, data members- height, weight. Member Functions- void get2(), void put2().
15. WAP to implement multilevel inheritance.
16. WAP to implement Multiple inheritance.
17. WAP to implement hierarchical inheritance.
18. WAP to implement hybrid inheritance using virtual base class.

Course Outcomes:

After completion of this subject students will be able to

CO1: Develop programs using control structures like switch cases, loops, and conditional statements to handle various scenarios and requirements.
CO2: Analyze problem requirements and design algorithmic solutions to solve programming tasks effectively.
CO3: Create and implement efficient algorithms to solve complex problems, such as palindrome checking, matrix addition, and array of structures.
CO4: Develop object-oriented programming solutions by designing and implementing classes and inheritance relationships to model real-world scenarios.
CO5: Apply problem-solving skills to design and implement programs that involve multiple layers of abstraction and complexity, such as multiple inheritance and hybrid inheritance.

Course Title	Operating System Lab	**Course Code:CSAB3031**	
Teaching Scheme		**Evaluation Scheme**	
Lectures / Tutorial	**4 Hrs Week**	**Internal Assessment Test**	**15 Marks**
Total Credits	**2**	**End-Semester Examination**	**35 Marks**

Course Objectives:

1. To provide hands-on experience with key operating system concepts
2. To develop the ability to write and simulate programs
3. To introduce students to system-level programming using C or shell scripting
4. To familiarize students with Linux/Unix commands and shell scripting
5. To cultivate problem-solving and debugging skills

List of Experiments:

1. Installation Process of various Operating System.
2. Commands for files & directories: cd, ls, cp, md, rm, mkdir, rmdir.
3. Creating and viewing files using cat. File comparisons. Disk related commands: checking disk free spaces.
4. Given the list of processes, their CPU burst times and arrival times, display/print the Gantt chart for FCFS and SJF. For each of the scheduling policies, compute and print the average waiting time and average turnaround time. (2 sessions).
5. Given the list of processes, their CPU burst times and arrival times, display/print the Gantt chart for Priority and Round robin. For each of the scheduling policies, compute and print the average waiting time and average turnaround time. (2 sessions).
6. Study of Inter Process communication (using shared memory, pipes or message queues).
7. Implement some memory management schemes – I
8. Implement some memory management schemes – II
9. Implement any file allocation technique (Linked, Indexed or Contiguous).
10. To write simple shell programs by using conditional, branching and looping statements

Course Outcomes:

After completion of this subject students will be able to

CO1: Apply knowledge of different installation processes for various operating systems, including Windows, Linux, and macOS.

CO2: Analyze the functionality and usage of various file and directory commands to perform tasks like file creation, deletion, copying, and directory management effectively.

CO3: Create and manipulate files using commands like cat, and perform file comparisons to identify differences.

CO4: Analyze disk-related commands to check disk free space and perform disk management tasks efficiently.

CO5: Develop shell scripts or programs using conditional statements, loops, and branching

	BCA- AI ML 2 Year- Semester IV										
Type		**Course Title**	**Hours/ Week**			**Theory Marks**		**Practical Marks**		**Total Marks**	**Credit**
			L	**T**	**P**	**IA**	**ESE**	**IA**	**ESE**		
DC	CSAB4010	Python Programming	3	1	-	30	70	-	-	100	4
DC	CSAB4020	JAVA Programming	3	1	-	30	70	-	-	100	4
DC	CSAB4030	Design Analysis & Algorithm	3	1	-	30	70	-	-	100	4
DC	CSAB4040	Computer Networks	3	1	-	30	70	-	-	100	4
DC	CSAB4050	Data Visualization	3	1	-	30	70			100	4
DE	**	Elective 1	3	-	-	30	70	-	-	100	3
DC	CSAB4011	Python Programming Lab	-	-	4	-	-	15	35	50	2
DC	CSAB4021	JAVA Programming Lab	-	-	4	-	-	15	35	50	2
SEC		Verbal Ability									
		TOTAL	**18**	**5**	**8**	**180**	**420**	**30**	**70**	**700**	**27**

Elective-1 CSAB4620/ CSAB4930
1. Data Warehousing and Data Mining
2. E-Commerce

Course Title	Python Programming	**Course Code:** CSAB4010	
Teaching Scheme		**Evaluation Scheme**	
Lectures / Tutorial	**3 / 1 Hrs Week**	**Internal Assessment Test**	**30 Marks**
Total Credits	**4**	**End-Semester Examination**	**70 Marks**

Course Objectives:

1. To make the students understand the concepts of programming before actually starting to write programs.
2. To train the students to develop logic for Problem Solving.
3. To make the students familiar with the basic constructs of programming such as data, operations, conditions, loops, functions etc.
4. To develop among students' problem-solving skills using syntactically simple language i.e. Python (version: 3.X or higher).

Detailed Syllabus:

Unit-I

Introduction to Python: The basic elements of python, basic syntax, Writing and executing simple programs, Basic Data Types such as numbers, strings, declaring variables, Performing assignments, arithmetic operations, Simple input-output, Features.

Unit-II

Python Program Flow Control: Conditional blocks using if, else and elif, Simple for loops in python, for loop using ranges list and dictionaries Use of while loops in python Loop manipulation using pass, continue, break and else, Programming using Python conditional and loops block.

Unit-III

Functions, Scoping and Abstraction: Functions and scoping, Specifications, Recursion, Global variables, Modules, Files, System Functions and Parameters.

Unit-IV

Structured Types, Mutability and Higher-Order Functions: Strings, Tuples, Lists and Dictionaries Lists and Mutability, Functions as Objects.

Unit-V

Exception handling: What is an exception, various keywords to handle exceptions such try, catch, except, else, finally, raise.
Regular Expressions: Concept of regular expression, various types of regular expressions, using match function, Plotting using PyLab.

Text /Reference Books:

1. John V Guttag. “Introduction to Computation and Programming Using Python”, Prentice Hall of India.
2. Paul Gries , Jennifer Campbell, Jason Montojo, Practical Programming: An Introduction to Computer Science Using Python 3, Pragmatic Bookshelf, 2/E 2014.
3. James Payne , Beginning Python: Using Python 2.6 and Python 3, Wiley India, 2010 2.
    a. Lukaszewski, MySQL for Python: Database Access Made Easy, Pact Publisher, 2010
4. R. Nageswara Rao, “Core Python Programming”, dreamtech.
5. Wesley J. Chun. “Core Python Programming - Second Edition”, Prentice Hall.
6. Michael T. Goodrich, Roberto Tamassia, Michael H. Goldwasser, “Data Structures and Algorithms in Pyhon”, Wiley.
7. Kenneth A. Lambert, “Fundamentals of Python – First Programs”, CENGAGE Publication.
8. Luke Sneeringer, “Professional Python”, Wrox.

Course Outcomes:

CO1: Understand the basics of python programming before actually starting to write programs.

CO2: Familiar with the basic constructs of programming such as data, operations, conditions, loops, functions etc.

CO3: Create functions in Python programming

CO4: Store data in advanced structures such lists, tuples, dictionaries, etc., and apply exception handling.

CO5: Apply the problem-solving skills using syntactically simple language i.e. Python (version: 3.X or higher).

Course Title	JAVA Programming	**Course Code:** CSAB4020	
Teaching Scheme		**Evaluation Scheme**	
Lectures / Tutorial	**3 / 1 Hrs Week**	**Internal Assessment Test**	**30 Marks**
Total Credits	**4**	**End-Semester Examination**	**70 Marks**

Course Objectives:

1. To make students understand the concept of Object-Oriented Programming.
2. To help students to understand the use of programming languages such as JAVA
3. To guide and teach the students problems understanding, analyzing skills in order to formulate Algorithms.
4. To provide knowledge about JAVA fundamentals: data types, variables, keywords and Control structures to the students.
5. To understand methods, arrays, inheritance, Interface, package and multithreading and Concept of Applet

Detailed Syllabus:

Unit-I

Fundamental Concepts of object-oriented programming:
Overview of object-oriented programming, classes, objects, messages, abstraction, encapsulation, inheritance, polymorphism, exception handling, and object-oriented containers, Differences and similarity between C++ and JAVA.

Unit-II

Fundamental of Java programming:
Features of Java, JDK Environment & tools, structure of Java program, Keywords, data types, variables, operators, expressions, decision making, looping, type casting, Input output using scanner class.

Unit-III

Classes and objects:
Creating classes and objects, Memory allocation for objects, passing parameters to Methods, returning parameters, Method overloading, Constructor, types of constructors and finalize (), Arrays: Creating an array, one dimensional Array, Two-Dimensional array.

Unit-IV

Inheritance, interface and package:
Types of inheritance: Single Inheritance, Multilevel Inheritance, Method overriding, super keyword, final keyword, abstract class, Virtual Functions, Interface, Packages

Unit-V

Multithreading and Applet:
Life cycle of thread, Methods, Multithreading, Priority in multithreading, Applet life cycle, creating applet

Text /Reference Books:

1. The Complete Reference -Java by Herbert Schild, TMH Publication.
2. Programming with Java- A Primer by E. Balagurusamy, 3rd Edition, TMH Publication.
3. The Complete Reference- JAVA.
4. 3rd Edition By Patrick Naughton, TMH Publ.
5. Java 6, Programming Black Book by Kogent Solution Inc., Dreamtech Press Publ.
6. Java 2, Black Book by Steve Holzner, Paraglyph Press, 2nd Ed.

Course Outcomes:

1. CO1: Use the syntax and semantics of java programming language and basic concepts of OOP.
2. CO2: Design reusable programs using the concepts of inheritance, polymorphism, interfaces and packages.
3. CO3: Apply the concepts of Multithreading to perform multitasking activity.
4. CO4: Implement the Exception handling to develop efficient and error free codes.
5. CO5: Design event driven GUI and web related applications which mimic the real-world scenarios.

Course Title	Design Analysis & Algorithm	Course Code: CSAB4030	
Teaching Scheme		Evaluation Scheme	
Lectures / Tutorial	3 / 1 Hrs Week	Internal Assessment Test	30 Marks
Total Credits	4	End-Semester Examination	70 Marks

Course Objectives:

1. To teach various problem-solving strategies to the students.
2. To make the students study background for algorithm analysis and implementation of various strategies like divide and conquer, Greedy method, Dynamic programming, Backtracking, branch and bound.
3. To enable students, solve the complex networking problems using graph algorithms.
4. To make the learners study different string-matching algorithms and network flow theorem.

Detailed Syllabus:

Unit-I

Algorithm Design Techniques
Divide and Conquer: Basic method, use, Examples – Binary Search, Merge Sort, Quick Sort and their complexity. Heap Sort and its complexity, Dynamic Programming: Basic method, use, Examples – Matrix Chain Manipulation, All pair shortest paths, single source shortest path. Backtracking: Basic method, use, Examples – 8 queens' problem, Graph coloring problem. Greedy Method: Basic method, use, Examples – Knapsack problem, Job sequencing with deadlines.

Unit-II

Lower Bound Theory
O(nlogn) bound for comparison sort, Disjoint set manipulation: Set manipulation algorithm like UNION- FIND, union by rank.

Unit-III

Graph traversal algorithm
Recapitulation, Breadth First Search (BFS) and Depth First Search (DFS) – Classification of edges - tree, forward, back and cross edges – complexity and comparison.

Unit-IV

String matching problem
Different techniques – Naive algorithm, string matching using finite automata, and Knuth, Morris, Pratt (KMP) algorithm with their complexities.

Unit-V

Network Flow
Ford Fulkerson algorithm, Max-Flow Min-Cut theorem (Statement and Illustration).

Text /Reference Books:

1. T. H. Cormen, C. E. Leiserson, R. L. Rivest and C. Stein, "Introduction to Algorithms". 2. A. Aho, J.Hopcroft and J.Ullman "The Design and Analysis of Algorithms".

Course Outcomes:

1. CO1: Select appropriate problem-solving strategies.
2. CO2: Calculate time complexity and space complexity of different algorithms and apply set manipulation algorithms.
3. CO3: Comprehend real world networking problems using graph algorithms.
4. CO4: Apply different string-matching algorithms.
5. CO5: Understand the Network flow algorithms.

Course Title	Computer Networks	**Course Code:** CSAB4040	
Teaching Scheme		**Evaluation Scheme**	
Lectures / Tutorial	**3 / 1 Hrs Week**	**Internal Assessment Test**	**30 Marks**
Total Credits	**4**	**End-Semester Examination**	**70 Marks**

Course Objectives:

1. To make students understand computer networking basics.
2. To develop an understanding of different components of computer networks, various protocols, modern technologies and their applications to the students.
3. To impart the knowledge about the functions of various networking layers.

Detailed Syllabus:

Unit-I

Introduction to Networking

Introduction to computer network, network application, network software and hardware components (Interconnection networking devices), Network topology, protocol hierarchies, design issues for the layers, connection oriented and connectionless services, Reference models: Layer details of OSI, TCP/IP models. Communication between layer.

Unit-II

Physical Layer

Electromagnetic Spectrum, Guided Transmission Media: Twisted pair, Coaxial, Fiber optics. Unguided, media (Wireless Transmission): Radio Waves, Microwave.

Unit-III

Data Link Layer

DLL Design Issues (Services, Framing, Error Control, Flow Control), Error Detection and Correction (Hamming Code, CRC, Checksum), Elementary Data Link protocols, Stop and Wait, Sliding Window (Go Back N, Selective Repeat). Access Control sublayer, Channel Allocation problem, Multiple , access Protocol (Aloha, Carrier SenseMultiple Access (CSMA/CD), (CSMA/CA).

Unit-IV

Network Layer

Network Layer design issues, Communication Primitives: Unicast, Multicast, Broadcast. IPv4 Addressing (classfull and classless), Subnetting, Supernetting design problems, IPv4 Protocol, Network Address Translation (NAT), Routing algorithms: Shortest Path(Dijkastra‘s), Distance Vector Routing, Protocols - ARP,RARP,Token & Leaky bucket algorithms.

Unit-V

Transport & Application Layer

Connection management (Handshake), UDP, TCP, DNS, HTTP, SMTP, Telnet, FTP

Text /Reference Books:

1. Forouzen, "Data Communication and Networking", TMH
2. A.S. Tanenbaum, Computer Networks, Pearson Education
3. W. Stallings, Data and Computer Communication, Macmillan Press
4. Bhavneet Sidhu, An Integrated approach to Computer Networks, Khanna Publishing House
5. Gary R.Wright,W.Richard Stevens "TCP/IP Illustrated,Volume2 The Implementation" Addison- Wesley
6. Michael A. Gallo and William M. Hancock "Computer Communication and Networking Technology" Cengage Learning
7. Anuranjan Misra, “Computer Networks”, Acme Learning
8. G. Shanmugarathinam,” Essential of TCP/ IP”, Firewall Media.

Course Outcomes:

1. CO1:Define basic networking terminologies such as LAN, WAN, TCP/IP, DNS, etc.
2. CO2:Understand layers of the OSI model and their functions
3. CO3:Recognize different types of network devices and their roles.
4. CO4:Discuss different types of protocols and technologies in different layers of OSI model.
5. CO5:Proficiency in subnetting and super netting techniques for efficient IP address allocation and management within networks.

Course Title	Data Visualization	**Course Code:CSAB4050**	
Teaching Scheme		**Evaluation Scheme**	
Lectures / Tutorial	**3 /1 Hrs Week**	**Internal Assessment Test**	**30 Marks**
Total Credits	**4**	**End-Semester Examination**	**70 Marks**

Course Objectives:

1. To introduce the fundamentals of data visualization and its role in enhancing data analysis and interpretation.
2. To familiarize students with popular and free visualization tools like Tableau Public, Google Data Studio, and Python libraries.
3. To develop the ability to create various types of visualizations such as bar charts, line graphs, scatter plots, and dashboards.
4. To teach the principles of effective visual communication, including layout, color use, and ethical representation of data.
5. To empower students to design interactive and story-driven visualizations that convey meaningful insights from real-world datasets.

Detailed Syllabus:

Unit-I

Introduction to Data Visualization: fundamental concepts of data visualization, its importance in data analysis, and the difference between data analysis and data presentation. Students will learn types of data (quantitative and qualitative), visual perception principles, and how to choose the appropriate chart types for different datasets. Case studies from journalism, business, and scientific data visualization will be discussed.

Unit-II

Tools and Technologies for Data Visualization: popular data visualization tools, especially free and open-source ones like Tableau Public, Power BI (Free), Google Data Studio, RAWGraphs, and Python libraries (Matplotlib, Seaborn). Installation and basic use of these tools will be demonstrated. Students will learn about data sources, data importing, and preparing datasets for visualization.

Unit-III

Creating Basic Charts and Graphs
create bar charts, pie charts, histograms, line charts, and scatter plots using selected tools. Emphasis will be on design principles, labeling, color use, and visual clarity. Real-world data will be used to explore practical application scenarios.

Unit-IV

Interactive Dashboards and Storytelling with Data: concept of data storytelling and how to design interactive dashboards that communicate insights clearly. Tools like Google Data Studio and Tableau ublic will be used to create dashboards with filters, interactive charts, and visual narratives. Best practices in layout and user experience will be discussed.

Unit-V

Advanced Visualization Techniques and Ethics
This unit covers advanced techniques such as heatmaps, tree maps, bubble charts, and geospatial maps. It also includes a discussion on the ethical aspects of data visualization, such as misrepresentation of data, visual bias, and the importance of source transparency. Students will also learn about accessibility and inclusivity in design.

Text /Reference Books:

1. "Storytelling with Data: A Data Visualization Guide for Business Professionals"
   Author: Cole Nussbaumer Knaflic, *Publisher:* Wiley, *Year:* 2015
   Note: Practical guide focusing on data storytelling and visualization principles using real-world examples.
2. "Data Visualization: A Practical Introduction", *Author:* Kieran Healy, *Publisher:* Princeton University Press, *Year:* 2018
   Note: Offers a clear introduction using R and ggplot2 but useful for understanding core visualization concepts across tools.
3. "Interactive Data Visualization for the Web", *Author:* Scott Murray
   Publisher: O'Reilly Media, *Year:* 2017 (2nd Edition)
   Note: Covers D3.js and web-based data visualization with a hands-on approach.
4. "The Visual Display of Quantitative Information", *Author:* Edward R. Tufte
   Publisher: Graphics Press, *Year:* 2001 (2nd Edition)
   Note: Classic book focusing on the theory and best practices of visualizing quantitative data.

Assignments

Sr. No.	Practical Title	Free Tools Used
1	Create bar and pie charts using real-world data	Google Sheets, RAWGraphs
2	Develop an interactive sales dashboard	Tableau Public
3	Visualize COVID-19 trends using a time series plot	Power BI Free
4	Design a student performance dashboard	Google Data Studio
5	Plot a geographic data visualization (India map)	Flourish, Datawrapper
6	Use RAWGraphs to visualize hierarchical data as tree maps	RAWGraphs.io
7	Build a bubble chart for product comparison	Google Sheets
8	Create a storytelling dashboard for population data	Tableau Public

Course Outcomes:

CO1:Understand and explain core principles of data visualization and the importance of choosing the right chart types for different data types.
CO2:Use free and open-source tools (e.g., Tableau Public, Google Data Studio, RAWGraphs, Python) to create professional-quality visualizations.
CO3:Create and customize charts, graphs, and dashboards using real-world data for business, academic, or journalistic contexts.
CO4:Demonstrate the ability to tell compelling stories using data, incorporating interactivity and visual design best practices.
CO5:Evaluate visualizations for clarity, ethics, and effectiveness, ensuring accurate and fair representation of data.

Elective-I

Course Title	**Data Warehousing and Data Mining**	**Course Code:CSAB4620**	
Teaching Scheme		**Evaluation Scheme**	
Lectures / Tutorial	**3 Hrs Week**	**Internal Assessment Test**	**30 Marks**
Total Credits	**3**	**End-Semester Examination**	**70 Marks**

Course Objectives:

1. To make students understand the basic concepts of data warehousing.
2. To make students extract knowledge from a data repository for data analysis, frequent pattern, classification and prediction**.**
3. To guide and teach the students to solve business problems by data analysis.
4. To help students to perform predictions based on the given data that can help in long term decision making.

Detailed Syllabus:

Unit-I

Data Warehousing
Introduction, Data lakes, Data marts, Data warehousing Components, Building a Data warehouse, Data warehouse architecture, ETL process, Data Extraction, Cleanup, and Transformation Tools, Metadata Repository.
Introduction to Online Analytical Processing (OLAP), Multidimensional Data Model, Stars, Snowflakes, and Fact Constellations Schemas for multidimensional data models, Multidimensional versus Multi Relational OLAP, OLAP tools operations. Difference between OLTP and OLAP.

Unit-II

Data mining:
Introduction, What Kinds of Data can be mined, Data mining functionalities, Technologies used, Interestingness of Patterns, Integration of a Data Mining System with a Data Warehouse, Issues, Data preprocessing – Data cleaning – Data Integration and Transformation – Data Reduction – Discretization, Data Mining Task Primitives.

Unit-III

Association Rule Mining:
Market basket analysis, Mining Frequent Patterns, Associations and Correlations, Apriori algorithm, Association Rule generation. Introduction to Mining Multilevel Association Rules and Mining Multidimensional Association rules

Unit-IV

Cluster Analysis – Introduction
A categorization of Major clustering methods - Partitioning methods, Hierarchical methods, Grid-based methods, Density-based methods.

Unit-V

Classification and Prediction:
Basic Concepts, Decision Tree, Rule Based Classification, Classification by Back propagation, Associative Classification, Lazy Learners, Other Classification Methods, Prediction.

Text /Reference Books:

1. Ralph Kimball, "The Data Warehouse Lifecycle toolkit', 2nd edition, Wiley India. 2. Han, Kamber, "Data Mining Concepts and Techniques", 2nd edition, Elsevier
2. Reema Theraja “Data warehousing”, Oxford University Press.
3. “Introduction to Data Mining”, 1/e Pang-Ning Tan, Vipin Kumar, Michael Steinbach Pearson
4. Education
5. M.H. Dunham, "Data Mining Introductory and Advanced Topics", Pearson
6. Education. 6.Paulraj Ponniah, "Data Warehousing Fundamentals", Wiley Student edition.
7. “Data Mining For Business Intelligence” Galit Shmueli, Nitin Patel, Peter Bruce; Wiley Student Edition.
8. “Data Warehousing, Data Mining & OLAP " Alex Berson & Stephen J Smith, Tat McGraw Hill.
9. “Data Mining with SQL Server 2008” Jamie McLennan & others, Wiley Indian Edition.
10. "Mastering Data Mining”, M Berry and G. Linoff, Wiley Student Edition. 6) R. KimballI, "The Data Warehouse Toolkit', John Wiley.

Course Outcomes:

At the end of this course students will able to

CO1: Explain the principles behind data warehousing architectures, including the differences between enterprise data warehouses and data marts.

CO2: Illustrate the basics of data mining, technologies used and the steps in data pre-processing.

CO3: Interpret the various techniques used in data mining, such as classification, clustering, association rule mining, and anomaly detection.

CO4: Acquire satisfactory competency in applying clustering techniques, classification and prediction.

CO5: Create predictive models using data mining techniques to support decision-making processes in various domains, such as marketing, finance, or healthcare.

Course Title	E-Commerce	Course Code:CSAB4930	
Teaching Scheme		**Evaluation Scheme**	
Lectures / Tutorial	**3 Hrs Week**	**Internal Assessment Test**	**30 Marks**
Total Credits	**3**	**End-Semester Examination**	**70 Marks**

Course Objectives:

1. Discuss fundamentals of e-commerce, types and applications with learners.
2. To make students evaluate the network infrastructure of ecommerce.
3. Assess business models and EDI of e-commerce.
4. To help students identify the major management challenges for building and using ecommerce websites and learn how to find appropriate solutions to those challenges.
5. To make students learn strategies for e-commerce and electronic payments.

Detailed Syllabus:

Unit-I

Introduction
Definition of Electronic Commerce, E-Commerce: technology and prospects, incentives for engaging in electronic commerce, needs of E-Commerce, advantages and disadvantages, framework, Impact of E- commerce on business, E-Commerce Models.

Unit-II

Network Infrastructure for E-commerce
Internet and Intranet based E-commerce- Issues, problems and prospects, Network Infrastructure, Network Access Equipments, Broadband telecommunication (ATM, ISDN, FRAME RELAY). Mobile Commerce: Introduction, Wireless Application Protocol, WAP technology, Mobile Information device.

Unit-III

Business Models of E-Commerce
Characteristics of Business to Business(B2B), Business to Consumers (B2C), C2C (Consumer-toConsumer), Business to Government (B2G)

Concepts of other models of E-commerce
Business to Consumer E-Commerce process, Business to Business E-Commerce Need and Importance, alternative models of B2B E-Commerce. E-Commerce Sales Product Life Cycle (ESLC) Model

Unit-IV

World Wide Web and E-enterprise
World Wide Web-Reasons for building own website, Benefits of Website, registering a Domain Name, Role of web site in B2C E-commerce; push and pull approaches; Web site design principles.
EDI and paperless trading; Pros & Cons of EDI; Related new technologies use in Ecommerce. Applications of E-commerce and E-enterprise - Applications to Customer Relationship Management-Types of E-CRM, Functional Components of E-CRM.

Unit-V

Electronic Payments Overview, The SET protocol, Payment Gateway, certificate, digital Tokens, Smart card, credit card, magnetic strip card, E-Checks, Credit/Debit card-based EPS, online Banking, EDI Application in business, E- Commerce Law, Forms of Agreement, Govt. policies and Agenda.

Text /Reference Books:

1. Ravi Kalakota, Andrew Winston, "Frontiers of Electronic Commerce", Addison Wesley.

Reference books:

1. Pete Lohsin , John Vacca "Electronic Commerce", New Age International
2. Goel, Ritendra "E-commerce", New Age International
3. Laudon, "E-Commerce: Business, Technology, Society", Pearson Education
4. Bajaj and Nag, "E-Commerce the cutting edge of Business", TMH
5. Turban, "Electronic Commerce 2004: A Managerial Perspective", Pearson Education

Course Outcomes:

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

CO1: Define key e-commerce concepts such as online transactions, digital marketing, and electronic funds transfer.

CO2: Identify the network infrastructure for e-commerce and mobile commerce

CO3: Explore various e-commerce business models such as B2B (Business-to-Business), B2C (Business-toConsumer), C2C (Consumer-to-Consumer), and their respective advantages, challenges, and applications.

CO4: Acquire skills in designing, developing, and maintaining e-commerce websites

CO5: Understand the various algorithms applied for e-commerce and mobile commerce transactions.

Course Title	Python Programming LAB	**Course Code:CSAB4011**	
Teaching Scheme		**Evaluation Scheme**	
Lectures / Tutorial	**4 Hrs Week**	**Internal Assessment Test**	**15 Marks**
Total Credits	**2**	**End-Semester Examination**	**35 Marks**

Course Objectives:

1. **To introduce the fundamentals of Python programming** including syntax, variables, data types, and operators, laying the foundation for developing simple applications.
2. **To provide hands-on experience with control structures** such as loops and conditional statements for implementing logic-based solutions.
3. **To enable students to design and implement functions and modules** to promote modularity, code reusability, and structured programming practices.
4. **To familiarize students with Python data structures** such as lists, tuples, dictionaries, and sets for efficient data manipulation.
5. **To develop problem-solving and debugging skills through practical assignments**, enhancing students' ability to apply programming logic to real-world scenarios.

1. Python program to print "Hello Python".
2. Python program to do arithmetical operations.
3. Python program to find the area of a triangle.
4. Python program to solve quadratic equations.
5. Python program to swap two variables.
6. Python program to generate a random number.
7. Python program to convert kilometers to miles.
8. Python program to convert Celsius to Fahrenheit.
9. Python program to display calendar.
10. Python Program to Check if a Number is Positive, Negative or Zero.
11. Python Program to Check if a Number is Odd or Even.
12. Python Program to Check Leap Year.
13. Python Program to Check Prime Number.
14. Python Program to Print all Prime Numbers in an Interval.
15. Python Program to Find the Factorial of a Number.
16. Python Program to Display the multiplication Table.
17. Python Program to Print the Fibonacci sequence.
18. Python Program to Check Armstrong Number.
19. Python Program to Find Armstrong Number in an Interval.
20. Python Program to Find the Sum of Natural Numbers.
21. Python program to check if the given number is a disarium Number.
22. Python program to print all disarium numbers between 1 to 100.
23. Python program to check if the given number is Happy Number.
24. Python program to print all happy numbers between 1 and 100.
25. Python program to find the frequency of each element in the array.
26. Python program to left rotate the elements of an array.

27. Python program to print the duplicate elements of an array.
28. Python program to print the elements of an array.
29. Python program to print the elements of an array in reverse order.
30. Python program to print the elements of an array present in an even position.
31. Python program to print the elements of an array present in an odd position.

Course Outcomes:

After completion of this subject students will be able to

CO1: Apply Python syntax and programming constructs to solve basic and intermediate level problems.

CO2: Develop programs using conditional statements, loops, and functions to handle different scenarios and requirements.

CO3: Create Python programs to solve complex problems such as finding the area of a triangle, solving quadratic equations, and checking for prime numbers.

CO4: Apply critical thinking to evaluate program logic, identify potential errors, and optimize code for performance and readability.

CO5: Develop programs to perform advanced operations such as generating random numbers, converting units, and working with arrays efficiently.

Course Title	Java Programming LAB	**Course Code:CSAB4021**	
Teaching Scheme		**Evaluation Scheme**	
Lectures / Tutorial	**4 Hrs Week**	**Internal Assessment Test**	**15 Marks**
Total Credits	**2**	**End-Semester Examination**	**35 Marks**

Course Objectives:

1. **To provide hands-on experience with core Java programming concepts**, including variables, data types, control structures, and arrays.
2. **To develop object-oriented programming (OOP) skills** using Java features like classes, objects, inheritance, polymorphism, abstraction, and encapsulation.
3. **To familiarize students with exception handling, file I/O operations, and multithreading** in Java applications.
4. **To introduce GUI development and event-driven programming** using Java Swing or JavaFX.
5. **To build problem-solving abilities by developing real-world applications** using Java and applying debugging and testing techniques.

List of Experiments

32. Swapping of two numbers without accepting numbers from the user.
33. Find out the largest of two numbers using command line arguments.
34. Print class of students according to given range using switch case.
35. Find GCD and LCM of two positive numbers using a while loop.
36. Display table of given number using for loop.
37. Java Program to check whether input character is vowel or consonant
38. Print the upper triangle using a nested for loop.

```
        *
    * *
    * * *
    * * * *
```

39. Java Program to calculate simple interest and compound interest.
40. Java program to find occurrence of a character in a String.
41. Write a class Circle with methods getdata() and area(). Create an object of that class, find the area and display the result.
42. Calculate absolute value of given number using method overloading.
43. Calculate addition of two numbers using constructor overloading.
44. Write a program to accept n integers from user into an array and display the average of these numbers.
45. Write a program to sort elements of an array in ascending order using bubble sort.
46. Write a program to find and display the sum of diagonal elements of a square matrix.
47. Write a program to implement single inheritance using super keywords.

48. Write a program to add two numbers using single inheritance such that the base class method must accept 2 numbers from the user and the derived class method must add these numbers and display the sum.
49. Write a program to find the area of a circle using single inheritance such that the base class method must accept the radius from the user and the derived class method must calculate and display the area.
50. Write a program to implement multiple inheritance.
51. Write a program to print 1A2B3C4D5E6F7G8H9I10J using thread.

Course Outcomes:

After completion of this subject students will be able to

CO1: Demonstrate proficiency in implementing Java programs to perform various tasks such as swapping numbers, finding the largest number, calculating interest, and manipulating arrays.

CO2: Develop programs using control structures like loops, conditionals, and methods to handle different scenarios and requirements.

CO3: Create Java programs to solve complex problems such as calculating GCD and LCM, sorting arrays, and working with matrices.

CO4: Develop programs to perform advanced operations such as working with inheritance, threads, and method overloading efficiently.

CO5: Apply problem-solving skills to design and implement programs that involve mathematical calculations, data manipulation, and object-oriented concepts.

	BCA- AI ML 3 Year- Semester V										
Type		**Course Title**	**Hours/ Week**			**Theory Marks**		**Practical Marks**		**Total Marks**	**Credit**
			L	**T**	**P**	**IA**	**ESE**	**IA**	**ESE**		
DC	CSAB5050	Open Source Technology	3	1	-	30	70	-	-	100	4
DC	CSAB5060	Internet of Things	3	-	-	30	70	-	-	100	3
DC	CSAB5030	Cloud Computing	3	-	-	30	70	-	-	100	3
DC	CSAB5070	Network Security	3	-	-	30	70	-	-	100	3
DC	CSAB5080	Natural Language Processing (NLP)	3	1	-	30	70			100	4
DE	**	Elective 1	3	-	-	30	70	-	-	100	3
DC	CSAB4011	Cloud Computing Lab	-	-	4	-	-	15	35	50	2
DC	CSAB4021	Open Source Technology Lab	-	-	4	-	-	15	35	50	2
DC	CSAB5083	Project Work-I			8			15	35	50	4
SEC		Verbal Ability									
		TOTAL	**18**	**2**	**16**	**180**	**420**	**45**	**105**	**750**	**28**

Elective-2 CSAB 5410/CSAB5420
1. Software Project Management
2. ERP(Enterprise Resource planning)

Course Title	Open Source Technologies	**Course Code:** CSAB5050	
Teaching Scheme		**Evaluation Scheme**	
Lectures / Tutorial	**3 / 1 Hrs Week**	**Internal Assessment Test**	**30 Marks**
Total Credits	**4**	**End-Semester Examination**	**70 Marks**

Course Objectives:

1. To make students understand the difference between open source software and commercial software.
2. Familiarity with Linux operating system.
3. To guide students in development of web applications using open source web technologies like Apache, MySql and PHP (LAMP/XAMP).

Detailed Syllabus:

Unit-I

BASIC PHP

Web Server, Apache, PHP, Data Types, User defined Variables, Constants, Operators, Control Structures, User defined Functions, Directory Functions, File system Functions, Arrays, String Functions, Date and Time Functions, Mathematical Functions, Miscellaneous Functions.

Unit-II

Advanced PHP with MySQL

Exceptions handling, Error Handling Functions, Predefined Variables, Cookies, Sessions, COM, DOM, CURL, SOAP, Classes and Objects-Mail Function, URL Functions. PHP with MySQL: PHP MySQL Functions, Database driven application.

Unit-III

Advanced PHP with AJAX, SEO and CMS

PHP with AJAX: Introducing Ajax-Ajax Basics-PHP and Ajax-Database Driven Ajax. PHP with SEO: Basic SEO-Provocative SE Friendly URLs-Duplicate Content- CMS: Word press Creating an SE-Friendly Blog.

Unit-IV

Basic PERL

Introduction-Scalar Data- Lists and Arrays -Subroutines-Input and Output- Hashes-Regular Expressions Control Structures-Perl Modules-File Tests.

Unit-V

Advanced PERL

Directory Operations-Strings and Sorting-Smart Matching-Process Management- Advanced Perl Techniques.

Text /Reference Books:

1. James Lee and Brent Ware ,"Open Source Web Development with LAMP using Linux, Apache, MySQL, Perl and PHP", , Dorling Kindersley(India) Pvt. Ltd, 2008.
2. Eric Rosebrock, Eric Filson ,"Setting Up LAMP: Getting Linux, Apache, MySQL, and PHP and working Together", Published by John Wiley and Sons, 2004.

Course Outcomes:

1. CO1: Apply basic PHP scripts to perform tasks using the concepts covered, such as handling user input, manipulating files and directories, working with arrays, strings, and dates.
2. CO2: Develop dynamic web applications that interact with MySQL databases using PHP scripts.
3. CO3: Analyze SEO strategies and implement them in PHP websites to improve search engine rankings and visibility.
4. CO4: Understand how Perl language constructs and features enable various programming tasks.
5. CO5: Develop Perl scripts that leverage advanced techniques to solve complex problems efficiently.

Course Title	Internet of Things	Course Code: CSAB5060	
Teaching Scheme		**Evaluation Scheme**	
Lectures / Tutorial	**3 Hrs Week**	**Internal Assessment Test**	**30 Marks**
Total Credits	**3**	**End-Semester Examination**	**70 Marks**

Course Objectives:

The objectives of this course are:

1. Understand the Fundamentals of IoT:
2. Explore IoT Hardware and Software Components
3. Design and Develop IoT Applications
4. Understand Networking and Communication Protocols
5. Analyze IoT Security and Data Management Challenges

Detailed Syllabus:

Unit-I

What is the IoT and why is it important? Elements of an IoT ecosystem, Technology drivers, Business drivers, Trends and implications, Overview of Governance, Privacy and Security Issues.

Unit-II

Protocol Standardization for IoT – Efforts – M2M and WSN Protocols – SCADA and RFIDProtocols –

Issues with IoT Standardization – Unified Data Standards – Protocols – IEEE802.15.4–BACNet Protocol– Modbus – KNX – Zigbee– Network layer – APS layer – Security

Unit-III

Architecture

IoT Open source architecture (OIC)- OIC Architecture & Design principles- IoT Devices and deployment models- IoTivity : An Open source IoT stack - Overview- IoTivity stack architecture- Resource model and Abstraction.

Unit-IV

Web of things

Web of Things versus Internet of Things – Two Pillars of the Web – Architecture Standardization for WoT– Platform Middleware for WoT – Unified Multi Tier WoT Architecture – WoT Portals and Business Intelligence.

Unit-V

Applications

IoT applications for industry: Future Factory Concepts, Brownfield IoT, Smart Objects, Smart Applications. Study of existing IoT platforms /middleware, IoT- A, Hydra etc.

Text /Reference Books:

1. Jan Holler, Vlasios Tsiatsis, Catherine Mulligan, Stefan Avesand, Stamatis Karnouskos, David Boyle, “From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence”, 1st Edition, Academic Press, 2014. Syllabus for Bachelor of Technology Computer Engineering.
2. Vijay Madisetti and Arshdeep Bahga, “Internet of Things (A Hands-onApproach)”, 1st Edition, VPT, 2014.
3. Francis daCosta, “Rethinking the Internet of Things: A Scalable Approach to Connecting Everything”, 1st Edition, Apress Publications, 2013
4. Cuno Pfister, Getting Started with the Internet of Things, O‟Reilly Media, 2011, ISBN: 978-1-4493- 9357-1.

Course Outcomes:

1. CO1: Understand concepts of Internet of Things (IoT).
2. CO2: Discuss the challenges and obstacles faced in standardizing protocols for IoT.
3. CO3: Identify the design principles behind IoT architecture.
4. CO4: Analyze the strengths and limitations of WoT compared to traditional IoT architectures.
5. CO5: Analyze the potential impact of Future Factory Concepts on industrial operations, productivity, and sustainability.

Course Title	Cloud Computing	**Course Code: CSAB5030**	
Teaching Scheme		**Evaluation Scheme**	
Lectures / Tutorial	**3 Hrs Week**	**Internal Assessment Test**	**30 Marks**
Total Credits	**3**	**End-Semester Examination**	**70 Marks**

Course Objectives:

The objectives of this course are

1. To understand the basic concepts and models of cloud computing.
2. To explore virtualization and its role in cloud infrastructure.
3. To learn about different cloud service providers and their offerings.
4. To develop and deploy applications using cloud platforms.
5. To study security, privacy, and management aspects of cloud computing.

Detailed Syllabus:

Unit-I

Introduction to Cloud Computing:
Virtualization Concepts, Cloud Computing Fundamental: Overview of Computing Paradigm, Evolution of cloud computing, Defining cloud computing, Components of a computing cloud, Essential Characteristics of Cloud Computing, Cloud Taxonomy.

Unit-II

Cloud Computing Architectural Framework:
Cloud architectural principles, Role of Networks in Cloud computing, Role of Web services, Benefits and challenges to Cloud architecture, Cloud Service Models, cloud computing vendors, Cloud Services Management, Performance and scalability of services, tools and technologies used to manage cloud services deployment.

Unit-III

Exploiting Cloud Services:
Software as a Service(SaaS): Introduction to SaaS, Inspecting SaaS technologies, Implementing web services, Deploying Infrastructure as a Service(IaaS): Introduction to IaaS, Scalable server clusters, Machine Image, Virtual Machine (VM). Elastic storage devices, Data storage in cloud computing, Delivering Platform as a Service(PaaS): Introduction to PaaS, Service Oriented Architecture (SOA), Cloud Platform and Management, Hardware-as-a-service: HaaS.

Unit-IV

Cloud Application Development:
Role of business analyst, Technical architecture considerations, Service creation environments to develop cloud based applications, Technologies and the processes required when deploying web services; Deploying a web service from inside and outside a cloud architecture, advantages and disadvantages, Cloud Economics.

Unit-V

Cloud Security and Risk Management:

Cloud Security: Understanding cloud based security issues and threats, Data security and Storage, Identity & Access Management, Risk Management in cloud, Governance and Enterprise Risk Management.

Analysis on Case study:

Business Case: Business case evaluation criteria, Business outcomes examples, Case Studies: Case Study on Open Source & Commercial Cloud: Eucalyptus, Microsoft Windows Azure, Amazon EC2, Amazon Elastic Block Storage – EBS. Google Cloud Infrastructure, MapReduce. AWS, Simple Storage Service – S3

Text /Reference Books:

1. Distributed and Cloud Computing: Clusters, Grids, Clouds, and the Future Internet by Kai Hwang, Jack Dongarra & Geoffrey C. Fox., Morgan Kaufmann Publishers, 2012.
2. Cloud Computing Bible, Barrie Sosinsky, WileyIndia, 2010.
3. Cloud Computing: Principles and Paradigms, Editors: Rajkumar Buyya, James Broberg, Andrzej M. Goscinski, Wile, 2011.
4. Cloud Computing: Principles, Systems and Applications, Editors: Nikos Antonopoulos, Lee Gillam, Springer, 2012.
5. Cloud Security: A Comprehensive Guide to Secure Cloud Computing, Ronald L. Krutz, Russell Dean Vines, Wiley-India, 2010.
6. Gautam Shroff, Enterprise Cloud Computing Technology Architecture Applications [ISBN: 978-0521137355].

Course Outcomes:

1. CO1: Explain the essential characteristics of cloud computing and categorize different components of a computing cloud.
2. CO2: Evaluate the benefits and challenges associated with cloud architecture.
3. CO3: Implement software as a service (SaaS) and infrastructure as a service (IaaS) technology.
4. CO4: Critique the advantages and disadvantages of deploying web services within a cloud architecture.
5. CO5: Analyze business case evaluation criteria and business outcomes examples related to cloud computing.

Course Title	Network Security	**Course Code: CSAB5070**	
Teaching Scheme		**Evaluation Scheme**	
Lectures / Tutorial	**3 Hrs Week**	**Internal Assessment Test**	**30 Marks**
Total Credits	**3**	**End-Semester Examination**	**70 Marks**

Course Objectives:

1. Provide knowledge of Cryptography to the learners.
2. To provide students' knowledge of Symmetric and Asymmetric Algorithms.
3. Give insight on Message Authentication and Hash Functions.
4. To help the students understand the concepts of Security Policies and Security HandshakePitfalls.

Detailed Syllabus:

Unit-I

Introduction:
Attacks, Services and Mechanisms, Security Attacks, Security Services, Integrity check, digital Signature, authentication, hash algorithms.

Unit-II

Key Cryptography:
Block Encryption, DES rounds, S-Boxes IDEA: Overview, comparison with DES, Key expansion, IDEA rounds, Uses of Secret key Cryptography; ECB, CBC, OFB, CFB, Multiple encryptions DES.

Unit-III

Functions and Message Digests:
Length of hash, uses, algorithms (MD2, MD4, MD5, SHS) MD2: Algorithm (Padding, checksum, passes.) MD4 and 5: algorithm (padding, stages, digest computation.) SHS: Overview, padding, stages.
Public key Cryptography:
Algorithms, examples, Modular arithmetic (addition, multiplication, inverse, and exponentiation) RSA: generating keys, encryption and decryption. Other Algorithms: PKCS, Diffie-Hellman, El-Gamal signatures, DSS, Zero-knowledge signatures.

Unit-IV

Authentication:
Password Based, Address Based, Cryptographic Authentication. Passwords in distributed systems, on-line vs offline guessing, storing. Cryptographic Authentication: passwords as keys, protocols, KDC's Certification Revocation, Interdomain, groups, delegation. Authentication of People: Verification techniques, passwords, length of passwords, password distribution, smart cards, biometrics.

Unit-V

Security Policies and Security Handshake Pitfalls:
What is security policy, high and low level policy, user issues? Protocol problems, assumptions, Shared secret protocols, public key protocols, mutual authentication, reflection attacks, use of timestamps, nonce and sequence numbers, session keys, one-and two-way public key based authentication.

Text /Reference Books:

1. Atul Kahate, Cryptography and Network Security, McGraw Hill. Kaufman, c., Perlman, R., and Speciner, M., Network Security, Private Communication in a public world, 2nd ed., Prentice Hall PTR., 2002.
2. Stallings, Cryptography and Network Security: Principles and Practice, 3rd ed., Prentice Hall PTR., 2003.
3. Stallings, W. Network security Essentials: Applications and standards, Prentice Hall, 2000.
4. Cryptography and Network Security; McGraw Hill; Behrouz A Forouzan.
5. Information Security Intelligence Cryptographic Principles and App. Calabrese Thomson.
6. Securing A Wireless Network Chris Hurley SPD.

Course Outcomes:

1. CO1: Understand the basic concept of network security.
2. CO2: Illustrate secret key cryptography algorithms.
3. CO3: Apply hash functions and public key cryptography to the plain text.
4. CO4: Understand various types of authentication techniques.
5. CO5: Evaluate security policies and their pitfalls

Course Title	Natural Language Processing (NLP)	**Course Code:** CSAB5080	
Teaching Scheme		**Evaluation Scheme**	
Lectures / Tutorial	**3 / 1 Hrs Week**	**Internal Assessment Test**	**30 Marks**
Total Credits	**3**	**End-Semester Examination**	**70 Marks**

Course Objectives:

1. To familiarize the concepts and techniques of Natural language Processing for analyzing words based on Morphology and CORPUS.
2. To Perform POS tagging for a given natural language using modeling technique based on the structure of the language.
3. To relate mathematical foundations, Probability theory with Linguistic essentials such as syntactic and semantic analysis of text.
4. To apply the Statistical learning methods and cutting-edge research models from deep learning.
5. To Check a current method for statistical approaches to machine translation

Detailed Syllabus:

Unit-I : NLP INTRODUCTION AND TEXT PREPROCESSING

Introduction to NLP - Various stages of NLP –The Ambiguity of Language:Why NLP Is Difficult, Parts of Speech: Nouns and Pronouns, Words:Determiners and adjectives, verbs, Phrase Structure, Statistics Essential Information Theory: Entropy, perplexity, the relation to language, Cross entropy. Character Encoding, Word Segmentation, Sentence Segmentation, Introduction to Corpora, Corpora Analysis.

Unit-II : MORPHOLOGY AND LANGUAGE MODELING

Inflectional and Derivation Morphology, Morphological analysis and generation using Finite State Automata and Finite State transducer, Words: Collocations- Frequency-Mean and Variance –Hypothesis testing: The t test, Hypothesis testing of differences, Pearson's chi-square test, Likelihood ratios. Statistical Inference: n-gram Models over Sparse Data: Bins: Forming Equivalence Classes- N gram model - Statistical Estimators- Combining Estimators.

Unit-III : WORD SENSE DISAMBIGUATION AND MARKOV MODEL

Supervised Disambiguation: Bayesian classification, An information theoretic approach, Dictionary-Based Disambiguation: Disambiguation based on sense, Thesaurus based disambiguation, Disambiguation based on translations in a second-language corpus. Hidden Markov model, Fundamentals, Probability of properties, Parameter estimation, Variants, Multiple input observation- Applying HMMs to POS tagging, Applications of Tagging.

Unit-IV : CONTEXT FREE GRAMMARS AND DISCOURSE STRUCTURE ANALYSIS

The Probability of a String, Problems with the Inside-Outside Algorithm, parsing for disambiguation, Tree banks, parsing models vs. language models, Phrase structure grammars and dependency, Lexicalized models using derivational histories, Dependency-based models- Discourse- Reference resolution, constraints on co-reference, algorithm for pronoun resolution,text coherence, discourse structure.

Unit-V : SYNTAX, SEMANTICS AND RECENT TRENDS

Shallow Parsing and Chunking, Shallow Parsing with Conditional Random Fields (CRF), Lexical Semantics, WordNet, Thematic Roles, Semantic Role, Labelling with CRFs. Statistical Alignment and Machine Translation, Text alignment, Word alignment, Information extraction, Text mining, Information Retrieval, NL interfaces, Sentimental Analysis, Question Answering Systems.

Text /Reference Books:

1. James Allen (2004)– "Natural Language Understanding ", Pearson Education, 2004.
2. Daniel Jurafsky and James H Martin (2018)" Speech and Language Processing: An introduction to Natural Language Processing, Computational Linguistics and Speech Recognition", Prentice Hall, 2nd Edition.
3. Nitin Indurkhya, Fred J. Damerau(2010) "Handbook of Natural Language Processing", Second Edition, CRC Press.
4. Hobson lane, Cole Howard, Hannes Hapke(2019), "Natural language processing in action" MANNING Publications.

Assignments:

Assignment 1: Text Preprocessing Basics

Aim: To perform basic text preprocessing operations on a given text corpus.

Tasks:

- Convert text to lowercase
- Remove punctuation and special characters
- Tokenize the text into words
- Remove stopwords
- Perform stemming and lemmatization

Tools: Python, NLTK/spaCy
Expected Output: Cleaned tokens after each preprocessing step.

Assignment 2: Bag of Words (BoW) and Term Frequency

Aim: To convert a text corpus into a numerical representation using BoW and TF.

Tasks:

- Create a sample dataset with 3–5 sentences
- Generate a Bag of Words model using `CountVectorizer`
- Generate Term Frequency using `TfidfVectorizer`
- Display the resulting matrix and vocabulary

Tools: Python, Scikit-learn
Expected Output: Feature matrix showing word frequency or TF-IDF values.

Assignment 3: Named Entity Recognition (NER)

Aim: To extract named entities (e.g., person, organization, location) from text.

Tasks:

- Load a paragraph using spaCy
- Perform NER to identify entities like PERSON, ORG, GPE
- Display entities along with their labels

- Highlight them in the text (optional visualization)

Tools: Python, spaCy
Expected Output: List of named entities with their types.

Assignment 4: Sentiment Analysis

Aim: To analyze the sentiment (positive, negative, neutral) of a given sentence or review.

Tasks:

- Load a set of movie/product reviews
- Use a pretrained sentiment analysis model or TextBlob
- Classify each review into Positive, Negative, or Neutral
- Display sentiment scores with classification

Tools: Python, TextBlob / VaderSentiment / HuggingFace
Expected Output: Table showing reviews and their sentiment classifications.

Assignment 5: POS Tagging and Chunking

Aim: To perform Part-of-Speech (POS) tagging and identify noun phrases.

Tasks:

- Load text and tokenize sentences
- Perform POS tagging
- Define grammar for noun phrase chunking (using RegEx)
- Use `nltk.RegexpParser` to extract chunks

Tools: Python, NLTK
Expected Output: Tagged text and noun phrase chunks.

Course Outcomes:

1. CO1 : Apply the principles and Process of Human Languages English etc. Languages using computers.
2. CO2 : Realize semantics and pragmatics of English language for text processing and Create CORPUS linguistics based on digestive approach (Text Corpus method).
3. CO3 : Perform POS tagging for a given natural language and select a suitable language modelling technique based on the structure of the language.
4. CO4 : Demonstrate the state-of-the-art algorithms and techniques for text-based processing of natural language with respect to morphology.
5. CO5 : Develop a Statistical Methods for Real World Applications and explore deep learning-based NLP and Check current methods for statistical approaches to machine translation.

Elective - II			
Course Title	**Software Project Management**	**Course Code:** CSAB5410	
Teaching Scheme		**Evaluation Scheme**	
Lectures / Tutorial	**3 Hrs Week**	**Internal Assessment Test**	**30 Marks**
Total Credits	**3**	**End-Semester Examination**	**70 Marks**

Course Objectives:

1. To make students understand the Software Project Planning and Evaluation techniques.
2. To plan and manage projects at each stage of the software development life cycle (SDLC).
3. To learn about the activity planning and risk management principles.
4. To make the students manage software projects and control software deliverables.
5. To help students develop skills to manage the various phases involved in project management and people management.
6. 6. To deliver successful software projects that support organization‘s strategic goals.

Detailed Syllabus:

Unit-I

Project Evaluation and Project Planning
Importance of Software Project Management – Activities – Methodologies – Categorization of Software Projects – Setting objectives – Management Principles – Management Control – Project portfolio Management – Cost-benefit evaluation technology – Risk evaluation – Strategic program Management – Stepwise Project Planning.

Unit-II

Project Life Cycle and Effort Estimation
Software process and Process Models – Choice of Process models – Rapid Application development – Agile methods – Dynamic System Development Method – Extreme Programming– Managing interactive processes – Basics of Software estimation – Effort and Cost estimation techniques – COSMIC Full function points – COCOMO II – a Parametric Productivity Model.

Unit-III

Activity Planning and Risk Management
Objectives of Activity planning – Project schedules – Activities – Sequencing and scheduling – Network Planning models – Formulating Network Model – Forward Pass & Backward Pass techniques – Critical path (CRM) method – Risk identification – Assessment – Risk Planning –Risk Management – – PERT technique – Monte Carlo simulation – Resource Allocation – Creation of critical paths – Cost schedules.

Unit-IV

Project Management and Control
Framework for Management and control – Collection of data – Visualizing progress – Cost monitoring – Earned Value Analysis – Prioritizing Monitoring – Project tracking – Change control – Software Configuration Management – Managing contracts – Contract Management.

Unit-V

Staffing in Software Projects

Managing people – Organizational behaviour – Best methods of staff selection – Motivation – The Oldham – Hackman job characteristic model – Stress – Health and Safety – Ethical and Professional concerns – Working in teams – Decision making – Organizational structures – Dispersed and Virtual teams – Communications genres – Communication plans – Leadership

Text /Reference Books:

1. Bob Hughes, Mike Cotterell and Rajib Mall: Software Project Management – Fifth Edition, Tata McGraw Hill, New Delhi, 2012.

References:

1. Robert K. Wysocki ―Effective Software Project Management – Wiley Publication, 2011.
2. Walker Royce: ―Software Project Management- Addison-Wesley, 1998.
3. Gopalaswamy Ramesh, ―Managing Global Software Projects – McGraw Hill Education (India), Fourteenth Reprint 2013.

Course Outcomes:

CO1: Explain the activities, methodologies, and categorization of software projects.

CO2: Apply various effort and cost estimation techniques in software projects.

CO3: Understand the concepts of project schedules, sequencing, and scheduling.

CO4: Analyze the importance of change control and software configuration management.

CO5: Analyze stress management techniques and ethical considerations.

Course Title	Enterprise Resource Planning	Course Code: CSAB5420	
Teaching Scheme		Evaluation Scheme	
Lectures / Tutorial	3 Hrs Week	Internal Assessment Test	30 Marks
Total Credits	3	End-Semester Examination	70 Marks

Course Objectives:

1. To make students understand the technical aspects and life cycle of ERP systems.
2. To understand the steps and activities in ERP.
3. To illustrate students how to identify and describe different types of ERP system.
4. To make the learners understand tools and methodology used for designing ERP for an Enterprise.

Detailed Syllabus:

Unit-I

Introduction to Enterprise Resource Planning (ERP)
Information System and Its Components, Value Chain Framework, Organizational Functional Units, Evolution of ERP Systems, Role of ERP in Organization, Three-Tier Architecture of ERP system.

Unit-II

ERP and Implementation
ERP implementation and strategy, Implementation Life cycle, Pre- implementation task, requirement definition, implementation Methodology.

Unit-III

ERP Business Modules
Finance, manufacturing, human resources, quality management, material management, marketing, Sales distribution and service. Case study on Supply Chain management (SCM), Customer relationship Management (CRM)

Unit-IV

Introduction to ERP related Technologies
Business Process Re-engineering (BPR), Data warehousing, Data **Mining**, On-line Analytical Processing (OLAP), Product Life Cycle Management (PLM), Geographical Information Management ,RFID, QR Code , Bar Coding, E-commerce and their application in Enterprise planning.

Unit-V

Extended ERP and security issues
Enterprise application Integration (EAI), open source ERP, cloud ERP, Managing ERP Securities: Types of ERP security Issues, System Access security, Data Security and related technology for managing data security.

Text /Reference Books:

1. Alexis Leon, ERP Demystified: II Edition, Tata McGraw Hill.
2. Rajesh Ray, Enterprise Resource Planning, Text and cases, Tata McGraw Hill.
3. Sandeep Desai, Abhishek Srivastava, ERP to E^2 ERP: A Case study approach, PHI. 4. Jyotindra Zaveri, Enterprise Resource Planning, Himalaya Publishing House, 2012.

Reference Books:

1. V.K. Garg & N.K. Venkatakrishnan, Enterprise Resource Planning: concepts & practices, by ; PHI.
2. Supply Chain Management Theories & Practices: R. P. Mohanty, S. G. Deshmukh, - Dreamtech Press.
3. Enterprise wide resource planning: Theory & practice: by Rahul Altekar, PHI
4. Customer Relationship Management, Concepts and cases, Second Edition.

Course Outcomes:

CO1: Understand the value chain framework and the role of ERP in organizational functional units.

CO2: Understand ERP implementation strategies and methodologies.

CO3: Apply ERP business modules to solve case studies on supply chain management and customer relationship management.

CO4: Understand the applications of technologies such as RFID, QR code, and e-commerce in enterprise planning.

CO5: Understand different types of ERP security issues and their implications.

Course Title	Open Source Technology Lab	**Course Code:**CSAB5051	
Teaching Scheme		**Evaluation Scheme**	
Practical per week	**4 Hrs Week**	**Internal Assessment Test**	**15 Marks**
Total Credits	**2**	**End-Semester Examination**	**35 Marks**

List of Experiments:

1. Linux command line: File System, Process Management User Administration
2. Setting Up Web server, DNS server, FTP Servers
3. Working with IPTABLES, OpenVAS
4. Version Control
5. Working with Drupal
6. Shell Script
7. Android Setup
8. Programming in Android
9. Programming in Android

Course Outcome

1. CO1: Apply techniques and commands for file system management, process management, user administration, web server setup, DNS server setup, FTP server setup, and firewall management.
2. CO2: Create shell scripts to automate system administration tasks, improve productivity, and enhance system security.
3. CO3: Analyze security vulnerabilities and risks using tools like OpenVAS and implement appropriate measures to mitigate them.
4. CO4: Apply critical thinking to troubleshoot server and network issues, optimize system performance, and enhance security measures.
5. CO5: Develop basic Android applications using programming languages and tools specific to Android development.

Course Title	Cloud Computing Lab	Course Code: CSAB5031	
Teaching Scheme		**Evaluation Scheme**	
Practicals/Week	**4 Hrs Week**	**Internal Assessment Test**	**15 Marks**
Total Credits	**2**	**End-Semester Examination**	**35 Marks**

Course Objectives:

1. Understand the fundamentals and environment of the R programming language.
2. Work with various R data structures and control flow mechanisms.
3. Perform data import, cleaning, transformation, and manipulation using R.
4. Create effective data visualizations using R libraries like ggplot2.
5. Apply basic statistical and machine learning techniques using R for data analysis.

List of Eexperiments:

1. Install Oracle Virtual box and create two VMs on your laptop.
2 Install Turbo C in guest OS and execute C program.
3 Test ping command to test the communication between the guest OS and Host OS 4 Install Hadoop single node setup.
5 Hopkinson‘s test on DC shunt machines
6 Develop hadoop application to count no of characters, no of words and each character frequency.
7 Develop a hadoop application to process given data and produce results such as finding the year of maximum usage, year of minimum usage.
8 Establish an AWS account. Use the AWS Management Console to launch an EC2 instance and connect to it.
9 Design a protocol and use Simple Queue Service(SQS)to implement the barrier synchronization after the first phase
10 Use the Zookeeper to implement the coordination model in Problem 9.
11 Develop a Hello World application using Google App Engine 12 Develop a Guestbook Application using Google App Engine 13 Develop a Windows Azure Hello World application using.
14 Create a Mashup using Yahoo! Pipes.

Course Outcomes:

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

1. CO1: Apply techniques to install and configure virtualization software
2. CO2: Create and deploy applications on cloud platforms like AWS, Google App
3. CO3: Apply knowledge of distributed computing concepts to design and implement
4. CO4: Design and implement distributed applications using technologies like Hadoop
5. CO5: Design and implement mashups using services like Yahoo! Pipes to aggregate and manipulate data from multiple sources, providing useful insights or functionalities for end-users.

Course Title	Project Work- I	Course Code: CSAB5083	
Teaching Scheme		**Evaluation Scheme**	
Practicals/Week	**8 Hrs/Week**	**Internal Assessment Test**	**15 Marks**
Total Credits	**4**	**End-Semester Examination**	**35 Marks**

Course Objectives:

1. To expose students to product development cycle using industrial experience, use of state of art technologies.
2. To encourage and expose students to participation in National/International paper presentation activities and funding agencies for sponsored projects.
3. Exposure to Learning and knowledge access techniques using Conferences, Journal papers and anticipation in research activities.

Reviews1: Based on Implementation (30% implementation expected)

Reviews 2: Complete Project and Testing

Project Exhibition: All final students must see all the projects in the exhibition The group will submit at the end of semester VI.

a) The Workable project.

b) Project report (in Latex/Lyx/latest Word) in the form of a bound journal complete in all respects – 1 copy for the Institute, 1 copy for guide and 1 copy for each student in the group for certification. The project report contains the details.

1. Problem definition

2. Requirement specification

3. System design details (UML diagrams)

4. System implementation – code documentation – data flow diagrams/ algorithm, protocols used.

5. Test result and procedure – test report as per ATP.

6. Conclusions.

7. Appendix a. Tools used b. References c. Papers published/certificates Plagiarism Report of paper and project report from any open source tool One paper should be published in a reputed International conference/International journal.

Course Outcome
After the completion of the course, student will be able to CO1: Discover potential research areas in the field of IT. CO2: Conduct a survey of several available literatures in the preferred field of study CO3: Compare and contrast the several existing solutions for research challenges. CO4: Demonstrate an ability to work in teams and manage the conduct. CO5: Formulate and propose a plan for creating a solution for the research plan identified. CO6: Report and present the findings of the study conducted in the preferred domain.

	BCA- AI ML 3 Year - Semester VI										
Type		Course Title	Hours/ Week			Theory Marks		Practical Marks		Total Marks	Credit
			L	T	P	IA	ESE	IA	ESE		
DC	CSAB6010	IT Infrastructure Management	3	1	0	30	70			100	4
DC	CSAB6020	Chatbot Development Basics	3	1	0	30	70			100	4
DE	**	Elective 3	3	-	0	30	70			100	3
DC	CSAB6023	Project Work-II	0	0	16	-	-	50	50	100	8
		TOTAL	**9**	**2**	**16**	**90**	**210**	**50**	**50**	**400**	**19**

Elective-3 CSAB6910/ CSAB6610
Web Dersigning
Data Analytics

Course Title	IT Infrastructure Management	**Course Code:** CSAB6010	
Teaching Scheme		**Evaluation Scheme**	
Lectures / Tutorial	**3/1 Hrs Week**	**Internal Assessment Test**	**30 Marks**
Total Credits	**4**	**End-Semester Examination**	**70 Marks**

Course Objectives:

1. To make students understand research, reporting and presentation approaches using the latest ICT tools to examine and critically analyze a combination of the technical and management issues in contemporary infrastructure management, with a focus on business alignment.
2. IT infrastructure Management evaluates new ICTs and case studies in the context of enterprise architecture to the learners.
3. It is suitable for combinations of students in information technology, business administration and electronic commerce.

Detailed Syllabus:

Unit-I

Definitions, Infrastructure, management activities, Evolutions of Systems since 1960s (Mainframes-to-Midrange-to-PCs-to-Client- server computing-to-New age systems) and their management, growth of internet, current business demands and IT systems issues, complexity of today's computing environment, Total cost of complexity issues, Value of Systems management for business.

Unit-II

Factors to consider in designing IT organizations and IT infrastructure, Determining customer's Requirements, Identifying System Components to manage, Exist Processes, Data, applications, Tools and their integration, Patterns for IT systems management, Introduction to the design process for information systems, Models, Information Technology Infrastructure Library (ITIL).

Unit-III

Service-level management, financial management and costing, IT services continuity management, Capacity management, Availability management.

Unit-IV

Configuration Management, Service desk, Incident management, Problem management, Change management, Release management.

Unit-V

Introduction Security, Identity management, Single sign-on, Access Management, Basics of network security, LDAP fundamentals, Intrusion detection, firewall, security information management. Introduction to Storage, Backup & Restore, Archive & Retrieve, Space Management, SAN & NAS, Disaster Recovery, Hierarchical space management, Database & Application protection, Bare machine recovery, Data retention.

Text /Reference Books:

1. **Title:** *IT Infrastructure and Its Management*
   Author: Phalguni Gupta
   Publisher: Tata McGraw-Hill Education
   Year: 2009
   Description: Covers the fundamentals of IT infrastructure including networks, servers, storage, and security from a management perspective.

---

2. **Title:** *Foundations of IT Service Management with ITIL 2011*
   Author: Brady Orand
   Publisher: CreateSpace Independent Publishing
   Year: 2011
   Description: Provides a strong foundation in IT Service Management concepts and ITIL framework, widely used in infrastructure planning and support.

---

3. **Title:** *IT Infrastructure Architecture – Infrastructure Building Blocks and Concepts*
   Author: Sjaak Laan
   Publisher: Lulu Press
   Year: 2017
   Description: A practical guide covering hardware, software, and network components,

Course Outcomes:

1. CO1: Describe the business value and processes of ICT services in an organization and apply that knowledge and skill with initiative to a workplace scenario.
2. CO2: Critically analyze and evaluate the impact of new and current ICT services to an organization.
3. CO3: Understand how effective IT Infrastructure Management requires strategic planning with alignment from both the IT and business perspectives in an organization.
4. CO4: Apply the technical and communications skills that contribute to the operation of ICT services in an organization.
5. CO5: Analyze the theoretical, technical and management issues that deliver ICT services to an organization.

Course Title	Chatbot Development Basics	**Course Code:** CSAB6020	
Teaching Scheme		**Evaluation Scheme**	
Lectures / Tutorial	**3/1 Hrs Week**	**Internal Assessment Test**	**30 Marks**
Total Credits	**4**	**End-Semester Examination**	**70 Marks**

Course Objectives:

1. To introduce students to the fundamentals of chatbot development and conversational AI.
2. To build foundational skills in NLP techniques relevant to chatbot design.
3. To apply machine learning models for understanding and classifying user intents.
4. To design dialogue systems capable of managing multi-turn conversations.
5. To enable students to build and deploy complete chatbot systems using open-source tools.

Detailed Syllabus:

Unit-I

Introduction to Chatbots and Conversational AI: fundamental concepts of chatbots and conversational agents. evolution of chatbot systems, their real-world applications, and the difference between rule-based and AI-driven chatbots. The unit also explores the different types of bots, including FAQ bots, virtual assistants, and social chatbots. Students will learn about the typical architecture of a chatbot, including components such as NLU (Natural Language Understanding), dialogue management, and response generation. An overview of popular chatbot development platforms such as Rasa, Dialogflow, and Botpress is also provided.

Unit-II

Natural Language Processing for Chatbots: NLP techniques required for understanding user input in chatbots. Text preprocessing steps including tokenization, stopword removal, stemming, and lemmatization. Students will also learn about Named Entity Recognition (NER) to identify names, locations, dates, and other key information in a conversation. Intent recognition and slot filling techniques are discussed to map user inputs to specific actions. Tools such as NLTK, spaCy, and Hugging Face Transformers are introduced for implementing these tasks.

Unit-III

Machine Learning for Intent Classification:applying machine learning to classify user intents accurately. Introduction to supervised learning methods relevant to NLP. Feature extraction techniques like Bag of Words and TF-IDF are taught to convert text into numerical vectors. Students will implement intent classifiers using algorithms such as Naive Bayes, SVM, and Logistic Regression. The unit also emphasizes model evaluation using metrics like precision, recall, accuracy, and F1-score. Building ML pipelines for chatbot intent recognition using Scikit-learn is also practiced.

Unit-IV

Dialogue Management and Response Generation:manage multi-turn conversations using rule-based and machine learning-based dialogue systems. The concepts of dialogue flow, slot tracking, and state management are introduced. Students will create response templates and dynamic responses based on user intent and extracted entities. The unit also briefly touches on advanced techniques such as

sequence-to-sequence models and transformer-based architectures for natural language response generation. Integration of response systems with knowledge bases is also discussed.

Unit-V

Building and Deploying a Chatbot: development and deployment of a chatbot. Students will use platforms like Rasa or Dialogflow to define intents, entities, and stories that form the logic of the chatbot. They will test and refine the chatbot's ability to respond appropriately using training data and feedback. The chatbot will then be integrated with real-world interfaces such as Telegram or Facebook Messenger. Finally, deployment options on cloud platforms like Heroku or AWS are introduced, along with tools for testing, logging, and performance evaluation.

Text /Reference Books:

1. **Title:** *Building Chatbots with Python: Using Natural Language Processing and Machine Learning*
   Author: Sumit Raj
   Publisher: Apress
   Year: 2019
   Description: A hands-on guide to building intelligent chatbots using Python, covering NLP with NLTK and ML models.

---

2. **Title:** *Natural Language Processing with Python*
   Authors: Steven Bird, Ewan Klein, Edward Loper
   Publisher: O'Reilly Media
   Year: 2009
   Description: A comprehensive introduction to NLP using Python and the NLTK library. Essential for text processing tasks in chatbot development.

---

3. **Title:** *Conversational AI: Dialogue Systems, Conversational Agents, and Chatbots*
   Authors: Michael McTear
   Publisher: Springer
   Year: 2020
   Description: Covers the full spectrum of conversational AI, from rule-based bots to deep learning-based dialogue systems.

Assignments:

Assignment 1: Rule-Based Chatbot with Python

Aim:To create a basic rule-based chatbot using Python without ML.

Tasks:

- Use simple `if-else` or dictionary-based logic.
- Create a chatbot that answers FAQs (e.g., college admission info, course details).
- Run it in the terminal.

Tools: Python, Jupyter Notebook or Google Colab
Expected Output: Text-based conversation in terminal or notebook.

Assignment 2: Intent Detection Chatbot using Dialogflow (Free Tier)

Aim:To build a chatbot with basic NLP-based intent recognition using Google Dialogflow.

Tasks:

- Create a Dialogflow agent.
- Define 5–6 intents like greeting, asking time, weather, etc.
- Test it in Dialogflow console or integrate with Telegram (optional).

Tools: Google Dialogflow Console (free with Gmail), Web browser
Expected Output: Interactive chatbot responding to defined intents.

Assignment 3: AI Chatbot using Rasa Open Source

Aim:To develop an AI-powered chatbot with custom NLU and dialogue management.

Tasks:

- Install Rasa open source.
- Define NLU data: intents, examples, and entities.
- Create a simple story (flow of conversation).
- Train the model and interact with the chatbot in terminal or browser.

Tools: Rasa (open-source CLI), VS Code, Python
Expected Output: Custom conversational AI chatbot in command line.

Assignment 4: Chatbot with Botpress (Visual No-Code)

Aim: To design a chatbot visually using a no-code/low-code open-source platform.

Tasks:

- Install and launch Botpress locally.
- Design conversation flows using drag-and-drop interface.
- Test chatbot using built-in emulator.

Tools: Botpress (open-source GUI), Web browser
Expected Output: Flow-based chatbot for simple use-cases like order tracking or feedback **collection.**

Assignment 5: NLP Chatbot with Python (NLTK or spaCy)

Aim: To integrate NLP techniques (tokenization, NER, etc.) into chatbot responses.

Tasks:

- Use NLTK or spaCy to preprocess user input.
- Extract named entities (like city names, dates).
- Respond dynamically based on extracted data.

Tools: Python, NLTK/spaCy, Google Colab
Expected Output: Chatbot capable of understanding and reacting to user inputs with NLP.

Course Outcomes:

1. Understand the structure and functioning of different types of chatbot systems.
2. Apply NLP preprocessing and entity recognition techniques to analyze user input.
3. Implement machine learning models for accurate intent classification.
4. Develop dialogue management strategies for natural and coherent interactions.
5. Build, test, and deploy chatbots integrated with real-world applications and platforms.

Elective-III			
Course Title	**Web Designing**	**Course Code:** CSAB6910	
Teaching Scheme		**Evaluation Scheme**	
Lectures / Tutorial	**3 Hrs Week**	**Internal Assessment Test**	**30 Marks**
Total Credits	**3**	**End-Semester Examination**	**70 Marks**

Course Objectives:

1. To provide insight into emerging technologies to design and develop state of - the art web applications using client-side scripting, server-side scripting.
2. To study database connectivity from frontend to back-end.
3. To make students use the cascaded style sheets for formatting the document.
4. To study scripting language.

Detailed Syllabus:

Unit-I

HTML5:

Fundamental Elements of HTML, Formatting Text in HTML, Organizing Text in HTML, Links and URLs in HTML, Tables in HTML, Images on a Web Page, Image Formats, Image Maps, Colors, FORMs in HTML, Interactive Elements, Working with Multimedia - Audio and Video File Formats, HTML elements for inserting Audio / Video on a web page

Unit-II

CSS:

Understanding the Syntax of CSS, CSS Selectors, Inserting CSS in an HTML Document, CSS properties to work with the background of a Page, CSS properties to work with Fonts and Text Styles, CSS properties for positioning an element.

Unit-III

JavaScript:

Using JavaScript in an HTML Document, Programming Fundamentals of JavaScript – Variables, Operators, Control Flow Statements, Popup Boxes, Functions – Defining and Invoking a Function,Defining Function arguments, Defining a Return Statement, Calling Functions with Timer, JavaScript Objects - String, RegExp, Math, Date, Browser Objects - Window, Navigator, History, Location, Document, Cookies, Document Object Model, Form Validation using JavaScript.

Unit-IV

XML:

Comparing XML with HTML, Advantages and Disadvantages of XML, Structure of an XML Document, XML Entity References, DTD.

Unit-V

XSLT: XSLT Elements and Attributes - xsl: template, xsl:apply-templates, xsl:import, xsl:call-template, xsl:include, xsl:element, xsl:attribute, e xsl:attribute-set, xsl:value-of.

Text /Reference Books:

1) HTML 5 Black Book, Covers CSS 3, JavaScript, XML, XHTML, AJAX, PHP and jQuery, 2ed, Dreamtech Press
2) Web Programming and Interactive Technologies, scriptDemics, StarEdu Solutions India.
3) PHP: A Beginners Guide, Vikram Vaswani, TMH

Additional Reference(s):

1) HTML, XHTML, and CSS Bible Fifth Edition, Steven M. Schafer, WILEY
2) Learn to Master HTML 5, scriptDemics, StarEdu Solutions Pvt Ltd.
3) Learning PHP, MySQL, JavaScript, CSS & HTML5, Robin Nixon, O'Reilly
4) PHP, MySQL, JavaScript & HTML5 All-in-one for Dummies, Steve Suehring, Janet Valade Wiley

Course Outcomes:

After completion of the course students will be able to:

CO1: Recall various HTML tags and their functions.
CO2: Describe the relationship between HTML, CSS, and JavaScript in web development.
CO3: Utilize HTML, CSS, and JavaScript to create responsive web pages.
CO4: Design and develop a fully functional database-driven website from scratch.
CO5: Differentiate between HTML, XML and XSLT document.

Course Title	Data Analytics	**Course Code:** CSAB6610	
Teaching Scheme		**Evaluation Scheme**	
Lectures / Tutorial	**3 Hrs Week**	**Internal Assessment Test**	**30 Marks**
Total Credits	**4**	**End-Semester Examination**	**70 Marks**

Course Objectives:

1. To guide and teach the students the concept of data and big data.
2. To apply the data analytics process on given data.
3. To illustrate students how to apply algorithmic strategies while solving problems.
4. To make students study statistical methods and analytics theory for evaluation
5. To teach classification algorithms such as decision trees.

Detailed Syllabus:

Unit-I

Introduction:
Big data overview, state of the practice in Analytics- BI Vs Data Science, Current Analytical Architecture, drivers of Big Data, Emerging Big Data Ecosystem and new approach. Data Analytic Life Cycle: Overview, phase 1- Discovery, Phase 2- Data preparation, Phase 3- Model Planning, Phase 4- Model Building, Phase 5- Communicate Results, Phase 6- Operationalize. Case Study: GINA

Unit-II

Statistical Methods for Evaluation
Hypothesis testing, difference of means, Wilcoxon rank–sum test, type 1 type 2 errors, power and sample size, ANNOVA. Advanced Analytical Theory and Methods: Clustering- Overview, K means- Use cases, Overview of methods, determining number of clusters, diagnostics, reasons to choose and cautions.

Unit-III

Advanced Analytical Theory and Methods:
Association Rules- Overview, a-priori algorithm, evaluation of candidate rules, case study-transactions in grocery store, validation and testing, diagnostics. Regression- linear, logistics, reasons to choose and cautions, additional regression models

Unit-IV

Decision trees
Overview, general algorithm, decision tree algorithm, evaluating a decision tree, Naïve Bayes – Bayes‟ Algorithm, Naïve Bayes‟ Classifier, smoothing, diagnostics. Diagnostics of classifiers, additional classification methods

Unit-V

Analytics for unstructured data
Use cases, Map Reduce, Apache Hadoop. The Hadoop Ecosystem- Pig, HIVE, HBase, Mahout, NoSQL. An Analytics Project-Communicating, operationalizing, creating final deliverables

Text /Reference Books:

1. Maheshwari Anil, Rakshit, Acharya, “Data Analytics”, McGraw Hill, ISBN: 789353160258.
2. Mark Gardner, “Beginning R: The Statistical Programming Language”, Wrox Publication, ISBN: 978-1-118-16430-3
3. Luís Torgo, “Data Mining with R, Learning with Case Studies”, CRC Press, Talay and Francis Group, ISBN9781482234893
4. Carlo Vercellis, “Business Intelligence - Data Mining and Optimization for Decision Making”, Wiley Publications, ISBN: 9780470753866.

Course Outcomes:

1. CO1: Understand the fundamentals of big data and its significance in modern analytics.
2. CO2: Understand and apply statistical methods for evaluation in big data analytics, including hypothesis testing, difference of means, Wilcoxon rank-sum test, and ANOVA.
3. CO3: Explore advanced analytical theory and methods, including association rules mining, using algorithms like the a-priori algorithm.
4. CO4: Evaluate decision trees effectively for classification tasks, considering metrics like accuracy, precision, and recall.
5. CO5: Apply analytics techniques to unstructured data and communicate findings effectively to stakeholders.

Course Title	Project Work- II	Course Code: CSAB6023	
Teaching Scheme		**Evaluation Scheme**	
Practicals/Week	**16 Hrs/Week**	**Internal Assessment Test**	**50 Marks**
Total Credits	**8**	**End-Semester Examination**	**50 Marks**

Course Objectives:

1. To train the learner to independently formulate and solve a social, philosophical, Commercial, or technological problem and present the results in written and oral form.
2. To expose learners to the real-life problems in the World of Work.
3. To provide opportunities to learners to interact with people and understand human relations.

The Bachelor of Computer Applications (BCA) programmed prepares the learners to take up positions as Programmers, Systems Analysts, Systems Designers in the field related to computer application and information technology or learners may go for higher studies in this area.

· The theoretical background of various courses provides you the necessary foundation, principles, and practices to develop effective ways to solve computing problems.

· The hands-on experience gained from the practical courses provide you the knowledge to work with various operating systems, programming languages and software tools.

· This project work is kept in the BCA program to give you an opportunity to develop quality software solutions.

· During the development of the project, you should be involved in all the stages of the software development life cycle (SDLC) like requirements analysis, systems design, software development/coding, testing and documentation, with an overall emphasis on the development of reliable software systems.

· The primary emphasis of the project work is to understand and gain the knowledge of the principles of software engineering practices, and develop a good understanding of SDLC.

· The BCA learners are encouraged to involve themselves completely on the project work in their Fifth semester. It is advised to learners to develop their project for solving problems of the software industry or any research organization.

· Doing this will give more exposure to handling real life problems of project development.

Learners should take this project work very seriously.

· Topics selected, should be complex and large enough to justify as a **BCA** project. The project should be genuine and original in nature and should not be copied from anywhere else.

· Learners, who wish to do their Project Report from any organization, are required to take NOC from the concerned organization, to carry out the Code of the Project done.

· The project covers Study of existing system & System Requirements, Analysis, Design and Coding. Learners should submit their Project Reports as per the guidelines.

· **Project Reports which are not as per the guidelines will not qualify for evaluation.**

Project Scheduling

Learners are supposed to complete their project work within a period of 6 months.

Activity	**Date**
Apply for Project	After completed theory courses of 4st Sem
Submission of Project Proposal	Within one month from apply the project work
Approval of Project work	Within 15 days from submission of project synopsis or proposal.
Start the project	Immediately after approval of the project.
Project Reporting	At the end of 5th month from start
Submission of Project Report	On or before 20 days of the end of examination.
Examination of Project Work	Date will be announced by the University.

About the Project

a. Project carries 10 credit points.
b. At the end of the project, learners should prepare a written document of their project report.
c. Learners can develop their project in the form of a group of 1 to 2 learners.
d. Project work carries marks total 100 marks:

Internal Assessment	50
Project Report	10
Presentation and Demonstration	30
Viva	10

A. Understanding the significance of Project

Most of the learners are under the impression that if a good layout is proposed then it will help them in scoring good marks but the quality of the project is analyzed by proper write-ups. It

provides an opportunity for learners to demonstrate originality and to plan and organize the project work and put the practical approach of all the topics studied in the entire curriculum.

B. Meaning of Project

A project is a study of factual information for comprehending and applying the various concepts of the course into practice. Its main purpose is not to generalize but to study the situation with a practical orientation.

C. Steps in Project Formulation

a. System Study
b. System Analysis
c. System Designing
d. System Development
e. Implementation and Testing

D. Project Supervisor (Guide) Projects can be guided by any:

A person having B.E/B.Tech (Computer Science), MCA, M.Sc. (Computer Science) with minimum 2 years' experience in Industry / Teaching.

E. Project Proposal

A proposal as per the format given should be prepared once the topic is selected. It should not be more than 10 pages and need not be sent separately. The format for the same is: **a.** Title of Project

b. Name of learner or group
c. Objectives
d. Problem Statement
e. Software platform.
f. Hardware platform
g. Methodology and Procedure of Work
h. Detailed information of Guide (Name, Address, qualification and Experience)

F. No Objection Certificate:

If the project is carried out in a company or organization, then a certificate for no objection of the same needs to be presented. It should mention that the organization has no objection in publishing the findings of the project study. The certificate should contain the name of authority with signature and company stamp and should be given on the company's letterhead and duly signed by authorized signatory.

G. Format of the project

a. **Preliminary pages**

1. Cover Page
2. Title Page
3. Completion Certificates
4. Acknowledgement
5. Abstract
6. Index
7. Table of Contents
8. List of Tables
9. List of Figures
10. List of Symbols, Abbreviations and Nomenclature

b. Body of the project

· Containing textual and graphical contents of the project.

- Requirement Analysis
- Project Analysis
- Data Flow Diagram
- Project Design
- ER Diagram
- Table Design
- Reports

c. Bibliography

· Reference List

d. Appendices

e. Paper Publication Details

The table and figures shall be introduced in the appropriate places.

H. Suggested list of topics for Project Report

A sample list of topics for the BCA General Project is provided below. This is just a suggested list and learners are free to choose any other IT project relevant to the BCA curriculum. Learners may choose any programming language such as C, JAVA, PHP, Android etc. There is no compulsion on choosing the project and the software language for the project.

- Employee Record System
- Library Management System
- Learner Record System
- Hotel Reservation System
- eBook Shopping

- Event Management System
- Hostel Management System
- Bus Ticket Reservation System
- Online Food Ordering Systematic Management
- Leave Management System
- Banking System
- Stock Management System
- Mailing Server
- Social Networking
- Online Discussion Forum
- Call Blocker
- Mobile Sensor
- Call Recorder
- Mobile Thesaurus
- Mobile Invoice
- Group Alarm System • Shop Management etc.

I. Technical Specifications of Project Report Length:

1. The length of the report should be between 100-150 pages including the cover page, summary, table of contents, list of figures, list of tables, and acknowledgement.
2. **Script and Page Format:** The report should be typed using a Word Processor on standard A4 (210 mm x 297 mm) paper size. A conventional font, size 12-point and line spacing of

   1.5 mm should be used.
3. **Margins: Left**-hand margins should have a width of not less than 38 mmt o facilitate binding. The right-hand, the top, and the bottom should be 25 mm. Each page must be typed in one side, leaving a wide margin.
4. **Paper and Print Quality: Paper** and print quality are important for successful legibility. The report can be printed on a standard quality paper, (e.g., photocopy paper)
5. **Pagination:** Positioning of page numbers should be on the top right-hand side. Pages starting from the summary until the last list of tables should be numbered using Latin numbers (I, II, III, IV…). Pages starting from the Introduction until the

appendices should be numbered using numbers (1, 2, 3…). Pages with figures and tables or illustrations must be also numbered.

6. **Guard pages/blank pages at the front and back.**
7. **Binding:** The report should be hard bound.
8. **Number of copies:** Three copies should be submitted – self copy, University copy, Supervisor (Faculty member in charge) Copy.

Project Outcome

After the completion of the course, student will be able to

CO1: Discover potential research areas in the field of IT.

CO2: Conduct a survey of several available literatures in the preferred field of study

CO3: Compare and contrast the several existing solutions for research challenges.

CO4: Demonstrate an ability to work in teams and manage the conduct.

CO5: Formulate and propose a plan for creating a solution for the research plan identified.

CO6: Report and present the findings of the study conducted in the preferred domain.