(B.Sc.- CS)
BACHELOR OF SCIENCE – COMPUTER SCIENCE

BACHELOR OF SCIENCE – COMPUTER SCIENCE

Detailed Syllabus

Program Code: B.Sc. (CS)
Duration: 3 Years

EFFECTIVE FROM SESSION: 2022-2023

Department of Computer Sciences
Faculty of Engineering
CHHATRAPATI SHIVAJI MAHARAJ UNIVERSITY
PANVEL, NAVI MUMBAI

SEMESTER I

	BCA 1 Year - Semester 1										
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
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	**15**	**3**	**8**	**150**	**350**	**30**	**70**	**600**	**22**

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

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. The subject aims to provide exposure to problem-solving through programming.
2. It aims to train the student to the basic concepts of the C-programming language.
3. This subject involves a lab component which is designed to give the student hands-on experience with the concepts.
4. Write algorithms, flowcharts and programs.
5. Implement different programming constructs and decomposition of problems into functions
6. Use and implement data structures like arrays and structures to obtain solutions.
7. Define and use pointers with simple applications.

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

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

Computer Fundamental & Office Automation

COURSE CODE: CSAB1020			
Teaching Scheme		**Evaluation Scheme**	
Lectures	**3 Hrs/Week**	**Internal Assessment Test**	**30 Marks**
Total Credits	**4**	**End-Semester Examination**	**70 Marks**

Course Objectives:

The objectives of this course are

1. To provide the foundational concepts of computer hardware, software, operating systems, peripherals, etc. along with how to get the most value and impact from computer technology.
2. To make knowledge for beginners as well as advanced learners who want to deal with computers.
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 understand how to use software packages in day-to-day activities and learn the essential and use of internet.

Detailed Syllabus:

UNIT I

Introduction

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 and Linux

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 petersen - 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

CO1: Remember the basic knowledge about computers.

CO2: Understand & Apply knowledge of flowcharts with algorithm concepts.

CO3: Utilize Windows operating environment with additional knowledge of word processors.

CO4: Understand the basic database applications & create spreadsheets which will be used in daily work life.

CO5: Execute the Linux operating system commands.

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 and 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.

3. Hellriegel, Slocum & Jackson, ' Management - A Competency Based Approach', Thomson South Western, 10th edition, 2007.

4.Harold Koontz, Heinz Weihrich and Mark V Cannice, 'Management - A global

5. Stephen A. Robbins & David A. Decenzo & Mary Coulter, "Fundamentals of Management" 7th Edition, Pearson Education, 2011.

6.Robert Kreitner & Mamata Mohapatra, "Management", Biztantra, 2008.

7. Harold Koontz & Heinz Weihrich "Essentials of management" Tata Mc Graw Hill, 1998. 8.Tripathy PC & Reddy PN, "Principles of Management", Tata McGraw Hill, 1999.

Course Outcomes:

After completion of this subject students will be able to

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.

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.

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.

References:

- Bhanumurthy K V: Ethics and Social Responsibility of Business, Pearson Education India.
- Kaur, Tripat; Values & Ethics in Management, Galgotia Publishers. ● Manuel G Velasquez : Business ethics- concepts and cases Pearson.
- Kaur, Tripat; Values & Ethics in Management, Galgotia Publishers.
- Chakraborty, S.K.; Human values for Managers
- 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.

English Communication

COURSE CODE: ENGG1000			
Teaching Scheme		**Evaluation Scheme**	
Lectures	**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.

Text Books and References:

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.

Mathematics - I

COURSE CODE: MTHG1000			
Teaching Scheme		**Evaluation Scheme**	
Lectures	**3 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 discipline

Detailed Syllabus:

UNIT 1

Matrices

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 Seidal 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.

Recommended 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.

C Programming Lab

COURSE CODE: CSAB1011			
Teaching Scheme		**Evaluation Scheme**	
Lectures	**4 Hrs/Week**	**Internal Assessment Test**	**15 Marks**
Total Credits	**2**	**End-Semester Examination**	**35 Marks**

Suggested list of experiments:

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.

Office Automation Lab

COURSE CODE: CSAB1021			
Teaching Scheme		**Evaluation Scheme**	
Lectures	**4 Hrs/Week**	**Internal Assessment Test**	**15 Marks**
Total Credits	**2**	**End-Semester Examination**	**35 Marks**

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.

Soft Skills

Teaching Scheme		Evaluation Scheme	
Lectures	**3 Hrs/Week**	**Internal Assessment Test**	**—--**
Total Credits		**End-Semester Examination**	**—--**

Course Objectives:

1. To help the students in building interpersonal skills.
2. To develop the skill to communicate clearly.
3. To enhance team building and time management skills.
4. To learn active listening and responding skills.

Detailed Syllabus:

UNIT I

Self-Awareness & self-Development

Self-Awareness: Self-Assessment, Self-Appraisal, SWOT, Goal setting: Personal & career: Self-Assessment, Self-Awareness, Perceptions and Attitudes, Positive Attitude, Values and Belief Systems, Self-Esteem, Self-appraisal, Personal Goal setting. b) Self Development: Career Planning, Personal success factors, Handling failure, Depression and Habit, relating SWOT analysis & goal setting, prioritization.

UNIT II

Communication Skill

Communication: Importance, types, barriers of communication, effective communication. b) Speaking Skills: Public Speaking, Presentation skills, Group discussion: Importance of speaking effectively, speech process, message, audience, speech style, feedback, conversation and oral skills, fluency and self-expression, body language phonetics and spoken English, speaking techniques, word stress, correct stress patterns, voice quality, correct tone, types of tones, positive image projection techniques. c) Listening Skills: Law of nature: you have 2 ears and 1 tongue so listening twice and speaking once is the best policy, Empathic listening, and avoid selective listening. d) Group Discussion: characteristics, subject knowledge, oral and leadership skills, team management, strategies and individual contribution and consistency. e) Presentation skills: planning, preparation, organization, delivery.

UNIT III

Corporate / Business Etiquettes

Corporate / Business Etiquettes: Corporate grooming & dressing, Email & telephone etiquettes, etiquettes in social & office setting: Understand the importance of professional behavior at the work place, Understand and Implement etiquettes in workplace, presenting oneself with finesse and making others comfortable in a business setting. b) Importance of first impression, Grooming, Wardrobe, Body language, Meeting etiquettes (targeted at young professionals who are just entering business environment), Introduction to Ethics in engineering and ethical reasoning, rights and responsibilities.

UNIT IV

Interpersonal relationship

Team work: Team effectiveness, Group discussion, Decision making: Team Communication. Team, Conflict Resolution, Team Goal Setting, Team Motivation Understanding Team Development, Team Problem Solving, Building the team dynamics. Multicultural team activity. b) Group Discussion (GD): Preparation for a GD, Introduction and definitions of a GD, Purpose of a GD, Types of GD, Strategies in a GD, Conflict management, Do's and Don'ts in GD.

Text and Reference Books:

1. Ethics in Engineering Practice and Research by Caroline & Whitbeck, Cambridge University Press.

2. NASSCOM-Global Business Foundation Skills: Accenture, Convergys, Dell et.al. Foundation Books: Cambridge University Press.

3. Basic Managerial Skills by E. H. McGrath, Eastern Economy Edition, Prentice hall India.

4.Personality Development and Group Discussions by Barun K. Mitra, Oxford University Press. 5. Group Discussions and Interview Skills by Priyadarshi Patnaik , Foundation Books , Cambridge University Press.

6. Thinks and Grow Rich by Napoleon Hill, Ebury Publishing, ISBN 9781407029252.

7. Awaken the Giant Within by Tony Robbins HarperCollins Publishers, ISBN139780743409384.

8. Change Your Thoughts; Change Your Life by Wayne Dyer, Hay House India, ISBN139788189988050.

Course Outcomes:

On completion of the course, learner will be able to

CO1: Make use of techniques for self-awareness and self-development.
CO2: Apply the conceptual understanding of communication into everyday practice.
CO3: Understand the importance of teamwork and group discussion skills.
CO4: Develop time management and stress management modules.
CO5: Apply business etiquette skills effectively that an engineer requires.

BSc CS Sem-II

Course Code	Course Title	Hours/ Week			Theory Marks		Practical Marks		Total Marks	Credit
		L	T	P	IA	ESE	IA	ESE		
CSAB2010	Data Structure using C	3	1		30	70			100	4
CSAB2020	Database Management System (DBMS)	3	1		30	70			100	4
EVSG2000	Environmental Studies	3			30	70			100	3
CSAB2030	Software Engineering	3			30	70			100	3
MTHG2010	Element of Statistics	3	1		30	70			100	4
CSAB2011	Data Structure using C Lab			4			15	35	50	2
CSAB2021	Database Management System (DBMS) Lab			4			15	35	50	2
	Aptitude Building									
	TOTAL	**15**	**3**	**8**	**150**	**350**	**30**	**70**	**600**	**22**

Data Structure using C

Course Code: CSAB2010			
Teaching Scheme		**Evaluation Scheme**	
Lectures	**3 Hrs/Week**	**Internal Assessment Test**	**30 Marks**
Total Credits	**4**	**End-Semester Examination**	**70 Marks**

Course learning objectives:

The objectives of this course are

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

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++", PHI Learning Private Limited, Delhi India
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.

CO3: Understand & apply basic data structure operations like searching, insertion, and deletion, traversing mechanism etc. on various problem domains.

CO4: Design and implement advanced data structures, including non-linear ones, and assess their efficiency in solving complex problems.

CO5: Analyze the efficiency of algorithm.

Database Management System

Course Code: CSAB2020			
Teaching Scheme		**Evaluation Scheme**	
Lectures	**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 make the students familiarize with the discipline of database management systems to the students.
2. To give a good formal foundation on the relational model of data and usage of Relational Algebra.
3. To introduce the learners the concepts of basic SQL as a universal Database language
4. To demonstrate the principles behind systematic database design approaches by covering conceptual design, logical design through normalization to the learners.
5. To provide an overview of physical design of a database system by discussing Database indexing techniques

Detailed Syllabus:

UNIT I

Introduction to Databases and Transactions

What is database system, purpose of database system, view of data, relational databases, database architecture, transaction management.

UNIT II

Data Models

The importance of data models, Basic building blocks, Business rules, The evolution of data models, Degrees of data abstraction.

UNIT III

Database Design, ER-Diagram and Unified Modelling Language

Database design and ER Model: Overview, ER-Model, Constraints, ER-Diagrams, ERD Issues, weak entity sets, Codd's rules, Relational Schemas, Introduction to UML

Relational database model: Logical view of data, keys, integrity rules. Relational Database design: features of good relational database design, atomic domain and Normalization (1NF, 2NF, 3NF, BCNF).

UNIT IV

Relational Algebra and Calculus

Relational algebra: introduction, Selection and projection, set operations, renaming, Joins, Division, syntax, semantics. Operators, grouping and ungrouping, relational comparison, Calculus: Tuple relational calculus, Domain relational Calculus, calculus vs. algebra, computational capabilities.

UNIT V

Constraints, Views and SQL

What is a constraint, types of constraints, Integrity constraints.

Views: Introduction to views, data independence, security, updates on views, comparison between tables and views

SQL: data definition, aggregate function, Null Values, nested sub queries, joined relations, Triggers.

Text /Reference Books:

1. Korth, Silbertz, Sudarshan," Database Concepts", McGraw Hill
2. Date C J, " An Introduction to Database Systems", Addision Wesley
3. Elmasri, Navathe, " Fudamentals of Database Systems", Addision Wesley
4. O'Neil, Databases, Elsevier Pub
5. Mathematical methods for Scientists & Engineers, D.A. McQuarrie, 2003, Viva Books

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.

Environmental Studies

Course Code: EVSG2000			
Teaching Scheme		**Evaluation Scheme**	
Lectures	**3 Hrs/Week**	**Internal Assessment Test**	**30 Marks**
Total Credits	**4**	**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
6. Evaluate and interpret various forms of evidence, including text, data, and other media about the environment
7. Work productively with those within and beyond the academy on interdisciplinary collaborative projects
8. Communicate clearly and competently matters of environmental concern and understanding to a variety of audiences in appropriate form.

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. a) Forest resources: Use and over-exploitation, deforestation, case studies. Timber extraction, mining, dams and their effects on forest and tribal people. b) Water resources: Use and over-utilization of surface and ground water, floods, drought, conflicts over water, dams-benefits and problems. c) Mineral resources: Use and exploitation, environmental effects of extracting and using mineral resources, case studies. d) Food resources: World food problems, changes caused by agriculture and overgrazing, effects of modern agriculture, fertilizer-pesticide problems, water logging, salinity, case studies. e) Energy resources: Growing energy needs, renewable and non-renewable energy sources, use of alternate energy sources. Case studies. f) 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. Bio geographical 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.

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 Encyclopedia, 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., Encyclopedia 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.
12. Mckinney, M.L. & School, R.M. 1996. Environmental Science systems & Solutions, Web enhanced edition. 639p.
13. Mhaskar A.K., Matter Hazardous, Techno-Science Publication (TB)

Course Outcomes:

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

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

CO2: Understand and apply the scientific process, as well as appreciate both the potential and limitations of the 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 such as environmental health, food and agriculture, energy, waste and pollution, climate change, population, resource management, and loss of biodiversity.

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 Objective:

The objectives of this course are

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, Program 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 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.

Element of Statistics

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

Course Objectives:

1. Demonstrate knowledge of probability and the standard statistical distributions.
2. Demonstrate knowledge of fixed-sample and large-sample statistical properties of point and interval estimators.
3. Demonstrate knowledge of the properties of parametric, semi-parametric and nonparametric testing procedures.
4. 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

Statistical 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.

Recommended 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.

Data Structure using C Lab

Course Code: CSAB2011			
Teaching Scheme		**Evaluation Scheme**	
Lectures	**4 Hrs/Week**	**Internal Assessment Test**	**15 Marks**
Total Credits	**2**	**End-Semester Examination**	**35 Marks**

Suggested list of experiments:

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, sequential search, binary search, stack, queue, circular queue, linked list, and graph representation.

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.

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**

Suggested list of experiments:

1. Creating a Table with constraints.
2. Writing SQL statements Using ORACLE
    a. Writing basic SQL SELECT statements.
    b. Restricting and sorting data.
    c. Displaying data from multiple tables.
    d. Aggregating data using group functions.
    e. 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.

Course Outcomes:

After completion of this subject students will be able to

CO1: Demonstrate proficiency in writing basic SQL SELECT statements, restricting and sorting data, displaying data from multiple tables, and aggregating data using group functions.

CO2: Analyze database requirements and design efficient database structures using normalization techniques.

CO3: Create SQL queries and Oracle database objects to meet specific business needs and requirements.

CO4: Apply critical thinking to evaluate database designs, SQL statements, and Oracle objects for efficiency, scalability, and maintainability.

CO5: Develop complex database solutions by integrating advanced features such as cursors, procedures, functions, packages, and triggers.

Aptitude Building

Teaching Scheme		Evaluation Scheme	
Lectures	**4 Hrs/Week**	**Internal Assessment Test**	**15 Marks**
Total Credits		**End-Semester Examination**	**35 Marks**

Course Objectives:

The objectives of this course are

Everyone has a desire to improve their aptitude and reasoning skills because nowadays wherever you want to succeed, Aptitude and Reasoning are weapons to seal it, because aptitude is a basic topic which is used to determine any person's problem-solving skills and numerical ability. Anything in life can be achieved by continuous practice because practice makes man intelligent. Key for developing any sort of skill is passionate handwork following a well-planned strategy.

Detailed Syllabus

UNIT I

Basics of Quantitative Abilities Problems

Basics of Quantitative Abilities Problems on Number System Problems on HCF and LCM Problems on Average Problems on Ratio and Proportion Problems on Percentage

UNIT II

Arithmetic Quantitative Abilities Problems

Arithmetic Quantitative Abilities Problems on Ages Problems on Profit and Loss Problems on Simple and Compound Interest Problems on Time and Distance

UNIT III

Logical Reasoning Number Series

Logical Reasoning Number Series Alphanumeric, Letter & Symbol Series Numerical and Alphabet Puzzles Seating Arrangement

UNIT IV

Verbal Reasoning Para

Verbal Reasoning Para – Jumble, Text Completion

Reference Books:

1. Quantitative abilities by Arun Sharma
2. Quantitative Aptitude for Competitive Examinations by R S Agrawal
3. Verbal and Non-Verbal reasoning by R S Agrawal

Course Outcomes:

On successful completion of the course the students will be able to:

CO1: Understand the basic concepts of quantitative ability.

CO2: Understand the basic concepts of logical reasoning Skills.

CO3: Apply satisfactory competency in use of verbal reasoning.

CO4: Solve campus placements aptitude papers covering Quantitative Ability, Logical Reasoning and Verbal Ability.

SEMESTER III

	BSc CS 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
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	**15**	**4**	**8**	**150**	**350**	**30**	**70**	**600**	**23**

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

Computer Organization & Architecture

Course Code: CSAB3010			
Teaching Scheme		**Evaluation Scheme**	
Lectures	**3 Hrs/Week**	**Internal Assessment Test**	**30 Marks**
Total Credits	**4**	**End-Semester Examination**	**70 Marks**

Course Objectives:

The objectives of this course are

1. To make students conceptualize the basics of organizational and architectural issues of a digital computer.
2. To analyze performance issues in processor and memory design of a digital computer.
3. To understand various data transfer techniques in digital computers to learners.
4. 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 multicomputer

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

Reference Book:

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.Anand kumar, “Fundamentals of digital circuits”, 4th edition, PHI.

Course Outcomes:

At the end of this course students will able to

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).

OBJECT ORIENTED PROGRAMMING WITH C++

Course Code: CSAB3020			
Teaching Scheme		**Evaluation Scheme**	
Lectures	**3 Hrs/Week**	**Internal Assessment Test**	**30 Marks**
Total Credits	**4**	**End-Semester Examination**	**70 Marks**

Course Objectives:

The objectives of this course are

1. To make students understand Object Oriented Programming concepts using the C++ language.
2. Introduces the principles of data abstraction, inheritance to the learner.
3. Introduces the principles of virtual functions and polymorphism.
4. To illustrate students how to handle formatted I/O and unformatted I/O.
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 deallocation operators- new and delete, Pre-processor 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.

References:

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).

Operating System

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

Course Objectives:

The objectives of this course are

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

Memory Management:
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, 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

References:

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.

Software Testing and Quality Assurance

Course Code: CSAB3040			
Teaching Scheme		**Evaluation Scheme**	
Lectures	**3 Hrs/Week**	**Internal Assessment Test**	**30 Marks**
Total Credits	**4**	**End-Semester Examination**	**70 Marks**

Course Objective:

The objectives of this course are

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 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

Software 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

Textbooks:

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

Additional References:

1. Software engineering: An Engineering approach, J.F. Peters, W. Pedrycz , John Wiley,2004
2. Software Testing and Quality Assurance Theory and Practice, Kshirsagar Naik, Priyadarshi Tripathy , John Wiley & Sons, Inc. , Publication, 2008 3. Software Engineering and Testing, B. B. Agarwal, S. P. Tayal, M. Gupta, Jones and Bartlett Publishers, 2010

Course Outcomes:

At the end of this course students will able to

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.

Financial Accounting & Management

Course Code: MGTG3100			
Teaching Scheme		**Evaluation Scheme**	
Lectures	**3 Hrs/Week**	**Internal Assessment Test**	**30 Marks**
Total Credits	**4**	**End-Semester Examination**	**70 Marks**

Course Objectives:

The objectives of this course are

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

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 and 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.

Management Information System Course

Course Code: CSEB3040			
Teaching Scheme		**Evaluation Scheme**	
Lectures	**3 Hrs/Week**	**Internal Assessment Test**	**15 Marks**
Total Credits	**3**	**End-Semester Examination**	**35 Marks**

Course Objective:

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.

Course Content

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.

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,

Object Oriented Programming with C++ Lab

Course Code: CSAB3021			
Teaching Scheme		**Evaluation Scheme**	
Lectures	**4 Hrs/Week**	**Internal Assessment Test**	**15 Marks**
Total Credits	**2**	**End-Semester Examination**	**35 Marks**

Suggested 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.

Operating System Lab

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

Suggested 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

Quantitative Aptitude and Logical Reasoning – I
(Internal Evaluation Only)

Teaching Scheme		Evaluation Scheme	
Lectures	**4 Hrs/Week**	**Internal Assessment Test**	**-**
Total Credits		**End-Semester Examination**	**-**

Course Objective:

The objectives of this course are

Logical Reasoning is a part of almost all major Government exams conducted in the country and is one of the most scoring sections as well. Candidates who are preparing for the upcoming competitive exams can find all topics, tips and some sample questions related to Logical reasoning.

Detailed Syllabus
The topics included in the logical reasoning section comprise of the following type of questions:

UNIT I

Verbal Questions – these types of questions can be solved verbally and no pen-paper solution is required.

UNIT II

Image-Based Questions – Mirror images or paper construction-based images may be given in the question and candidates may be asked to find similar or dissimilar figures.

UNIT III

Puzzle Questions – This includes seating arrangement or arranging people/days/months/places, etc. in different formats.

UNIT IV

Sequence Questions – In such questions a series or sequence of people/numbers/alphabets, etc. may be given and questions based on the same may be asked.

Course Outcomes:
At the end of this course students will able to
CO1: Understand the basic concepts of quantitative ability
CO2: Understand the basic concepts of logical reasoning skills
CO3: Acquire satisfactory competency in use of verbal reasoning
CO4: Solve campus placements aptitude papers covering Quantitative Ability, Logical Reasoning and Verbal Ability
CO5: Compete in various competitive exams like CAT, CMAT, GATE, GRE, GATE, UPSC, GPSC etc.

SEMESTER IV

	BCA 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
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	15	4	8	150	350	30	70	600	23

Elective-1: CSSB4620/ CSSB4920
1. Big Data Technology
2. Theory Of Computation

Python Programming

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

Course objectives:

The objectives of this course are

1. Students should be able to understand the concepts of programming before actually starting to write programs.
2. Students should be able to develop logic for Problem Solving.
3. Students should be made familiar with the basic constructs of programming such as data, operations, conditions, loops, functions etc.
4. Students should be able to apply the 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, etc . 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.

References:

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:

At the end of this course students will able to
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).

JAVA Programming

Course Code: CSAB4020			
Teaching Scheme		**Evaluation Scheme**	
Lectures	**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 make students understand the concept of Object-Oriented Programming
2. To help students to understand the use of programming languages such as JAVA to resolve problems.
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 constructor 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.

References:

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:

At the end of this course students will able to

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

Design Analysis & Algorithm

Course Code: CSAB4030			
Teaching Scheme		**Evaluation Scheme**	
Lectures	**3 Hrs/Week**	**Internal Assessment Test**	**30 Marks**
Total Credits	**4**	**End-Semester Examination**	**70 Marks**

Course Objectives:

The objectives of this course are

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 make the learners study different string-matching algorithms.

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)

References:

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:

At the end of this course students will able to

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

Computer Networks

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

Course Objectives:

The objectives of this course are

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.

Detailed Syllabus:

UNIT I

Basic Network Concepts:

Fundamentals of Computer Network- Definition Need of Computer Network, Applications, Component of Computer Network, Network Benefits- Sharing Information (File Sharing, E-mail), Sharing Resources (Printer Sharing, Application Services), Facilitating Centralized Management-Managing Software, Maintaining the Network, Backing up data, Computer Network Classifications-Classification of Network by their Geography. -PAN, CAN, LAN, MAN, WAN, Classification of Network by their Component Role-Peer-to-Peer Network, Server-Based Network, Types of servers

UNIT II

Network Topologies and Networking Devices:

Network Topologies - Introduction, Definition, Selection Criteria, Types of Topologies- Bus, Ring, Star , Mesh, Tree, Hybrid, Network Control / Connecting Devices - Need of Network Control devices, Role of Network Control devices in a Network, Connectors, Hub, Repeater, Bridges, Switches, Router, Gateway, Modem, Network software: NIC Device Driver, client-server software e.g., telnet, ftp

UNIT III

Transmission Media:

Introduction - Need of Transmission Media, Selection Criteria, Types of Transmission Media-

1) Guided Media: Cable Characteristics, Types of Cable-Twisted Pair Cable, Coaxial Cable, Fibre Optic Cable. 2) Unguided media: Types of Communication Band-Microwave Communication, Radio wave

Communication, Satellite Communication, Infrared Communication, Latest Technologies in Wireless Network-Bluetooth Architecture, Wi-Fi, Wi- Max, Cellular (Mobile) Telephone - Band in Cellular Telephony, Calls using Mobile Phones, Transmitting receiving / Handoff operations.

UNIT IV

OSI Reference Model:

Introduction– Layered Architecture, Peer-to- Peer Processes- Interfaces between Layer, Protocols, Organization of the Layers, Encapsulation, Layers of the OSI Reference Model (Functions of eachLayer & Protocols used) – Physical Layer, Data-Link Layer, Network Layer, Transport Layer, Session Layer,

Presentation Layer, Application Layer.

UNIT V

TCP / IP Suite:

Introduction –Addressing mechanism in the Internet, IP Addressing – IP Address classes, classless IP addressing, Subnetting, super netting, Masking, Layered Structure of the TCP / IP Model – Host-to-Network, Internet, Transport, Application, TCP / IP Protocol Suite: Host-to-Network-SLIP and PPP, Internet Layer-ARP, RARP and IP: Introduction, IPv4, IPv6 (Header Format), Difference between IPv4 & IPv6. Transport Layer- TCP and UDP (Frame Format, port addresses), Application Layer- FTP, SMTP, DNS, Comparison between OSI and TCP/IP Network Model.

References:

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:

At the end of this course students will able to

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

Elective – I

Big Data Technology

Course Code: CSSB4620			
Teaching Scheme		Evaluation Scheme	
Lectures	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 make students understand Data Analytics Life Cycle and Business Challenges
2. To study Analytical Techniques and Statically Models
3. To make students to understand Statically Modelling Language

Detailed Syllabus:

UNIT I

Introduction to Big Data
Business Intelligence, Decision Support Systems, Data Warehousing; Definition of Big Data, Big data characteristics & considerations, Introduction to Hadoop.

UNIT II

Big Data Analytics
Big data analytics, Drivers of Big data analytics, Big Data Stack, Typical analytical architecture, Virtualization & Big Data, Virtualization Approaches, Business Intelligence Vs Data science, Applications of Big data analytics.

UNIT III

Data Analytics Lifecycle
Need of Data analytic lifecycle, Key roles for successful analytic projects, various phases of Data analytic lifecycle: Discovery, Data Preparation, Model Planning, Model Building, Communicating Results, Operationalization.

UNIT IV

Machine Learning: Supervised Learning
What is Machine Learning? Applications of Machine Learning; Supervised Learning: Structure of Regression Model, Linear Regression, Logistics Regression, Time series analysis, Support Vector Machine.

UNIT V

Classification & Unsupervised Learning
Classification: Classification Problem, Classification Models, Classification Trees, Bayesian Method; Association Rule: Structure of Association Rule, Apriori Algorithm, General Association; Clustering: Clustering Methods, Partition Methods, Hierarchical Methods.

Reference Books:

1. Business Intelligence – Data Mining and Optimization for Decision Making – Carlo Vercellis – Wiley Publications.
2. Big Data & Analytics – Seema Acharya & Subhashini Chellappan – Wiley Publications
3. Big Data (Black Book) – DT Editorial Services – Dreamtech Press.
4. Data Mining: Concepts and Techniques Second Edition – Jiawei Han and Micheline Kamber – Morgan KaufMan Publisher
5. Beginning R: The Statistical Programming Language – Mark Gardner – Wrox Publication

Course Outcomes:

On completion of the course, student will be able to

CO1: Understand the concepts of Business Intelligence, Decision Support Systems, and Data Warehousing.

CO2: Explore various applications of Big Data analytics.

CO3: Understand the different phases of the Data Analytics Lifecycle

CO4: Explore specific algorithms like Linear Regression, Logistic Regression, Time Series Analysis, and Support Vector Machine within the context of Supervised Learning.

CO5: Apply Association Rule mining and algorithms like Apriori Algorithm.

Elective – I

Theory of Computation

Course Code: CSSB4920			
Teaching Scheme		**Evaluation Scheme**	
Lectures	**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 provide comprehensive insight into theory of computation by understanding grammar, languages and other elements of modern language design to the learners.
2. Students should be able to develop capabilities to design and develop formulations for computing models and identify its applications in diverse areas.

Detailed Syllabus:

UNIT I

Automata Theory:
Defining Automaton, Finite Automaton, Transitions and Its properties, Acceptability by Finite Automaton, Nondeterministic Finite State Machines, DFA and NDFA equivalence, Mealy and Moore Machines, Minimizing Automata.

UNIT II

Formal Languages:
Defining Grammar, Derivations, Language generated by Grammar, Comsky Classification of Grammar and Languages, Recursive Enumerable Sets, Operations on Languages, Languages and Automata

UNIT III

Regular Sets and Regular Grammar:
Regular Grammar, Regular Expressions, Finite automata and Regular Expressions, Pumping Lemma and its Applications, Closure Properties, Regular Sets and Regular Grammar

UNIT IV

Context Free Languages:
Context-free Languages, Derivation Tree, Ambiguity of Grammar, CFG simplification, Normal Forms, Pumping Lemma for CFG Pushdown Automata: Definitions, Acceptance by PDA, PDA and CFG

UNIT V

Linear Bound Automata:
The Linear Bound Automata Model, Linear Bound Automata and Languages.

Turing Machines:
Turing Machine Definition, Representations, Acceptability by Turing Machines

Tutorials:

1. Problems on generating languages for given simple grammar
2. Problems on DFA and NFA equivalence
3. Problems on generating Regular Expressions
4. Problems on drawing transition state diagrams for Regular Expressions
5. Problems on Regular Sets and Regular Grammar
6. Problems on Ambiguity of Grammar
7. Problems on working with PDA
8. Problems on working with Turing Machines
9. Problems on generating derivation trees 10. Problems on Linear Bound Automata/Universal Turing Machine

Reference Books:

1. Theory of Computer Science, K. L. P Mishra, Chandrasekharan, PHI,3rd Edition
2. Introduction to Computer Theory, Daniel Cohen, Wiley,2nd Edition
3. Introductory Theory of Computer Science, E.V. Krishnamurthy,Affiliated East-West Press.
4. Additional References:
5. Theory of Computation, Kavi Mahesh, Wiley India
6. Elements of The Theory of Computation, Lewis, Papadimitriou, PHI
7. Introduction to Languages and the Theory of Computation, John E Martin, McGraw-Hill Education
8. Introduction to Theory of Computation, Michel Sipser, Thomson

Course Outcomes:
On completion of the course, student will be able to

CO1: Describe transitions in Finite Automata and their significance.

CO2: Classify grammars and languages according to Chomsky's hierarchy.

CO3: Apply the Pumping Lemma and understand its applications in proving languages are not regular.

CO4: Understand derivation trees and the ambiguity of grammars.

CO5: Describe various representations of Turing Machines.

Python Programming Lab

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

Suggested list of Practicals:

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 equation.
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 on even position.
31. Python program to print the elements of an array present on 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.

JAVA Programming Lab

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

Suggested list of experiments:

1. Swapping of two numbers without accepting numbers from the user.
2. Find out the largest of two numbers using command line arguments.
3. Print class of students according to given range using switch case.
4. Find GCD and LCM of two positive numbers using a while loop.
5. Display table of given number using for loop.
6. Java Program to check whether input character is vowel or consonant
7. Print the upper triangle using a nested for loop.
   *
   * *
   * * *
   * * * *
8. Java Program to calculate simple interest and compound interest
9. Java program to find occurrence of a character in a String
10. Write a class Circle with methods getdata() and area(). Create an object of that class, find the area and display the result.
11. Calculate absolute value of given number using method overloading.
12. Calculate addition of two numbers using constructor overloading.
13. Write a program to accept n integers from user into an array and display the average of these numbers.
14. Write a program to sort elements of an array in ascending order using bubble sort.
15. Write a program to find and display the sum of diagonal elements of a square matrix.
16. Write a program to implement single inheritance using super keywords.
17. 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.
18. 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.
19. Write a program to implement multiple inheritance.
20. 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.

Verbal Ability and Reasoning (Internal Evaluation Only)

Teaching Scheme		Evaluation Scheme	
Lectures	**2 Hrs/Week**	**Internal Assessment Test**	**—**
Total Credits		**End-Semester Examination**	**—**

Course Objectives:

The objectives of this course are:

1. To acquaint the students with appropriate language skills with the purpose of improving the existing ones
2. To make the learners understand the importance and effective use of verbal communication.
3. To make the learner proficient in Writing.
4. To guide and teach the students to utilize the principles of professional business and technical writing for effective communication in the global world.
5. To deploy technology to communicate effectively in various situations.

Detailed Syllabus:

UNIT I

Verbal Ability- Definition, Demonstrated Skills, Comprehension skills, Spotting Errors, Spellings, Selecting Words, Sentence Formation, Ordering of Words

UNIT II

Paragraph formation, One Word Substitutes, Idioms and Phrases, Change of Voice, Change Of Speech, Verbal Analogues.

UNIT III

Reasoning Ability- Definition, Types of Reasoning: Deductive Reasoning, Inductive Reasoning, Critical Thinking, Intuition.

UNIT IV

The Organs of Speech, The description and Classification of Speech Sounds, Accent and Rhythm in Connected Speech, Varieties of English Pronunciation

UNIT V

Critical thinking, cause and effect, Artificial Language, verbal reasoning and non-verbal reasoning

Course Outcomes:

At the end of the course student will be able to

CO1: Define the concept of verbal ability.
CO2: Apply these advanced verbal skills to interpret and create complex verbal content.
CO3: Understand the different types of reasoning and their applications.
CO4: Understand the anatomy and mechanics of speech production and the nuances of English pronunciation.
CO5: Apply reasoning skills, and solve problems effectively.

SEMESTER V

	B.Sc-CS 3 Year - Semester V										
Type	Course Code	Course Title	Hours / Week			Theory Marks		Practical Marks		Total Marks	Credit
			L	T	P	IA	ESE	IA	ESE		
DC	CSAB5050	Open Source Technologies	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
DE	**	Elective – II	3	-	-	30	70	-	-	100	3
DC	CSAB5051	Open Source Technologies Lab	-	-	4	-	-	15	35	50	2
DC	CSAB5031	Cloud Computing Lab	-	-	4	-	-	15	35	50	2
DC	CSAB5083	Project Work - 1	-	-	8	-	-	15	35	50	4
		TOTAL	**15**	**1**	**16**	**150**	**350**	**45**	**105**	**650**	**24**

Elective-II CSSB5410 / CSSB5970
1. Software Project Management
2. Object Oriented Design Using UML

Open-Source Technologies

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

Course learning objectives:

The objectives of this course are

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.

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:

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

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.

CO2: Develop dynamic web applications that interact with MySQL databases using PHP scripts.

CO3: Analyze SEO strategies and implement them in PHP websites to improve search engine rankings and visibility.

CO4: Understand how Perl language constructs and features enable various programming tasks.

CO5: Develop Perl scripts that leverage advanced techniques to solve complex problems efficiently.

Internet of Things

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

Course learning objectives:
The objectives of this course are:

In this course, students will explore various components of the Internet of things such as Sensors, internetworking and cyber space. In the end they will also be able to design and implement IoT circuits and solutions.

Detailed Syllabus:

UNIT I

IOT

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

IOT Protocols

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

IOT 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 StandardizationforWoT– Platform Middleware for WoT – Unified Multitier WoT Architecture – WoT Portals andBusiness Intelligence.

UNIT V

IOT 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:

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

CO1: Understand concepts of Internet of Things (IoT).

CO2: Discuss the challenges and obstacles faced in standardizing protocols for IoT.

CO3: Identify the design principles behind IoT architecture.

CO4: Analyze the strengths and limitations of WoT compared to traditional IoT architectures.

CO5: Analyze the potential impact of Future Factory Concepts on industrial operations, productivity, and sustainability.

Cloud Computing

Course Code: CSAB5030			
Teaching Scheme		**Evaluation Scheme**	
Lectures	**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 provide students with the fundamentals and essentials of Cloud Computing.
2. To provide students a sound foundation of Cloud Computing so that they are able to start using and adopting Cloud Computing services and tools in their real-life scenarios.
3. To enable students exploring some important cloud computing driven commercial systems and applications.
4. To expose the students to frontier areas of Cloud Computing and information systems, while providing sufficient foundations to enable further study and research.

Detailed syllabus:

UNIT I

Introduction to Cloud Computing – Definition of Cloud – Evolution of Cloud Computing – Underlying Principles of Parallel and Distributed Computing – Cloud Characteristics – Elasticity in Cloud – On-demand Provisioning.

UNIT II

Service Oriented Architecture – REST and Systems of Systems – Web Services – Publish Subscribe Model – Basics of Virtualization – Types of Virtualizations – Implementation Levels of Virtualization
– Virtualization Structures – Tools and Mechanisms – Virtualization of CPU – Memory – I/O Devices –Virtualization Support and Disaster Recovery.

UNIT III

Layered Cloud Architecture Design – NIST Cloud Computing Reference Architecture – Public, Private and Hybrid Clouds – laaS – PaaS – SaaS – Architectural Design Challenges – Cloud Storage – Storage-as-a-Service – Advantages of Cloud Storage – Cloud Storage Providers – S3.

UNIT IV

Inter Cloud Resource Management – Resource Provisioning and Resource Provisioning Methods – Global Exchange of Cloud Resources – Security Overview – Cloud Security Challenges – Software- as-a-Service Security – Security Governance – Virtual Machine Security – IAM – Security Standards.

UNIT V

Hadoop – MapReduce – Virtual Box — Google App Engine – Programming Environment for Google App Engine — Open Stack – Federation in the Cloud – Four Levels of Federation – Federated Services and Applications – Future of Federation.

Text Books:

1. Kai Hwang, Geoffrey C. Fox, Jack G. Dongarra, "Distributed and Cloud Computing, From Parallel Processing to the Internet of Things", Morgan Kaufmann Publishers, 2012.
2. Rittinghouse, John W., and James F. Ransome, ―Cloud Computing: Implementation, Management and Security, CRC Press, 2017.
3. Rajkumar Buyya, Christian Vecchiola, S. ThamaraiSelvi, ―Mastering Cloud Computing, Tata Mcgraw Hill, 2013.
4. Toby Velte, Anthony Velte, Robert Elsenpeter, "Cloud Computing – A Practical Approach, Tata Mcgraw Hill, 2009.
5. George Reese, "Cloud Application Architectures: Building Applications and Infrastructure in the Cloud: Transactional Systems for EC2 and Beyond (Theory in Practice), O'Reilly, 2009.

Course Outcomes:

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

CO1: Explain the essential characteristics of cloud computing and categorize different components of a computing cloud.

CO2: Evaluate the benefits and challenges associated with cloud architecture.

CO3: Implement software as a service (SaaS) and infrastructure as a service (IaaS) technology.

CO4: Critique the advantages and disadvantages of deploying web services within a cloud architecture.

CO5: Analyze business case evaluation criteria and business outcomes examples related to cloud computing.

Network Security

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

Course Objective:

The objectives of this course are

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 Handshake Pitfalls.

Detailed Syllabus:

UNIT I

Introduction:

Attacks, Services and Mechanisms, Security Attacks, Security Services, Integrity check, digital Signature, authentication, has algorithms.

UNIT II

Secret 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

Hash 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, Inter domain, 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.

References:

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

Course Outcome:

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

CO1: Understand the basic concept of network security.

CO2: Illustrate secret key cryptography algorithms.

CO3: Apply hash functions and public key cryptography to the plain text.

CO4: Understand various types of authentication techniques.

CO5: Evaluate security policies and their pitfalls

Elective II

Software Project Management

Course Code: CSSB5410			
Teaching Scheme		**Evaluation Scheme**	
Lectures	**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 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. 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 behavior – 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 BOOK:

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:

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

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.

Elective II

Object oriented design using UML

Course Code: CSSB5970			
Teaching Scheme		**Evaluation Scheme**	
Lectures	**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 get an overview of Object basics, state and properties.
2. To get an overview of Object behavior, methods and messages.
3. To understand various Object-oriented methodologies.
4. To know about Object Oriented system development life cycle.

Detailed Syllabus:

1. Overview of Prominent OO Methodologies:
   a. The Rumbaugh OMT.
   b. The Booch methodology.
   c. Jacobson's OOSE methodologies.
   d. Unified Process.
   e. Introduction to UML.
   f. Important views & diagram to be modelled for system by UML.

2. Factional view(models):
   - Use case diagram
     a. Requirement Capture with Use case.
     b. Building blocks of Use Case diagram - actors, use case guidelines for use case models.
     c. Relationships between use cases - extend, include, generalize.
   - Activity diagram
     a. Elements of Activity Diagram - Action state, Activity state, Object. node, Control and Object flow, Transition (Fork, Merge, Join)
     b. Guidelines for Creating Activity Diagrams.
     c. Activity Diagram - Action Decomposition (Rake). d. Partition - Swim Lane.

3. Static structural view (Models):
   a. Classes, values and attributes, operations and methods, responsibilities for classes, abstract classes, access specification (visibility of attributes and operations).
   b. Relationships among classes: Associations, Dependencies., Inheritance - Generalizations, Aggregation.
   c. Adornments on Association: association names, association classes, qualified association, n- ary associations, ternary and reflexive association.
   d. Dependency relationships among classes, notations
   e. Notes in class diagram, Extension mechanisms, Metadata, Refinements, Derived, data, constraint, stereotypes, Package & interface notation.

f. Object diagram notations and modelling, relations among objects (links).

4. Class Modelling and Design Approaches:
   a. Three approaches for identifying classes - using Noun phrases, Abstraction, Use Case Diagram. II
   b. Comparison of approaches.
   c. Using combination of approaches.
   d. Flexibility guidelines for class diagram: Cohesion, Coupling, Forms of coupling (identity, representational, subclass, inheritance), class Generalization, class specialization versus aggregation.

5. Behavioral (Dynamic structural view):
   - State diagram
     a. State Diagram Notations, events (signal events, change events, Time events).
     b. State Diagram states (composite states, parallel states, History states), transition and condition, state diagram behaviour (activity effect, do-activity, entry and exit activity), completion transition, sending signals.
   - Interaction diagrams:
     a. Sequence diagram - Sequence diagram notations and examples, iterations, conditional messaging, branching, object creation and destruction, time constraints, origin of links, Activations in sequence diagram.
     b. Collaboration diagram - Collaboration diagram notations and examples, iterations, conditional messaging, branching, object creation and destruction, time constraints, origin of links, activations in sequence diagram.

6. Approaches for developing dynamic systems:
   a. Top - down approach for dynamic systems.
   b. Bottom - up approach for dynamic systems.
   c. Flexibility Guidelines for Behavioral Design - guidelines for allocating and designing behaviors that lead to more flexible design.

7. Architectural view:
   a. Logical architecture: dependency, class visibility, subs systems.
   b. Hardware architecture: deployment diagram notations, nodes, object migration between nodes
   c. Process architecture: what are process and threads and their notations in UML, object synchronization, invocation schemes for threads (UML notations for different types of invocations).
   d. Implementation architecture: component diagram notations and examples.

8. Reuse: Libraries, Frame works components and Patterns:
   a. Reuse of classes.
   b. Reuse of components.
   c. Reuse of frameworks, black box framework, white box frame.
   d. Reuse of patterns: Architectural pattern and Design pattern.

Term Work / Assignment: Each candidate will submit an approximately 10-page written report on a case study or mini project. Students have to do OO analysis & design for the project problem, and develop use case model, analysis model and design model for it, using UML. III

Reference books:

1. Designing Flexible Object-Oriented systems with UML - Charles Ritcher Object Oriented Analysis & Design, Sat/.inger. Jackson, Burd Thomson
2. Object oriented Modeling and Design with UML - James Rumbaugh. Micheal Blaha (second edition)
3. The Unified Modeling Language User Guide - Grady Booch, James Rumbaugh, Ivar Jacobson.
4. Object Oriented Modeling and Design - James Rumbaugh
5. Teach Yourself UML in 24 Hours - Joseph Schmuilers
6. Object-Oriented Analysis and Design: using UML Mike O'Docherty Wiley Publication

Practical assignment: Nine assignments, one on each of the diagrams learnt in UML

Course Outcomes:

After completion of the course students are expected to be able to:

CO1: Explain the basics of UML and its importance in software development.

CO2: Analyze building blocks of the use case diagram, including actors and use cases.

CO3: Analyze relationships among classes including associations, dependencies, and inheritance.

CO4: Explain top-down and bottom-up approaches for developing dynamic systems.

CO5: Develop use case, analysis, and design models using UML.

Open-Source Technologies Lab

Course Code: CSAB5051			
Teaching Scheme		**Evaluation Scheme**	
Lectures	**4 Hrs/Week**	**Internal Assessment Test**	**15 Marks**
Total Credits	**2**	**End-Semester Examination**	**35 Marks**

Suggested 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, Open VAS
4. Version Control
5. Working with Drupal
6. Shell Script
7. Android Setup
8. Programming in Android
9. Programming in Android

Course Outcomes:

After completion of this subject students will be able to

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.

CO2: Create shell scripts to automate system administration tasks, improve productivity, and enhance system security.

CO3: Analyze security vulnerabilities and risks using tools like OpenVAS and implement appropriate measures to mitigate them.

CO4: Apply critical thinking to troubleshoot server and network issues, optimize system performance, and enhance security measures.

CO5: Develop basic Android applications using programming languages and tools specific to Android development.

Cloud Computing Lab

Course Code: CSSB5031			
Teaching Scheme		**Evaluation Scheme**	
Lectures	**4 Hrs/Week**	**Internal Assessment Test**	**15 Marks**
Total Credits	**2**	**End-Semester Examination**	**35 Marks**

Suggested List of Experiments

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 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! Pipe

Course Outcomes:

After completion of this subject students will be able to

CO1: Apply techniques to install and configure virtualization software like Oracle VirtualBox and create virtual machines.

CO2: Create and deploy applications on cloud platforms like AWS, Google App Engine, and Windows Azure by designing and implementing appropriate protocols and coordination models.

CO3: Apply knowledge of distributed computing concepts to design and implement applications using Hadoop, AWS, Google App Engine, Windows Azure, and other cloud platforms.

CO4: Design and implement distributed applications using technologies like Hadoop, Zookeeper, and Simple Queue Service (SQS) to address specific problem scenarios and requirements.

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.

Project Work- I

Course Code: CSSB5083			
Teaching Scheme		**Evaluation Scheme**	
Lectures	**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.

Project Outcomes:

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.

SEMESTER VI

B.Sc.-CS 3 Year - Semester VI											
Type	Course Code	**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	CSSB6030	Artificial Intelligence and Machine Learning	3	1		30	70			100	4
DE	**	Elective 3	3	-	0	30	70			100	3
DC	CSSB6031	Artificial Intelligence and Machine Learning Lab			4			15	35	50	2
DC	CSSB6053	Project Work-II	0	0	12			50	50	100	6
		TOTAL	**09**	**2**	**16**	**90**	**210**	**65**	**85**	**450**	**19**

Elective-3: CSAB6610 / CSSB6810
1. Data Analytics
2. Soft Computing

IT Infrastructure Management

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

Course Objectives:

The objectives of this course are

1. To make students understand research, reporting and presentation approaches using the latest ICT tools to examine and critically analyse 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 Books and References:

1. E Turban, E Mclean and James Wetherbe, ―Information Technology for Management
2. Kenneth C Laudon, Jane P Laudon, ―Management Information Systems
3. Roger S Pressman, ―Software Engineering: A Practitioner‘s Approach
4. James A O‘Brien, ―Management Information Systems
5. Walker Royce, ― Software Project Management: A Unified Framework

Course Outcomes:

After completion of the course students are expected to be able to:

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.

CO2: Critically analyze and evaluate the impact of new and current ICT services to an organization.

CO3: Understand how effective IT Infrastructure Management requires strategic planning with alignment from both the IT and business perspectives in an organization.

CO4: Apply the technical and communications skills that contribute to the operation of ICT services in an organization.

CO5: Analyze the theoretical, technical and management issues that deliver ICT services to an organization.

Artificial Intelligence and Machine Learning

Course Code: CSSB6030			
Teaching Scheme		**Evaluation Scheme**	
Lectures	**3 Hrs/Week**	**Internal Assessment Test**	**30 Marks**
Total Credits	**4**	**End-Semester Examination**	**70 Marks**

Course Objectives:

The objectives of this course are

1. Introduce and define the meaning of Intelligence and explore various paradigms for knowledge encoding in computer systems to the students.
2. Introduce subfields of AI such as NLP, Game Playing, Bayesian Models, etc to the learners.

Detailed Syllabus:

UNIT I

Foundations for AI & Convolution Neural Networks

AI: Application areas, AI Basics (Divide and Conquer, Greedy, Branch and Bound, Gradient Descent) NN basics (Perceptron and MLP, FFN, Back propagation), Convolution Neural Networks, Image classification: Text classification, Image classification and hyper-parameter tuning, Emerging NN architectures.

UNIT II

Recurrent Neural Networks & Deep Learning

Building recurrent NN, Long Short-Term Memory, Time Series Forecasting, Auto-encoders and unsupervised learning, stacked auto-encoders and semi-supervised learning, Regularization - Dropout and Batch normalization.

Foundations for ML

ML Techniques overview, Validation Techniques (Cross-Validations), Feature Reduction/Dimensionality reduction, Principal components analysis (Eigen values, Eigen vectors, Orthogonality).

UNIT III

Clustering & Classification

Distance measures, Different clustering methods (Distance, Density, Hierarchical), Iterative distance-based clustering; Dealing with continuous, categorical values in K-Means, constructing a hierarchical cluster, K-Medoids, k-Mode and density-based clustering, Measures of quality of clustering, Naïve Bayes Classifier: Model Assumptions, Probability estimation, Required data processing, M-estimates, Feature selection: Mutual information, Classifier.

UNIT IV

K-Nearest Neighbors & Support Vector Machines

K-Nearest Neighbors: Computational geometry; Voronoi Diagrams; Delaunay Triangulations, K-Nearest Neighbor algorithm; Wilson editing and triangulations, Aspects to consider while designing K-Nearest Neighbor, Support Vector Machines: Linear learning machines and Kernel space, Making Kernels and working in feature space, SVM for classification and regression problems.

UNIT V

Association Rule mining

Association Rule mining: The applications of Association Rule Mining: Market Basket, Recommendation Engines, etc. A mathematical model for association analysis; Large item sets; Association Rules, Apriori: Constructs large item sets with mini sup by iterations; Interestingness of discovered association rules; Application examples; Association analysis vs. classification, FP-trees.

Text /Reference Books:

1. Artificial Intelligence: A Modern Approach, Stuart Russel, Peter Norvig
2. Artificial Intelligence, 2nd Edition, Rich and Knight

Course Outcomes:

At the end of this course students will demonstrate the ability to:

CO1: Explain AI basics such as Divide and Conquer, Greedy algorithms, Branch and Bound, and Gradient Descent.

CO2: Explain Long Short-Term Memory (LSTM) networks for time series forecasting.

CO3: Apply AI and machine learning techniques to solve practical problems.

CO4: Describe different clustering methods including Distance-based, Density-based, and Hierarchical clustering.

CO5: Analyze the performance of different machine learning models and algorithms.

Elective III

Data Analytics

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

Course Objectives:

The objectives of this course are

1. To guide and teach the students to develop problem solving abilities using Mathematics
2. To illustrate students how to apply algorithmic strategies while solving problems
3. To develop time and space efficient algorithms.
4. To make students study algorithmic examples in distributed, concurrent and parallel environments

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

References:

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:

On completion of the course, student will be able to

CO1: Understand the fundamentals of big data and its significance in modern analytics.

CO2: Understand and apply statistical methods for evaluation in big data analytics, including hypothesis testing, difference of means, Wilcoxon rank-sum test, and ANOVA.

CO3: Explore advanced analytical theory and methods, including association rules mining, using algorithms like the a-priori algorithm.

CO4: Evaluate decision trees effectively for classification tasks, considering metrics like accuracy, precision, and recall.

CO5: Apply analytics techniques to unstructured data and communicate findings effectively to stakeholders.

Elective III

Soft Computing

Course Code: CSSB6810			
Teaching Scheme		**Evaluation Scheme**	
Lectures	**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. Understanding differential behavior of Human and Intelligence Systems.
2. Understanding the nature of problems solved with Soft Computing.
3. Understanding components of Soft Computing.

Detailed Syllabus:

UNIT I

Introduction to Intelligent System and Soft Computing

Characteristic behavior of Intelligent systems, Knowledge based systems, Knowledge Representation and Processing, Soft Computing characteristics, Constitutes of Soft Computing-Fuzzy Logic and Computing, Neural Computing, Evolutionary Computing, Rough Sets, Probabilistic Reasoning and Machine Learning.

UNIT II

Neuro Computing- Supervised Learning

Biological background, Pattern recognition tasks, Features of artificial neural networks, Activation functions, Perceptron model, Perceptron for classification and its limitations, Architectures of multilayer feed-forward neural networks, Back-propagation learning algorithm, Limitations of MLP.

Neuro Computing- Unsupervised Learning

Hebb's learning rule for competitive learning, Kohonen's self-organizing map and network topology, applications of SOM, Hopfield network and its topology, Boltzmann Machines, Adaptive Resonance Theory.

UNIT III

Fuzzy Logic and Fuzzy Systems

Evolution of fuzzy logic, fuzzy sets, fuzzy logic operations, fuzzy relations, Fuzzy arithmetic and fuzzy measures. Fuzzy rules and reasoning, Fuzzy inference systems, Fuzzy modeling and decision making, Neuro-fuzzy modeling.

UNIT IV

Evolutionary Computing

Biological background and Overview of evolutionary computing, Genetic algorithm and search space, Operators in genetic algorithm- encoding, selection, crossover, and mutation, Classification of GA, Evolutionary Programming and Strategies.

UNIT V

Applications of Soft Computing Techniques

Applications of fuzzy in pattern recognition-character recognition, Applications of evolutionary computing in Image processing and computer vision, Soft computing in mobile ad-hoc networks, Soft computing in Information Retrieval and Semantic web, Soft Computing in Software Engineering.

Reference Books:

1. Fakhreddine O. Karray, Clarence De Silva, 'Soft Computing and Intelligent systems design' Pearson Education, ISBN 978-81-317-2324-1.

2. B. K. Tripathy, J. Anuradha, 'Soft Computing: advances and applications', Cengage learning, ISBN- 13: 978-81-315-2619-4. 1. S. N. Sivanandam, S. N. Deepa, Principles of Soft Computing, Wiley publications, 2nd Edition.

3. J. S. R. Jang, C. T. Sun, E. Mizutani, 'Neuro-Fuzzy and Soft Computing- A computational approach to Learning and Machine Intelligence' PHI,

4. David E. Goldberg , Genetic Algorithms - Pearson Education, 2006 4. Satish Kumar, "Neural Networks - A Classroom Approach", Tata McGraw,Hill

Course Outcomes:

After completion of the course students will be able to:

CO1: Explain Knowledge Representation and Processing in Knowledge-based Systems.

CO2: Analyze the limitations of Multilayer Feed-forward Neural Networks (MLP).

CO3: Apply fuzzy modeling and decision-making techniques.

CO4: Describe Genetic Algorithms (GA) and their operators.

CO5: Analyze applications of evolutionary computing in image processing, computer vision, and other fields.

AI & ML Lab

Course Code: CSSB6031			
Teaching Scheme		**Evaluation Scheme**	
Lectures	**4 Hrs/Week**	**Internal Assessment Test**	**15 Marks**
Total Credits	**2**	**End-Semester Examination**	**35 Marks**

Suggested Program List:

1. Implement A* Search algorithm
2. Implement AO* Search algorithm.
3. Implement Candidate-Elimination algorithm
4. Decision tree based ID3 algorithm
5. Implement Back propagation algorithm
6. Implement the naive Bayesian classifier.
7. EM algorithm to cluster a set of data stored in a .CSV file.
8. Implement k-Nearest neighbor algorithm.
9. Implement the non-parametric Locally Weighted Regression algorithm

Course Outcomes:

After completion of this subject students will be able to

CO1: Develop programs using appropriate data structures and algorithms to analyze and process data, make predictions, classify data points, and cluster data sets.
CO2: Apply critical thinking to evaluate algorithm performance, optimize code for efficiency, and handle different data types and formats effectively.
CO3: Analyze problem requirements and design algorithmic solutions to implement machine learning and data mining algorithms effectively.
CO4: Create and customize algorithms to address specific problem scenarios, datasets, or application domains.
CO5: Apply advanced techniques such as optimization, parallelization, and feature engineering to enhance algorithm performance and accuracy.

Project Work -II

Course Code: CSSB6053			
Teaching Scheme		**Evaluation Scheme**	
Lectures	**12Hrs/Week**	**Internal Assessment Test**	**50 Marks**
Total Credits	**6**	**End-Semester Examination**	**50 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 participate 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 (50% 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.

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.