

THANTHAI PERIYAR GOVERNMENT ARTS AND SCIENCE COLLEGE (AUTONOMOUS), TIRUCHIRAPPALLI-23
DEPARTMENT OF COMPUTER APPLICATIONS

BCA PROGRAMME

(Syllabus applicable to the candidates admitted from the academic year 2023 -2024 onwards)

S. No.	Part	Course		Exam Hours	Hours	Credit	Int.	Ext.	Total
SEMESTER – I									
1	I	Language – I	Tamil – I	3	6	3	25	75	100
2	II	Language – II	English – I	3	6	3	25	75	100
3	III	Core – I	Programming in C	3	6	5	25	75	100
		Core – II*-P	C Programming & HTML Lab	-	2	-	-	-	-
4		First Allied - I	Generic Elective - I Either (or)	3	4	4	25	75	100
		First Allied – II*	Generic Elective - II Either (or)	-	2	-	-	-	-
5	IV	Skill Based Elective - I	PCSEC: HTML Fundamentals	3	2	2	25	75	100
6		VE	Value Education	3	2	2	25	75	100
			Total		30	19	150	450	600
SEMESTER – II									
7	I	Language – I	Tamil – II	3	6	3	25	75	100
8	II	Language – II	English – II	3	4	3	25	75	100
9	III	Core – II-P	C Programming & HTML Lab	3	4	4	40	60	100
10		Core-III	Data Structures and Algorithms	3	5	5	25	75	100
11		First Allied –II	Generic Elective - II Either (or)	3	3	3	25	75	100
12		First Allied – III	Generic Elective -III Either (or)	3	4	4	25	75	100
13	IV	ES	Environmental Science	3	2	2	25	75	100
14		NMSDC*	NMSDC-I :		2	2	25	75	100
			Total		30	26	215	585	800
SEMESTER – III									
15	I	Language – I	Tamil – III	3	6	3	25	75	100
16	II	Language – II	English – III	3	6	3	25	75	100
17	III	Core – IV	Database Management Systems	3	4	4	25	75	100
		Core – V*-P	SQL & Java Programming Lab	-	2	-	-	-	-
18		Core – VI : ME	Computer Organization and Architecture / Artificial Intelligence	3	4	4	25	75	100
19		Second Allied – I	Generic Elective-IV Either (or)	3	4	4	25	75	100
		Second Allied – II*	Discipline Specific Elective - I: Either (or)	-	2	-	-	-	-
20	IV	Non Major Elective – I	NME I: HTML Basics / E-Commerce	3	2	2	25	75	100
			Total		30	20	150	450	600

S. No.	Part	Course	Exam Hours	Hours	Credit	Int.	Ext.	Total	
SEMESTER – IV									
21	I	Language – I	Tamil – IV	3	6	3	25	75	100
22	II	Language – II	English – IV	3	6	3	25	75	100
23	III	Core – V-P	SQL & Java Programming Lab	3	4	4	40	60	100
24		Core – VII	Programming in Java	3	5	5	25	75	100
25		Second Allied – II- P	Discipline Specific Elective - I: Either (or)	3	3	3	25	75	100
26		Second Allied – III	Discipline Specific Elective - II: Either (or)	3	4	3	25	75	100
27	IV	NMSDC* II	NMSDC-II :		2	2	25	75	100
Total				30	23	190	510	700	
SEMESTER – V									
28	III	Core – VIII	Operating Systems	3	5	5	25	75	100
29		Core – IX	JavaScript Programming	3	5	5	25	75	100
30		Core – X	Computer Networks	3	6	5	25	75	100
31		Core – XI – P	Scripting Languages Lab	3	5	5	40	60	100
32		Major Elective – II	DSE III: PHP Programming / Introduction to Data Science	3	5	3	25	75	100
33	IV	Non Major Elective – II	NME II: Computer Fundamentals / Multimedia Systems	3	2	2	25	75	100
34		SSD	Soft Skill Development	3	2	2	25	75	100
35	V	EA	Extension Activities	-	-	1	25	75	100
Total				30	28	215	585	800	
SEMESTER – VI									
36	III	Core – XII	Dot Net Programming	3	6	6	25	75	100
37		Core – XIII	Programming in Python	3	6	5	25	75	100
38		Core – XIV	Software Testing	3	5	5	25	75	100
39		Core – XV-P	Dot Net & Python Programming Lab	3	6	5	40	60	100
40		Major Elective - III	DSE-IV: Image Editing Tool / Data Mining and Warehousing	3	5	3	25	75	100
41		NMSDC*	NMSDC-III :		2	2	25	75	100
Total				30	26	165	435	600	
Grand Total				180	142	1085	3015	4200	
*Optional			Extra/Additional : Computer Hardware and Troubleshooting						

Course	BCA	Part III	Core I	Semester	I
Major	CA	Programming in C		Credit	5
				Hours	6

Course Objectives

1. Understand the fundamental components of a C program, such as its structure, data types and variables, including how to declare and initialize them, assign values to them and declare them as constants.
2. To understand the different types of operators in C and their uses, as well as how to evaluate arithmetic expressions and perform input and output operations in a formatted manner.
3. To understand and use decision-making statements and looping statements in C programming, including if-else, switch, ternary operator, while, do-while and for loops.
4. To understand the concept and implementation of arrays and strings and to learn how to define and use user-defined functions in C programming, including recursion
5. To understand and apply the concepts of structures and pointers in C programming and to learn how to work with files in C, including defining, opening and closing files.

UNIT I

Structure of C Program – Character Set – C Tokens – Keywords and Identifiers – Constants – Variables – Data Types – Declaration of Variables – Assigning Values to Variables – Defining Symbolic Constants – Declaring a Variable as Constant.

UNIT II

Operators: Arithmetic – Relational – Logical – Assignment – Increment and Decrement – Conditional – Bitwise – Special. Arithmetic Expressions – Evaluation of Expressions – Precedence of Arithmetic Operators. Input and Output Operations: Reading a Character – Writing a Character – Formatted Input and Formatted Output.

UNIT III

Decision Making and Branching: Simple If Statement – The If...Else Statement – The Else If Ladder – The Switch Statement – The Ternary Operator – The Goto Statement. Decision Making and Looping: While Statement – Do Statement – For Statement.

UNIT IV

Arrays: One-Dimensional Arrays – Declaration and Initialization – Two Dimensional Arrays. Strings: Declaration and Initialization – Reading Strings from Terminal – Writing Strings to Screen. User Defined Functions: Defining and Declaring Functions – Return Values and their Types – Function Calls – Recursion.

UNIT V

Structures: Defining a Structure – Declaration and Initialization – Accessing Structure Members – Pointers: Understanding Pointers – Accessing the Address of a Variable – Declaring and Initializing Pointer Variables – Files: Defining and Opening a File – Closing a File.

Course Outcomes

Upon completion of this course, the students will be able to:

1. Understand the fundamental structure of a C program and the various data types, keywords, identifiers and constants used in C programming.
2. Develop a solid understanding of arithmetic, relational, logical and other operators in C, as well as input/output operations and the evaluation of arithmetic expressions.
3. Gain proficiency in implementing decision-making and looping constructs, including if statements, switch statements, while loops, do-while loops and for loops.

4. Learn how to work with arrays, strings and user-defined functions in C programming, including declaring and initializing them, as well as performing operations on them.
5. Develop proficiency in working with structures, pointers and files in C programming, including defining and initializing structures, accessing structure members and reading and writing to files.

Text Book

1. E. Balagurusamy, Programming in C, Tata McGraw Hill Education Pvt. Ltd. (2018).

Reference Book

1. Let us C, Yashavant Kanetkar, Thirteenth Edition, BPB Publications, New Delhi (2014).

Course	BCA	Part III	Core II - Practical	Semester	I & II
Major	CA	C Programming & HTML Lab		Credit	4
				Hours	2 + 4

Course Objectives

1. Develop an understanding of basic programming concepts (conditional statements, loops and functions) and gain proficiency in using control structures, functions, pointers, arrays and structures in C programming language.
2. Learn to read and write data from files and perform basic file operations.
3. Learn the fundamental concepts of HTML markup language and develop proficiency in creating and formatting text, images and links using HTML tags.
4. Develop the ability to structure content using headings, paragraphs, lists and tables.
5. Build proficiency in creating web pages (with appropriate colour and design) and learn how to incorporate interactivity on web pages using hyperlinks and clickable elements.

C Programming

1. Write a program to find whether the given number is odd or even (Using if-else).
2. Write a program to find the roots of a quadratic equation (Using if-else).
3. Write a program to find the biggest of three given numbers (Using if-else).
4. Write a program to find the sum of the digits of a given number (Using while).
5. Write a program to print multiplication table (Using do-while).
6. Write a program to implement simple calculator (Using switch).
7. Write a program to find the size of data types (Using sizeof()).
8. Write a program to sort the given set of numbers using single dimensional array.
9. Write a program to perform matrix addition and subtraction.
10. Write a program to convert the given string to upper case.
11. Write a program to illustrate a function with arguments with return values.
12. Write a program to find the factorial of a given number using recursion.
13. Write a program to swap two numbers using pointers.
14. Write a program to assign values to structure variable and retrieve the values.
15. Write a program to print employee details using fscanf() and fprintf() (Using files).

HTML

1. Design a HTML document describing you. Assign a suitable background design, background color and a text color.
2. Design a colourful web page of your home town with images and pictures.
3. Develop a HTML document which displays your country name as <h1> heading and displays its states name as hot text. When you click the states name, it opens another page, which tells about that state.
4. Design a web page of your favourite teacher, explaining his/her academic and personal facets and give suitable headings and horizontal rules. Design it in appropriate color.
5. Develop a HTML program to print the courses offered at your college using ordered and unordered lists.
6. Design a telephone bill using table tag and various font styles.

Course Outcomes

Upon completion of this course, the students will be able to:

1. Ability to identify and implement conditional statements and loops in C programming language to solve problems.
2. Proficiency in using C programming language to find the roots of quadratic equations, perform basic calculations and sort data.

3. Understanding of HTML markup language to create web pages with appropriate structure, style and formatting.
4. Skill in designing web pages with elements such as headings, lists, tables and images and ability to create interactive pages with hyperlinks.
5. Knowledge of how to use HTML to create forms, manipulate text and apply styling, as well as ability to work with files and perform various operations on them.

Reference Books

1. E. Balagurusamy, Programming in C, Tata McGraw Hill Education Pvt. Ltd. (2018).
2. Let us C, Yashavant Kanetkar, Thirteenth Edition, BPB Publications, New Delhi (2014).
3. C. Xavier, World Wide Web Design with HTML, McGraw-Hill Education, 2019.
4. N.P. Gopalan, J. Akilandeswari, Web Technology – A Developer’s Perspective, Prentice Hall of India Private Ltd, New Delhi, 2007.

Course	BCA	Part IV	Skill-Based Elective – I	Semester	I
Major	CA	HTML Fundamentals		Credit	2
				Hours	2

Course Objectives

1. To introduce the basics of HTML including the structure of HTML documents, anchor tags, hyperlinks, headers, titles and other elements necessary for creating web pages.
2. Designing the Body Section in HTML including headings, horizontal rules, paragraphs, tab settings, images and borders for images.
3. To teach students how to create ordered and unordered lists in HTML, including using attributes and nesting lists.
4. To teach students how to create tables in HTML, specify the width of tables and cells and customize cells using colour and spanning techniques.
5. To teach students how to use HTML to create framesets with nested frames and to understand the different units of measurement that can be used to define frame size.

UNIT I

Introduction to HTML: HTML Documents – Anchor Tag – Hyper Links – Header Section – Title – Prologue – Links – Colourful Web Page – Background Design – Background Colour – Text Colour – Comment Lines.

UNIT II

Designing the Body Section: Heading Printing – Aligning the Headings – Horizontal Rule – Paragraph – Tab Settings – Images and Pictures – Low Resolution Images – Adding Space around the Image – Border for an Image.

UNIT III

Ordered and Unordered Lists: Lists – Unordered Lists – Bullets in a List – Plain Attribute – Headings in a List – Ordered Lists – Type Attribute – Nested Lists.

UNIT IV

Table Handling: Tables – Tables Creation in HTML – Width of the Table and Cells – Cells Spanning Multiple Rows / Columns – Colouring Cells – Column Specification.

UNIT V

Frames: The Frameset Definition – Pixel Unit – Percentage Unit – Fraction Unit – Frame Definition – Nested Framesets.

Course Outcomes

Upon completion of this course, the students will be able to:

1. Design and create HTML documents using the basic elements, including hyperlinks, headers, titles, prologues and comments.
2. Demonstrate proficiency in designing the body section of an HTML document using various formatting techniques such as headings, paragraphs, images and borders.
3. Construct and format ordered and unordered lists with nested lists using appropriate attributes.
4. Create tables in HTML, including adjusting the width and colour of cells and spanning multiple rows or columns.
5. Apply frameset definitions to create nested frames using various units of measurement such as pixels, percentages and fractions.

Text Book

1. C. Xavier, World Wide Web Design with HTML, McGraw-Hill Education, 2019.

Reference Book

1. N.P. Gopalan, J. Akilandeswari, Web Technology – A Developer’s Perspective, Prentice Hall of India Private Ltd, New Delhi, 2007.

Course	BCA	Part III	Core III	Semester	II
Major	CA	Data Structures and Algorithms		Credit	5
				Hours	5

Course Objectives

1. To understand the basic terminology, operations and representation of linear arrays in memory and perform operations like traversal, insertion and deletion on them.
2. To understand the concept and implementation of linked lists and perform basic operations like insertion, deletion and searching.
3. Design and implement algorithms for implementing stacks, queues, dequeues and priority queues using both array and linked representations.
4. To teach students the concept of trees, specifically binary trees and binary search trees and how to represent, traverse, search, insert and delete nodes in these data structures.
5. To enable students to understand the basics of graph theory and to apply various graph traversal techniques and sorting algorithms.

UNIT I

Introduction – Basic Terminology – Data Structures – Data Structure Operations – Linear Arrays – Representation of Linear Arrays in Memory – Traversing Linear Arrays – Inserting into a Linear Array – Deleting from a Linear Array.

UNIT II

Linked Lists – Representation of Linked Lists in Memory – Traversing a Linked List – Searching a Linked List – Garbage Collection – Insertion into a Linked List – Deletion from a Linked List – Header Linked Lists.

UNIT III

Stacks – Array Representation of Stacks – Linked Representation of Stacks – Queues – Linked Representation of Queues – Deques – Priority Queue – One-Way List Representation of Priority Queue – Array Representation of Priority Queue.

UNIT IV

Trees: Binary Trees – Representing Binary Trees in Memory – Traversing Binary Trees – Binary Search Trees – Searching and Inserting in Binary Search Trees – Deleting in a Binary Search Tree.

UNIT V

Graphs: Sequential Representation of a Graph – Adjacency Matrix – Path Matrix – Linked Representation of a Graph – Traversing a Graph – Sorting: Insertion Sort – Selection Sort – Merge Sort – Radix Sort.

Course Outcomes

Upon completion of this course, the students will be able to:

1. Understand and apply basic terminology related to data structures.
2. Analyze and perform operations on linear arrays such as insertion and deletion.
3. Demonstrate an understanding of linked lists and their representation in memory, as well as perform operations on them such as insertion and deletion.
4. Explain the concepts of stacks, queues and priority queues and their representations in memory and apply relevant operations on them.
5. Understand graphs and their sequential and linked representations. Learn how to traverse a graph and understand path matrix. Learn different sorting algorithms including insertion, selection, merge and radix sort.

Text Book

1. Seymour Lipschutz, Data Structures, McGraw Education Pvt. Ltd., Revised First Edition, 2018.

UNIT I : Chapter 1 (1.1 to 1.4) and Chapter 4 (4.2 to 4.5).

UNIT II : Chapter 5 (5.2 to 5.9).

UNIT III : Chapter 6 (6.2 to 6.4 and 6.10 to 6.13).

UNIT IV : Chapter 7 (7.2 to 7.4 and 7.7 to 7.9).

UNIT V : Chapter 8 (8.3, 8.5 and 8.7) and Chapter 9 (9.3, 9.4, 9.6 and 9.7).

Reference Books

1. Seymour Lipschutz, Data Structures with C, McGraw Education Pvt. Ltd., Special Indian Edition, 2017.

2. A. Chitra and P.T. Rajan, Data Structures, Tata McGraw – Hill Publishing Company Limited, New Delhi.

Course	BCA	Part III	Core IV	Semester	III
Major	CA	Database Management Systems		Credit	4
				Hours	4

Course Objectives

1. To provide an understanding of the basic concepts of database systems, including data independence, database architecture, database management systems and client/server architecture.
2. To understand and apply the entity-relationship model and its diagramming technique to design database systems and comprehend the basics of various data models, including relational, hierarchical and network models.
3. Develop an understanding of normalization and the various normal forms in database design.
4. To familiarize students with SQL commands for creating and manipulating database tables, including inserting, viewing, sorting, deleting, updating, renaming, truncating and dropping tables.
5. To understand advanced SQL topics including pattern matching, aggregate and numeric functions, group by clause, set operations, views creation, selection and update.

UNIT I

Database Systems – Database – Data Independence – The Three Levels of the Architecture – Database Administrator – Database Management System – Client / Server Architecture.

UNIT II

Entity-Relationship Model – Entity-Relationship Diagram – Entities and Entity Sets – Relationships and Relationship Sets – Mapping Cardinalities – Mapping Constraints – Keys. Data Models: Relational Model – Hierarchical Model – Network Model.

UNIT III

Normalization – Definition of Functional Dependence – Normal Forms – First Normal Form – Second Normal Form – Third Normal Form – Boyce-Codd Normal Form (BCNF) – Fourth Normal Form – Fifth Normal Form.

UNIT IV

The Create Table Command – Data Insertion – Viewing Data – Distinct Keyword – Sorting Data in a Table – Delete Operations – Updating the Content of a Table – Alter Table – Rename Table – Truncate Table – Drop Table.

UNIT V

Pattern Matching – Dual Table – Aggregate Functions – Numeric Functions – String Functions – Group by Clause – Set Operations: Union – Intersect – Minus – Views: Creating View – Selecting a Dataset from View – Updatable Views – Drop a View.

Course Outcomes

Upon completion of this course, the students will be able to:

1. Understand the fundamentals of database systems, including data independence, architecture levels and client/server architecture.
2. Analyze and design entity-relationship models using the entity-relationship diagram, mapping cardinalities and mapping constraints.
3. Apply normalization techniques to achieve database efficiency and avoid data anomalies.
4. Perform operations on tables, including data insertion, deletion, updating and table manipulation using SQL commands.

5. Apply SQL functions, including pattern matching, aggregate functions, numeric functions and set operations to manage and retrieve data and create and manage views.

Text Books

1. C.J. Date, A. Kannan & S. Swamynathan, An Introduction to Database Systems, Pearson Eighth Edition.
UNIT I : Chapter 1 (1.2, 1.3 & 1.5), Chapter 2 (2.2, 2.7 to 2.9).
2. Rajesh Narang, Database Management Systems. PHI, Second Edition (2012).
UNIT II : Chapter 2 & 3.
UNIT III : Chapter 7.
3. Ivan Bayross, SQL, PL/SQL The Programming Language of Oracle, BPB Publications, Third Revised Edition.
UNIT IV : Chapter 7.
UNIT V : Chapter 9, 10 & 11.

Reference Book

1. Panneerselvam R, Database Management Systems, PHI, Second Edition, 2015.

Course	BCA	Part III	Core V - Practical	Semester	III & IV
Major	CA	SQL & Java Programming Lab		Credit	4
				Hours	2 + 4

Course Objectives

1. Learn and apply Data Definition Language (DDL) commands such as table creation with primary and foreign keys, table alteration and table dropping.
2. Gain proficiency in Data Manipulation Language (DML) operations such as data insertion, updating, deletion, renaming and pattern matching.
3. Develop a strong understanding of the various built-in and aggregate functions and learn about the set operations and views in SQL to manage and organize data effectively.
4. Acquire the skills to implement method overloading, file handling, threading, exception handling, inheritance and packages in Java for advanced software development.
5. Demonstrate knowledge of string handling functions in Java and their practical applications and gain practical experience in creating applets in Java for displaying text and geometrical objects.

SQL

Data Definition Language

Table Creation using primary key and foreign key

Table Alteration

Drop Table

Data Manipulation Language

Data Insertion

Data Updating

Data Deletion

Rename Operation

Pattern Matching

Built-in Functions

Aggregate Functions

Set Operations

Views

Java

1. Write a program to reverse a given number using while loop.
2. Write a program to find the biggest and smallest number using for loop.
3. Write a program to find area of rectangle and square using method overloading.
4. Write a program to demonstrate the various string handling functions.
5. Write a program to find area of triangle and volume of rectangle using inheritance.
6. Write a program to implement the concept of packages.
7. Write a program to create multiple threads using thread class.
8. Write a program to implement the concept of exception handling.
9. Write a program to copy a file into another.
10. Write a program to display a text in the applet window using Applet.
11. Write a program to display geometrical objects using Applet.

Course Outcomes

Upon completion of this course, the students will be able to:

1. Ability to create, modify and delete database tables using DDL commands (CREATE, ALTER, DROP).
2. Proficiency in using DML commands (INSERT, UPDATE, DELETE) to manipulate data stored in the database.
3. Capability to utilize SQL functions, aggregate functions and set operations to perform complex data operations.
4. Proficiency in using control structures like while loop and for loop to manipulate program flow.
5. Capability to utilize object-oriented programming concepts like inheritance and method overloading to design efficient programs.

Reference Books

1. Ivan Bayross, SQL, PL/SQL The Programming Language of Oracle, BPB Publications, Third Revised Edition.
2. E. Balagurusamy, Programming With Java a Primer, McGraw Hill Education, Fifth Edition, 2018.

Course	BCA	Part III	Major Elective – I (a)	Semester	III
Major	CA	Computer Organization and Architecture		Credit	4
				Hours	4

Course Objectives

1. To enable students to understand the fundamentals of digital logic and develop skills in designing and analyzing combinational logic circuits using Boolean algebra, truth tables and Karnaugh maps.
2. To understand the fundamental concepts of data processing circuits and arithmetic circuits, including multiplexers, demultiplexers, decoders, encoders, half adders, full adders and fast adders.
3. To understand the fundamentals of Flip-Flops, Shift Registers and Counters in digital circuits, including the operation and design of various types of Flip-Flops and Counters.
4. To understand the concepts of register transfer and micro operations and to learn about register transfer language, arithmetic and logic micro operations and the arithmetic logic shift unit.
5. Understand the organization and architecture of the central processing unit, including general register organization, stack organization, instruction formats and addressing modes.

UNIT I

Digital Logic: The Basic Gates – NOT, OR, AND – Universal Logic Gates – NOR, NAND - Combinational Logic Circuits: Boolean Laws and Theorems – Sum of Products Method – Product of Sums Method - Truth Table to Karnaugh Map – Pairs, Quads and Octets – Karnaugh Simplifications.

UNIT II

Data Processing Circuits: Multiplexers – Demultiplexers – Decoders – Encoders – Arithmetic Circuits: Arithmetic Building Blocks – Half Adder – Full Adder – The Adder-Subtract – Fast Adder (Parallel Adder).

UNIT III

Flip-Flops: SR Flip-Flop – D Flip-Flop – JK Flip-Flop – T Flip-Flop – Shift Registers – Binary Counters – Asynchronous Counters – Synchronous Counters.

UNIT IV

Register Transfer and Micro Operations: Register Transfer Language – Register Transfer – Arithmetic Micro Operations – Logic Micro Operations – Shift Micro Operations – Arithmetic Logic Shift Unit.

UNIT V

Central Processing Unit: General Register Organization – Stack Organization – Instruction Formats: Three-Address Instructions – Two-Address Instructions – One-Address Instructions – Zero-Address Instructions – Addressing Modes.

Course Outcomes

Upon completion of this course, the students will be able to:

1. Understand the principles of digital logic and combinational logic circuits, including Boolean laws and theorems, Karnaugh map simplifications and universal gates.
2. Analyze and design basic data processing circuits, including multiplexers, demultiplexers, decoders and encoders, as well as arithmetic circuits such as adders and subtractors.
3. Understand the operation of flip-flops, shift registers and counters, both asynchronous and synchronous and use them in practical applications.

4. Learn about register transfer language, arithmetic and logic micro-operations and arithmetic logic shift units.
5. Understand the organization of a Central Processing Unit (CPU), including general register organization, stack organization and instruction formats, including addressing modes.

Text Books

1. Donald P. Leach | Albert Paul Malvino | Goutam Saha, Digital Principles and Applications, McGraw Hill Education, Eighth Edition, 2018.
UNIT I: Chapter 2 (Sections 2.1 and 2.2) and Chapter 3 (Sections 3.1 to 3.5 and 3.7).
UNIT II: Chapter 4 (Sections 4.1 to 4.3 and 4.6) and Chapter 6 (Sections 6.7 to 6.9).
2. Morris Mano M, Computer System Architecture, Pearson Revised Third Edition, 2018.
UNIT III: Chapter 2 (Section 2.6) and Chapter 3 (Sections 3.5 and 3.6)
UNIT IV: Chapter 5 (Sections 5.1, 5.2 and 5.4 to 5.7)
UNIT V: Chapter 9 (Sections 9.2 to 9.5)

Reference Books

1. Morris Mano M and Michael D. Ciletti, Digital Design, Pearson, Sixth Edition, 2018.
2. Thomas C. Bartee, Digital Computer Fundamentals, Tata McGraw Hill, 6th Edition, 25th Reprint, 2006.

Course	BCA	Part III	Major Elective – I (b)	Semester	III
Major	CA	Artificial Intelligence		Credit	4
				Hours	4

Course Objectives

1. To learn various concepts of AI Techniques.
2. To learn various Search Algorithm in AI.
3. To learn probabilistic reasoning and models in AI.
4. To learn about Markov Decision Process.
5. To learn various type of Reinforcement learning.

UNIT I

Introduction: Concept of AI – History – Current Status – Scope – Agents – Environments – Problem Formulations – Review of Tree and Graph Structures – State Space Representation – Search Graph and Search Tree.

UNIT II

Search Algorithms: Random Search – Search with Closed and Open List – Depth First and Breadth First Search – Heuristic Search – Best First Search – A* Algorithm – Game Search.

UNIT III

Probabilistic Reasoning: Probability – Conditional Probability – Bayes Rule – Bayesian Networks – Representation – Construction and Inference – Temporal Model – Hidden Markov Model.

UNIT IV

Markov Decision Process: Mdp Formulation – Utility Theory – Utility Functions – Value Iteration – Policy Iteration and Partially Observable Mdps.

UNIT V

Reinforcement Learning: Passive Reinforcement Learning – Direct Utility Estimation – Adaptive Dynamic Programming – Temporal Difference Learning – Active Reinforcement Learning – Q Learning.

Course Outcomes

On completion of this course, students will

1. Understand the various concepts of AI Techniques.
2. Understand various Search Algorithm in AI.
3. Understand probabilistic reasoning and models in AI.
4. Understand Markov Decision Process.
5. Understand various type of Reinforcement learning Techniques.

Text Books

1. Stuart Russell and Peter Norvig, “Artificial Intelligence: A Modern Approach” , 3rd Edition, Prentice Hall.
2. Elaine Rich and Kevin Knight, “Artificial Intelligence”, Tata McGraw Hill

Reference Books

1. Trivedi, M.C., “A Classical Approach to Artificial Intelligence”, Khanna Publishing House, Delhi.
2. Saroj Kaushik, “Artificial Intelligence”, Cengage Learning India, 2011.

Course	BCA	Part IV	Non-Major Elective – I (a)	Semester	III
Major	CA	HTML Basics		Credit	2
				Hours	2

Course Objectives

1. Design and create basic web pages using HTML elements, including HTML documents, anchor tags and hyperlinks. To enable students to create a basic webpage using HTML, including the use of header sections, titles, prologues, links and comment lines to produce a visually appealing and well-organized webpage.
2. To understand and apply the body section elements in HTML, including headings, horizontal rule, paragraph, tab settings and images and pictures.
3. To be able to create both unordered and ordered lists in HTML and understand their differences and use cases.
4. To understand the creation of tables in HTML, including specifying table width, cell width and alignment, as well as colouring cells and specifying columns.
5. To teach students how to use HTML to create framesets with nested frames and to understand the different units of measurement that can be used to define frame size.

UNIT I

Introduction to HTML: HTML Documents – Anchor Tag – Hyper Links – Header Section – Title – Prologue – Links – Colourful Web Page – Background Design – Background Colour – Text Colour – Comment Lines.

UNIT II

Designing the Body Section: Heading Printing – Aligning the Headings – Horizontal Rule – Paragraph – Tab Settings – Images and Pictures – Low Resolution Images – Adding Space around the Image – Border for an Image.

UNIT III

Ordered and Unordered Lists: Lists – Unordered Lists – Bullets in a List – Plain Attribute – Headings in a List – Ordered Lists – Type Attribute – Nested Lists.

UNIT IV

Table Handling: Tables – Tables Creation in HTML – Width of the Table and Cells – Cells Spanning Multiple Rows / Columns – Colouring Cells – Column Specification.

UNIT V

Frames: The Frameset Definition – Pixel Unit – Percentage Unit – Fraction Unit – Frame Definition – Nested Framesets.

Course Outcomes

Upon completion of this course, the students will be able to:

1. Understand the fundamentals of HTML, including the basic syntax and structure of HTML documents. Create and format headers, titles and prologues using HTML tags and attributes.
2. Utilize HTML tags to create and format body sections, including headings, paragraphs and images.
3. Create lists using HTML tags to organize content in a structured manner.
4. Design tables using HTML tags to present data in an organized and visually appealing way, including specifying table width and cell alignment, as well as colouring cells and columns.
5. Apply frameset definitions to create nested frames using various units of measurement such as pixels, percentages and fractions.

Text Book

1. C. Xavier, World Wide Web Design with HTML, McGraw-Hill Education, 2019.

Reference Book

1. N.P. Gopalan, J. Akilandeswari, Web Technology – A Developer’s Perspective, Prentice Hall of India Private Ltd, New Delhi, 2007.

Course	BCA	Part IV	Non-Major Elective – I (b)	Semester	III
Major	CA	E-Commerce		Credit	2
				Hours	2

Course Objectives

1. To introduce the framework of electronic commerce, discuss the convergence of electronic commerce and media and explore the anatomy of e-commerce applications for consumers and organizations.
2. To understand the components of the I-way and the network access equipment, as well as the architecture and components of the Internet as a network infrastructure for electronic commerce.
3. To understand the architectural framework and technology behind consumer-oriented electronic commerce, including Mercantile Process Models and the role of the World Wide Web in electronic commerce, as well as security considerations
4. To understand the various types of electronic payment systems, their features and risks and how to design effective electronic payment systems.
5. To understand the importance of corporate digital libraries as an internal e-commerce system, to identify the various types of digital documents and to explore the issues related to document infrastructure and corporate data warehouses.

UNIT I

Electronic Commerce: Electronic Commerce Framework – Electronic Commerce and Media Convergence – The Anatomy of E-Commerce Applications – Electronic Commerce Consumer Applications.

UNIT II

The Network Infrastructure for Electronic Commerce: Components of the I-way – Network Access Equipment. The Internet as a Network Infrastructure: NSFNET Architecture and Components.

UNIT III

Electronic Commerce and the World Wide Web: Architectural Framework for Electronic Commerce – World Wide Web as the Architecture – Technology behind the Web – Security and the Web.

UNIT IV

Electronic Payment Systems: Types of Electronic Payment Systems – Digital Token-based Electronic Payment Systems – Smart Cards and Electronic Payment Systems – Credit Card-Based Electronic Payment Systems.

UNIT V

The Corporate Digital Library: Dimensions of Internal Electronic Commerce Systems – Making a Business Case for a Document Library – Types of Digital Documents.

Course Outcomes

1. Understand the concept of electronic commerce and its framework.
2. Identify the components of the I-way and network access equipment.
3. Explain the World Wide Web as the architecture for electronic commerce and its technology behind the Web.
4. Analyze the types of electronic payment systems and the risks associated with them.
5. Evaluate the dimensions of internal electronic commerce systems, making a business case for a document library and the issues behind document infrastructure.

Text Book

1. Ravi Kalakota & Andrew B. Whinston, Frontiers of Electronic Commerce, Pearson Edition, India, 2009.

Reference Book

1. Manish Chandra Thrived, Electronic Commerce, Jiao Publishing House, 3rd Edition, 2006.

Course	BCA	Part III	Core VII	Semester	IV
Major	CA	Programming in Java		Credit	5
				Hours	5

Course Objectives

1. To introduce students to the fundamental concepts of programming in Java, including program structure, syntax, data types, operators, expressions, decision making and loop constructs.
2. To teach students how to create and work with classes, objects, constructors, method overloading, inheritance and arrays in Java programming language.
3. To learn how to create and use interfaces, work with Java packages and develop multithreaded programs in Java.
4. To learn how to manage errors and exceptions in Java and manage input/output files using stream classes, byte and character stream classes and file creation/reading/writing techniques.
5. To enable students to develop skills in applet program and graphics programming in Java.

UNIT I

Simple Java Program – Java Program Structure – Java Tokens – Implementing a Java Program – Java Virtual Machine – Command Line Arguments – Constants – Variables – Data types – Operators and Expressions – Decision Making : Branching and Looping.

UNIT II

Defining a Class – Creating Objects – Accessing Class Members – Constructors – Method Overloading – Static Members – Inheritance: Extending a Class – Overriding Methods – Arrays: One-dimensional Arrays – Creating an Array – Two-dimensional Arrays – Strings.

UNIT III

Interfaces: Defining Interfaces – Extending Interfaces – Implementing Interfaces – Accessing Interface Variables – Packages: Java API Packages – Creating Packages – Accessing a Package – Using a Package – Multithreaded Programming: Creating Threads – Extending the Thread Class – Stopping and Blocking a Thread – Life Cycle of a Thread.

UNIT IV

Managing Errors and Exceptions: Types of Errors – Exceptions – Syntax of Exception Handling Code – Multiple Catch Statements – Using Finally Statement – Managing Input / Output Files in Java: Stream Classes – Byte and Character Stream Classes – Creation of Files – Reading / Writing Characters.

UNIT V

Applet Programming: Building Applet Code – Applet Life Cycle – Creating an Executable Applet – Applet Tag – Running the Applet – Graphics Programming: The Graphics Class – Lines and Rectangles – Circles and Ellipses – Drawing Arcs – Drawing Polygons.

Course Outcomes

Upon completion of this course, the students will be able to:

1. Design and implement Java programs using the basic syntax, data types, operators and expressions, including decision making and looping structures.
2. Create and use classes, objects, constructors, methods and arrays and understand inheritance and method overloading.
3. Understand and use interfaces and packages in Java, as well as multithreaded programming concepts.

4. Implement proper error and exception handling in Java, as well as manage input/output files using stream classes.
5. Develop Java applets and graphics programs, including knowledge of the applet life cycle and the Graphics class.

Text Book

1. E. Balagurusamy, Programming With Java a Primer, McGraw Hill Education, Fifth Edition, 2018.

UNIT I : Chapter 3 (3.2, 3.5, 3.6, 3.9 to 3.11), Chapter 4 (4.2 to 4.6 & 4.10),
Chapter 5 (5.2 to 5.9), Chapter 6 (6.3, 6.4, 6.6 & 6.7) and Chapter 7 (7.2 to 7.4).

UNIT II : Chapter 8 (8.2 to 8.9, 8.11 & 8.12) and Chapter 9 (9.2 to 9.5).

UNIT III : Chapter 10 (Sections 10.2 to 10.5), Chapter 11 (11.2, 11.5 to 11.7)
and Chapter 12 (12.2 to 12.5).

UNIT IV : Chapter 13 (13.2 to 13.6) and Chapter 16 (16.3 to 16.5, 16.10 & 16.11)

UNIT V : Chapter 14 (14.4 to 14.6, 14.8 & 14.10) and Chapter 15 (15.2 to 15.6).

Reference Book

1. P. Radha Krishna, Object Oriented Programming through Java, University Press (India) Private Ltd (2007).

Course	BCA	Part III	Core VIII	Semester	V
Major	CA	Operating Systems		Credit	5
				Hours	5

Course Objectives

1. Gain an in-depth understanding of the basic concepts, terminologies and different viewpoints of an Operating System and its role as a resource manager, as well as gain practical knowledge about I/O programming and different types of I/O channels.
2. Analyze and evaluate the different techniques for memory management in operating systems, including their advantages and limitations and learn how to choose the appropriate memory management technique for specific scenarios.
3. Develop a comprehensive understanding of processor management, process synchronization and job scheduling in operating systems and be able to evaluate and optimize their performance in different contexts.
4. Acquire knowledge and skills in device management, including techniques for managing I/O devices, I/O traffic control, I/O scheduling and I/O device handlers, as well as understanding the concepts of virtual devices and virtual systems.
5. Gain a thorough understanding of the concepts of information management in operating systems, including file systems, access control verification and the difference between logical and physical file systems and learn how to design and implement efficient file systems.

UNIT I

Basic Concepts and Terminologies – Operating System as Resource Manager – Process View Point – Hierarchical Machine View – Extended Machine View – I/O Programming: Types of I/O Channels.

UNIT II

Memory Management – Single Contiguous Allocation – Partitioned Allocation – Relocatable Partitioned Memory Management – Paged Memory Management – Demand Paged Memory Management.

UNIT III

Processor Management – Process State Model – Job Scheduling Functions – Job Scheduling in Nonmultiprogrammed Environment – Process Scheduling Functions – Process Synchronization – Race Condition – Synchronization Mechanisms – Deadly Embrace.

UNIT IV

Device Management – Techniques for Device Management – I/O Traffic Controller – I/O Scheduler – I/O Device Handlers – Virtual Devices – Virtual System.

UNIT V

Information Management – General Model of a File System – Symbolic File System – Basic File System – Access Control Verification – Logical File System – Physical File System.

Course Outcomes

Upon completion of this course, the students will be able to:

1. Understand the basic concepts and terminologies of operating systems, including its role as a resource manager and the different viewpoints.
2. Analyze and compare the different techniques used for memory management in operating systems.
3. Analyze and evaluate the concepts of processor management, process synchronization and job scheduling in operating systems.

4. Understand the different techniques and strategies for managing I/O devices and virtual systems in operating systems.
5. Develop an understanding of the concepts of information management in operating systems, including file systems, access control verification and logical and physical file systems.

Text Book

1. Stuart E. Madnick & John J. Donovan, Operating Systems, McGraw Hill Education, 2017.

Unit I : Chapter 1 (1.2 to 1.5) and Chapter 2 (2.3, 2.3.1).

Unit II : Chapter 3 (3.1, 3.3 to 3.6).

Unit III : Chapter 4 (4.1, 4.2.1, 4.2.3, 4.3.1, 4.5.1 to 4.5.3).

Unit IV : Chapter 5 (5.1, 5.5.1 to 5.5.3 and 5.6.2).

Unit V : Chapter 6 (6.3 to 6.8).

Reference Book

1. James L. Peterson & Abraham Silberschatz, An Introduction to Operating Systems, Addison-Wesley Publishing Co., New York (1987).

Course	BCA	Part III	Core IX	Semester	V
Major	CA	JavaScript Programming		Credit	5
				Hours	5

Course Objectives

1. To provide students with a foundational understanding of JavaScript and its integration with HTML, covering topics such as creating and structuring JavaScript code, defining variables and data types and working with functions.
2. To deepen students' understanding of JavaScript operators, including arithmetic, assignment, comparison and logical operators and how they can be used to manipulate data.
3. To teach students how to use conditional statements and loops in JavaScript for control flow and how to define and use objects to store and manipulate data.
4. To provide students with an understanding of the document object in JavaScript and its properties and methods, as well as the use of event handlers to handle user events.
5. To enable students to apply JavaScript concepts related to forms and frames, including accessing and manipulating form elements, using form properties and methods and working with frames.

UNIT I

Placing JavaScript in an HTML file: Creating First Script – External JavaScript Files – Variables: Defining Variables for Scripts – Data Types – Variables in Script – Writing a Page of JavaScript – Structuring Functions.

UNIT II

JavaScript Operators: Operator Types – Arithmetic Operators – Assignment Operators – Comparison Operator – Logical Operators.

UNIT III

Conditional Statements and Loops: Using Conditional Statements: if /else – Switch – Conditional Operator – User Input from a Prompt – Using Loops: for – while – do while – for in and for each in – break and continue – Objects: Defining Objects – Creating Objects.

UNIT IV

The Document Object: Defining the document object – Document object model – Properties of the document object – Methods of the document object. Event Handlers: What is an Event Handler? – Learning the Events – Other Ways to Register Events.

UNIT V

JavaScript and Forms: Accessing Forms – Properties and Methods of the Form Object – Proper Element and Label Order – <label> </label> Tags – <fieldset> </fieldset> Tags. JavaScript and Frames: Purpose of Frames – Accessing Frames – Breaking out of Frames.

Course Outcomes

Upon completion of this course, the students will be able to:

1. Develop a foundational understanding of JavaScript by learning to place it in an HTML file, create scripts, define variables and data types, structure functions and write a page of JavaScript.
2. Learn about different types of JavaScript operators, including arithmetic, assignment, comparison and logical operators.
3. Understand how to use conditional statements, loops and objects in JavaScript, including if/else statements, switch statements, for and while loops and defining and creating objects.

4. Explore the document object model and event handlers in JavaScript, including defining the document object, understanding its properties and methods and registering events.
5. Discover how to access and manipulate forms in JavaScript, including accessing form properties and methods, using proper element and label order and working with frames

Text Book

1. Java Script: A Beginner's Guide 4th Edition, John Pollock, Tata MC Graw-Hill Edition 2013.

Reference Book

1. N.P. Gopalan, J. Akilandeswari, Web Technology – A Developer's Perspective, Prentice Hall of India Private Ltd, New Delhi, 2007.

Course	BCA	Part III	Core X	Semester	V
Major	CA	Computer Networks		Credit	5
				Hours	6

Course Objectives

1. To provide a comprehensive understanding of networks, including data flow, connection types, topologies and OSI model layers to enable students to design and implement network systems.
2. To equip students with an advanced knowledge of physical layer concepts, various guided media types and switching techniques so that they can proficiently create, implement and evaluate robust and secure communication systems
3. To develop students' expertise in error detection and correction techniques and protocols for noiseless and noisy channels at the data link layer.
4. To enable students to acquire in-depth knowledge of connecting devices and routing algorithms used in the network layer of computer networks.
5. To teach students about transport layer and TCP/IP protocol suite for process to process delivery, as well as application layer protocols including DNS, FTP, WWW architecture and HTTP.

UNIT I

Networks: Data flow – Types of Connection – Topologies of Networks – Categories of Networks – The Open System Interconnection (OSI) Model: Layers in the OSI Model – Physical – Data Link – Network – Transport – Session – Presentation - Application.

UNIT II

Physical Layer: Multiplexing – Frequency Division Multiplexing – Wave Division Multiplexing – Time Division Multiplexing. Guided Media: Twisted Pair – Coaxial Cable – Fiber Optic Cable. Switching: Circuit Switching – Packet Switching – Message Switching.

UNIT III

Data Link Layer: Error Detection – Vertical Redundancy Check – Longitudinal Redundancy Check – Cyclic Redundancy Check – Checksum – Error Correction. Protocols: Noiseless Channels – Simplest Protocol – Stop and Wait Protocol – Noisy Channels – Stop and Wait ARQ – Go-Back-N ARQ – Selective Repeat ARQ.

UNIT IV

Network Layer: Connecting Devices – Passive Hubs – Repeaters – Active Hubs – Bridges – Two Layer Switches – Routers – Three Layer Switches – Gateways. Routing Algorithms: Distance Vector Routing Algorithm – Link State Routing Algorithm.

UNIT V

Transport Layer: Process to Process Delivery – TCP/IP Protocol Suite. Application Layer: Domain Name System – File Transfer Protocol – WWW Architecture – Hypertext Transfer Protocol.

Course Outcomes

Upon completion of this course, the students will be able to:

1. Understand the fundamentals of networking, including data flow, types of connections, topologies and categories of networks.
2. Describe the OSI model and its layers, including physical, data link, network, transport, session, presentation and application layers.

3. Understand the principles of multiplexing, guided media and switching in the physical layer of networking.
4. Explain error detection and correction techniques, protocols for noiseless and noisy channels and various network layer devices such as hubs, bridges, routers and gateways.
5. Learn about process-to-process delivery in the transport layer and different application layer protocols, including DNS, FTP, WWW architecture and HTTP.

Text Book

1. Behrouz A. Forouzan, Data Communications and Networking, Tata McGraw-Hill, Fourth Edition, 2017.

Reference Book

1. William Stallings, Data and Computer Communication, PHI, Tenth Edition (2013).

Course	BCA	Part III	Core XI - Practical	Semester	V
Major	CA	Scripting Languages Lab		Credit	5
				Hours	5

Course Objectives

1. Develop JavaScript programs to perform basic arithmetic operations, validate user input and display results using alerts.
2. Use JavaScript to create dynamic effects such as growing or shrinking text, sorting numbers and displaying tables.
3. Design PHP web pages that demonstrate the use of loops to generate multiplication tables, prime number series and maximum values in multidimensional arrays.
4. Develop PHP programs that showcase the use of conditional statements, string functions and arrays to manipulate data.
5. Implement PHP programs that create dynamic web pages with interactive features, such as online quizzes, random string generation and visitor counters.

JAVASCRIPT

1. Write a JavaScript program to display the current day and time.
2. Write a JavaScript program to find the area of a triangle by getting input from the user.
3. Write a JavaScript program to design a simple calculator and perform operations: addition, subtraction, multiplication and division of two numbers (input from user).
4. Write a JavaScript program using conditional statement to find the largest of five numbers. Display an alert box to show the result.
5. Write a JavaScript program to find whether the given number (3 digits) is Armstrong or not.
6. Write a JavaScript program to sort the given numbers in ascending order.
7. Write a JavaScript that calculates the squares and cubes of the numbers from 0 to 10 and outputs HTML text that displays the resulting values in an HTML table format.
8. Write a JavaScript code that displays text "TEXT-GROWING" with increasing font size in the interval of 100ms in RED COLOR, when the font size reaches 50pt it displays "TEXT-SHRINKING" in BLUE color. Then the font size decreases to 5pt.
9. Create a HTML form that has 'n' number of Textboxes. When the form runs in the Browser fill the textboxes with data. Write JavaScript code that verifies that all textboxes has been filled. If a textboxes has been left empty, popup an alert indicating which textbox has been left empty.
10. Create a page with dynamic effects. Write the code to include layers and basic animation.
11. Write a JavaScript code to find the sum of N natural Numbers. (Use user-defined function).

PHP

1. Design a PHP web page to generate multiplication table for a given number.
2. Write a program to demonstrate the use of conditional statements in PHP.
3. Design a PHP web page to generate prime number series.
4. Develop a PHP program using string functions.
5. Develop a PHP program to create a random string using functions.
6. Develop a PHP program to find the first vowel in a given string.
7. Develop a PHP program to reverse a given number.
8. Develop a PHP program to design an online quiz.
9. Develop a PHP program to change web page colour based on random number generation using reference button.
10. Develop a PHP program using arrays as input from user.
11. Develop a PHP script to create a visitor counter.
12. Develop a PHP program to find the maximum value in multidimensional array.

Course Outcomes

Upon completion of this course, the students will be able to:

1. Ability to use JavaScript to implement various functionalities on web pages, such as displaying current date and time, calculating triangle area and sorting numbers.
2. Proficiency in using JavaScript to create interactive web page elements, such as simple calculators, dynamic text effects, form validation, animations and user-defined functions for mathematical operations.
3. Design PHP web pages for generating multiplication tables, prime numbers series and implementing an online quiz.
4. Develop PHP programs using string functions, arrays as input from the user and finding the maximum value in a multidimensional array.
5. Use PHP to demonstrate the use of conditional statements, generate random strings, find the first vowel in a given string, reverse a given number and change the color of a web page based on random number generation.

Reference Books

1. Java Script: A Beginner's Guide 4th Edition, John Pollock, Tata MC Graw-Hill Edition 2013.
2. Steven Holzner, The Complete Reference PHP, McGraw Hill Education, Indian Edition, 2018.

Course	BCA	Part III	Major Elective – II (a)	Semester	V
Major	CA	PHP Programming		Credit	3
				Hours	5

Course Objectives

1. To enable students to learn the basics of PHP, including how to create and run PHP pages, work with variables and constants and use various types of operators.
2. To teach students how to use conditional statements and loops in PHP, including the if/else, elseif and switch statements, as well as different types of loops such as for, while, do...while and foreach loops and to introduce string formatting and array manipulation in PHP.
3. To enable students to understand how to create functions in PHP, pass data to functions, use default arguments, pass variable numbers of arguments and return data from functions.
4. Developing skills in creating web pages to reading data using text fields, text areas, check boxes, radio buttons, list boxes, password controls, image maps and buttons.
5. To enable students to learn how to read data from web pages using different form elements and how to handle files in PHP, including opening, reading and closing files using various file handling functions and teach how to use cookies.

UNIT I

Creating First PHP Page – Running First PHP Page – Printing Text – Working with Variables – Storing data in Variables – Creating Variable Variables – Creating Constants – Data Types Operators: Math – Assignment – Increment – Decrement – String – Bitwise – Execution – Comparison – Logical - Ternary.

UNIT II

If Statement – Else Statement – Elseif Statement – Switch Statement – For Loops – While Loops – Do...While Loops – Foreach Loops. String: Formatting Text Strings. Arrays: Building Arrays – Modifying Data in Arrays – Deleting Array Elements – Extracting Data from Arrays – Sorting Arrays – Comparing Arrays.

UNIT III

Functions: Creating Functions in PHP – Passing Data to Functions – Passing by Reference – Using Default Arguments – Passing Variable Numbers of Arguments – Returning Data from Functions.

UNIT IV

Reading Data in Web Pages: Text Fields – Text Areas – Check Boxes – Radio Buttons – List Boxes – Password Controls – Image Maps – Buttons.

UNIT V

File Handling: Opening files using fopen – Looping over File's Contents with feof – Reading Text from File using fgets – Closing a File. Cookies: Setting a Cookie – Reading a Cookie – Setting Cookies' Expiration – Deleting Cookies.

Course Outcomes

Upon completion of this course, the students will be able to:

1. Develop the ability to write and execute basic PHP programs to print text, work with variables and utilize operators to perform math and string operations.
2. Demonstrate proficiency in using conditional statements like if-else, elseif and switch statements, as well as different types of loops, including for, while, do-while and foreach loops.

3. Build skills in defining and calling functions in PHP, passing arguments, returning values and handling variable numbers of arguments.
4. Understand how to create web forms with various types of input fields, such as text fields, text areas, checkboxes, radio buttons and list boxes and process the data submitted by these forms in PHP.
5. Acquire proficiency in handling file operations in PHP, including opening and closing files, looping over file contents and reading text from files. Additionally, develop skills in setting and manipulating cookie, such as reading a cookie, setting expiration and deleting cookies.

Text Book

1. Steven Holzner, The Complete Reference PHP, McGraw Hill Education, Indian Edition, 2018.

Reference Book

1. Vikram Vaswani, A Beginner's Guide: PHP, McGraw Hill Education, Indian Edition, 2017.

Course	BCA	Part III	Major Elective – II (b)	Semester	V
Major	CA	Introduction to Data Science		Credit	3
				Hours	5

Course Objectives

1. To learn about basics of Data Science and Big data.
2. To learn about overview and building process of Data Science.
3. To learn about various Algorithms in Data Science.
4. To learn about Hadoop Framework.
5. To learn about case study about Data Science.

UNIT I

Introduction: Benefits and Uses – Facts of Data – Data Science Process – Big Data Ecosystem and Data Science.

UNIT II

The Data Science Process: Overview – Research Goals – Retrieving Data – Transformation – Exploratory Data Analysis – Model Building.

UNIT III

Algorithms: Machine Learning Algorithms – Modelling Process – Types – Supervised – Unsupervised – Semi-Supervised.

UNIT IV

Introduction to Hadoop: Hadoop Framework – Spark – Replacing Map Reduce – Nosql – ACID – CAP – BASE – Types.

UNIT V

Case Study: Prediction of Disease – Setting Research Goals – Data Retrieval – Preparation – Exploration – Disease Profiling – Presentation And Automation.

Course Outcomes

On completion of this course, students will

1. Understand the basics in Data Science and Big data.
2. Understand overview and building process in Data Science.
3. Understand various Algorithms in Data Science.
4. Understand Hadoop Framework in Data Science.
5. Case study in Data Science.

Text Book

1. Davy Cielen, Arno D. B. Meysman, Mohamed Ali, “Introducing Data Science”, manning publications 2016.

Reference Books

1. Roger Peng, “The Art of Data Science”, lulu.com 2016.
2. MurtazaHaider, “Getting Started with Data Science – Making Sense of Data with Analytics”, IBM press, E-book.
3. Davy Cielen, Arno D.B. Meysman, Mohamed Ali, “Introducing Data Science: Big Data, Machine Learning and More, Using Python Tools”, Dreamtech Press 2016.
4. Annalyn Ng, Kenneth Soo, “Numsense! Data Science for the Layman: No Math Added”, 2017, 1st Edition.

5. Cathy O'Neil, Rachel Schutt, "Doing Data Science Straight Talk from the Frontline", O'Reilly Media 2013.
6. Lillian Pierson, "Data Science for Dummies", 2017 II Edition

Course	BCA	Part IV	Non-Major Elective – II (a)	Semester	V
Major	CA	Computer Fundamentals		Credit	2
				Hours	2

Course Objectives

1. Gain an understanding of the history and evolution of computer technology, including the first to fifth generation of computers and their key characteristics. Classify various types of computers, such as tablets, laptops, PCs, servers, mainframes and supercomputers and their respective applications.
2. Identify different input devices such as keyboard, mouse, touchpad, touch screen and magnetic ink character recognition (MICR) and their functionalities.
3. Understand the functioning of various output devices such as flat-panel displays, printers (inkjet, laser, dot matrix and line printers) and their respective advantages and disadvantages.
4. Gain knowledge about primary memory (RAM and ROM) and secondary memory (magnetic disk and tape) and their importance in computer storage.
5. Develop an understanding of the different number systems, including decimal, binary, octal and hexadecimal and be able to perform conversions between them.

UNIT I

Generation of Computers: First Generation – Second Generation – Third Generation – Fourth Generation – Fifth Generation – Classification of Computers: Tablets – Laptop – PC – Server – Main Frame – Super Computers.

UNIT II

Input Devices: Keyboard – Mouse – Touch Pad – Touch Screen – Magnetic Ink Character Recognition (MICR).

UNIT III

Output Devices: Flat Panel Display – Printers – Inkjet Printers – Laser Printers – Dot Matrix Printers – Line Printers.

UNIT IV

Primary Memory: Random Access Memory (RAM) – Read Only Memory (ROM) – Secondary Memory: Magnetic Disk – Magnetic Tape.

UNIT V

Number System: Decimal – Binary – Octal – Hexadecimal – Number System Conversion – Binary to Decimal – Binary to Octal – Binary to Hexadecimal – Decimal to Binary – Decimal to Octal – Decimal to Hexadecimal.

Course Outcomes

Upon completion of this course, the students will be able to:

1. Demonstrate an understanding of the history and evolution of computer technology, from the first generation to the fifth generation of computers and their respective characteristics.
2. Classify different types of computers, including tablets, laptops, PCs, servers, mainframes and supercomputers and their respective applications, based on their functionalities.
3. Apply knowledge of input and output devices, including keyboard, mouse, touchpad, touch screen, magnetic ink character recognition (MICR), flat-panel displays and printers, to effectively interact with computer systems.
4. Evaluate the different types of computer memory, including primary memory (RAM and ROM) and secondary memory (magnetic disk and tape) and understand their respective functions and importance in computer storage.

5. Apply knowledge of the various number systems, including decimal, binary, octal and hexadecimal and their respective conversions to solve problems and interpret data in computer systems.

Text Book

1. Rajaraman V and Neeharika Adabala – Fundamentals of Computers, Sixth Edition, PHI, 2018.

Reference Book

1. Turban-Rainer-Potter, Introduction to Information Technology, Second Edition, Wiley Publications, 2007.

Course	BCA	Part IV	Non-Major Elective – II (b)	Semester	V
Major	CA	Multimedia Systems		Credit	2
				Hours	2

Course Objectives

1. To introduce students to the concept of multimedia, its applications and the use of text in multimedia by providing an understanding of font types, font faces, designing text and editing fonts using tools such as Fontographer.
2. Designing multimedia still images and audio, including bitmap and vector drawings, 3-D rendering and digital audio editing, while exploring file formats, compression techniques and MIDI audio.
3. To enable students to understand and apply the principles and techniques of animation and video production, including file formats, shooting, editing, storyboarding, lighting and chroma keys.
4. To comprehend the stages, skills and hardware components involved in a multimedia project and the significance of intangibles like creativity, organization and communication, as well as the importance of teamwork.
5. To gain knowledge about the design elements and internet-related aspects of multimedia projects, such as navigation, hyperlinks, bandwidth, web servers, browsers and search engines.

UNIT I

Multimedia: Definition – Where to Use Multimedia? – Text: Fonts and Faces – Using Text in Multimedia: Designing with Text – Fields for Reading – HTML Documents – Font Editing and Design Tools: Fontographer – Making Pretty Text.

UNIT II

Making Still Images: Bitmaps – Vector Drawing – 3-D Drawing and Rendering – Image File Formats – Image File Compression – Sound: Digital Audio – Making Digital Audio Files – Editing Digital Recordings – MIDI Audio – Audio File Formats.

UNIT III

Animation: Principles of Animation – Animation Techniques – Animation File Formats – Video: Analog Video – Digital Video – Shooting and Editing Video: Camcoders – Smartphones and Tablets – Storyboarding – Lighting – Chroma keys.

UNIT IV

Making Multimedia: The Stages of a Multimedia Project – What You Need: The Intangibles – Creativity – Organisation – Communication – Multimedia Skills: The Team – The Sum of Parts – Hardware: Memory and Storage Devices – Input Devices – Output Devices.

UNIT V

Designing the Structure: Navigation – Hotspots – Hyperlink – Buttons – Icons. The Internet and Multimedia: Internetworking – Internet Address – Connections – Bandwidth – Multimedia on the Web: Web Server – Web Browser – Search Engines.

Course Outcomes

Upon completion of this course, the students will be able to:

1. Understand the concept of multimedia and its use in different contexts and learn about text design and editing tools.

2. Acquire knowledge and skills related to creating still images, including bitmap and vector drawing, 3-D rendering and image file formats and compression and explore digital audio and MIDI.
3. Learn about animation principles and techniques, animation file formats and shooting and editing analog and digital video, including storyboarding, lighting and chroma keys.
4. Comprehend the stages and components involved in a multimedia project, including intangibles like creativity, organization and communication, multimedia skills and hardware components such as memory, storage, input and output devices.
5. Develop an understanding of the design elements of multimedia projects, such as navigation, hotspots, hyperlinks, buttons and icons and explore internet-related aspects such as internetworking, internet addresses, bandwidth, web servers, browsers and search engines.

Text Book

1. TayVaughan, Multimedia: Making it Work, McGraw-Hill Education, Ninth Edition, 2018.

Reference Book

1. David Hillman, Multimedia Technology and Applications, Galgotia Publications Pvt. Ltd., Year 1998.

Course	BCA	Part III	Core XII	Semester	VI
Major	CA	Dot Net Programming		Credit	6
				Hours	6

Course Objectives

1. To introduce students to the basic concepts of VB.Net programming language and enable them to create console applications.
2. Develop proficiency in using control statements and form controls such as if-else statements, looping statements and form controls including checkbox, radio button and picture box.
3. Design and implement applications using arrays and combo box controls in VB.NET and demonstrate proficiency in using the For-Each loop, ReDim statement and One-Dimensional and Multi-Dimensional arrays.
4. Understand and apply the concepts of procedures, functions and properties in VB.Net programming including the return statement, call by reference, message box and input box functions.
5. To be able to create menus and use dialog boxes in VB.Net programming.

UNIT I

VB.Net Program: Solution Explorer Window – Class View Window – Toolbox – Output Window – Task List Window – Data Types and Operators: Literal – Variables – Data Types – Declaration of Variables – Constant – Statements – Operators – Keywords – Comments – Scope of Variables – Console Applications in VB.Net.

UNIT II

Control Statements: If Statement – Block-if – Looping: For-To-Next – While-End While – Do-Loop – Select Case Statement – Goto Statement – Form Control – Events – Label – Textbox – Checkbox Control – Radio Button Control – Scroll Bar Control – Picture Box.

UNIT III

Arrays: One-Dimensional Array – Array Initialisation – For Each-Next Loop – Redim Statement – Multi-Dimensional Array – Initialization of Two-Dimensional Array – List Box Control – Checked List Box Control – Combo Box Controls.

UNIT IV

Procedures: Subroutine Procedures – Function Procedure – Property Procedure – Functions – Value Returned by its Function Name – Return Statement – Calling a Function – Call by Reference – Message Box Function – Input Box Function.

UNIT V

Creating Menus and using Dialog Boxes: Menu – Multiple Document Interface (MDI) Forms – Context Menu – Rich Text Box – Color Dialog Control – Font Dialog Control.

Course Outcomes

Upon completion of this course, the students will be able to:

1. Develop an understanding of the fundamentals of programming and basic concepts of VB.Net.
2. Understand and implement control statements to make decisions and control flow of the program.
3. Utilize arrays to organize and store data in VB.Net applications.
4. Design and implement procedures and functions to organize code and improve code reusability.

5. Create menus and dialog boxes to enhance the user interface and user experience of VB.Net applications.

Textbook

1. P. RadhaGanesan, VB.Net, Scitech Publications Pvt. Ltd., 2017.

Reference Books

1. Jeffrey R. Shapiro, The Complete Reference Visual Basic .Net, Mc Graw Hill Education, New Delhi, 2016.
2. Steven Holzner, Visual Basic .Net Black Book, 2007.

Course	BCA	Part III	Core XIII	Semester	VI
Major	CA	Programming in Python		Credit	5
				Hours	6

Course Objectives

1. To introduce the basics of Python programming language including data types, variables, operators, reserved words and input operations.
2. To teach students about decision control statements in Python such as if, if-else, nested if, if-elif-else, while and for loops and the use of break, continue, pass statements and else statements with loops.
3. To understand and apply the concepts of functions, including function definition, function call, the return statement and string methods and functions such as ord(), chr(), comparing strings, match(), search() and sub().
4. To teach the basics of object-oriented programming in Python, including the concepts of classes, objects, methods and inheritance.
5. To teach students how to handle exceptions and work with files in Python, including opening, writing to and reading from files, as well as how to handle multiple exceptions and use the try-except-finally blocks.

UNIT I

Basics of Python Programming: First Python Program – Literal Constants – Variables and Identifiers – Data Types – Input Operation – Reserved Words – Indentation – Operators – Operation on Strings – Other Data Types.

UNIT II

Decision Control Statements: if Statement – if-else Statement – Nested if Statement – if-elif-else Statement – While Loop – For Loop – Break Statement – Continue Statement – Pass Statement – Else Statement used with Loops.

UNIT III

Functions: Function Definition – Function Call – The Return Statement – Strings: Built-in String Methods and Functions – ord() Function – chr() Function – Comparing Strings – Match() Function – Search() Function – Sub() Function.

UNIT IV

Classes and Objects – Class Method and Self Argument – The_init() Method – Class Variables – Object Variables – The_del() Method – Inheritance – Inheriting Classes – Types of Inheritance.

UNIT V

Handling Exceptions – Multiple Except Blocks – Multiple Exceptions in a Single Block – Except Block without Exception – The Else Clause – The Finally Block – File Handling: The Open() Function – The Close() Function – Write() Method – Writelines() Method – Append() Method – Read() Method – Readline() Method.

Course Outcomes

Upon completion of this course, the students will be able to:

1. Demonstrate an understanding of the basics of Python programming, including data types, variables, operators and input/output operations.
2. Apply decision control statements such as if statements, loops and break/continue statements to control the flow of a Python program.

3. Develop and use functions, including the return statement and built-in string methods and functions, in Python programming.
4. Create and utilize classes and objects, including inheritance and different types of inheritance, in Python programming.
5. Handle exceptions and file input/output in Python programming using multiple except blocks, the else clause and the finally block, as well as the open(), close(), write(), append(), read() and readline() methods.

Text Book

1. Reema Thareja, Python Programming using Problem Solving Approach, Oxford University Press, 2017.

UNIT I : Chapter 3 (Sections 3.4 to 3.8, 3.10 to 3.12, 3.14 & 3.15).

UNIT II : Chapter 4 (Sections 4.2, 4.3, 4.5 to 4.8).

UNIT III : Chapter 5 (Sections 5.2, 5.3 & 5.5),
Chapter 6 (Sections 6.4, 6.6, 6.8, 6.11.1 to 6.11.3).

UNIT IV : Chapter 9 (Sections 9.2 to 9.6), Chapter 10 (Sections 10.2 & 10.3).

UNIT V : Chapter 12 (Sections 12.2 to 12.6, 12.11),
Chapter 7 (Sections 7.4.1 & 7.4.3, 7.5.1 to 7.5.3).

Reference Books

1. Ashok Namdev Kamthane and Amit Ashok Kamthane, Programming and Problem Solving with Python, McGraw Hill Education, 2018.
2. Martin C. Brown, The Complete Reference, Mc Graw Hill Education, 2018.

Course	BCA	Part III	Core – XIV	Semester	VI
Major	CA	Software Testing		Credit	5
				Hours	5

Course Objectives

1. To comprehend the phases of a software project and different testing techniques, including white-box and structural testing, to ensure quality and functional performance.
2. To gain knowledge and understanding of black box testing techniques, including boundary value analysis and decision tables and integration testing approaches, such as top-down and bottom-up integration.
3. Understand the overview of system testing and its different types, including functional testing, business vertical testing, deployment testing and beta testing. Learn about acceptance testing criteria and the process of selecting test cases
4. Gain an understanding of performance testing, including the factors that govern it, the methodology of collecting requirements, writing and automating test cases, executing test cases and analyzing performance test results.
5. Develop proficiency in regression testing, including knowledge of types such as smoke or sanity test, criteria for selecting and classifying test cases and methodologies for selecting and resetting test cases to conclude the test results.

UNIT I

Phases of Software Project – Quality, Quality Assurance and Quality Control – Testing, Verification and Validation – Waterfall Model – White Box Testing – Static Testing: Humans – Analysis Tools. Structural Testing: Unit/Code Functional Testing – Code Coverage Testing.

UNIT II

Black Box Testing: What is Black Box Testing? – Requirements Based Testing – Positive and Negative Testing – Boundary Value Analysis – Decision Tables. Integration Testing: What is Integration Testing? – Top-Down Integration – Bottom-up Integration.

UNIT III

System Testing: Overview – Functional Testing: Design/Architecture Verification – Business Vertical Testing – Deployment Testing – Beta Testing. Acceptance Testing: Acceptance Criteria – Selecting Test Cases.

UNIT IV

Performance Testing: Factors Governing Performance Testing – Methodology: Collecting Requirements – Writing Test Cases – Automating Performance Test Cases – Executing Performance Test Cases – Analyzing the Performance Test Results.

UNIT V

Regression Testing: What is Regression Testing? Types of Regression Testing – Smoke or Sanity Test – Criteria for Selecting Test Cases – Classifying Test Cases – Methodology for Selecting Test Cases – Resetting Test Cases – Concluding the Results.

Course Outcomes

Upon completion of this course, the students will be able to:

1. Understand the phases of software projects, including quality assurance and control, testing and validation.
2. Apply different testing techniques, such as white box, black box, integration, system, acceptance, performance and regression testing.
3. Develop proficiency in writing and executing test cases and analyzing the results.

4. Evaluate the various factors that govern performance testing and select appropriate tools and methodologies to automate the process.
5. Develop a critical understanding of testing methodologies and strategies and apply them to various software development projects.

Textbook

1. Srinivasan Desikan and Gopalaswamy Ramesh, Software Testing Principles and Practices, Pearson, 2017.

Reference Book

1. Aditya P. Mathur, Foundations of Software Testing, Second Edition, Pearson Education, 2013.

Course	BCA	Part III	Core XV - Practical	Semester	VI
Major	CA	Dot Net & Python Programming Lab		Credit	5
				Hours	6

Course Objectives

1. To develop proficiency in VB.NET programming.
2. To create interactive user interfaces using VB.NET.
3. To introduce fundamental programming concepts and syntax of Python.
4. To enable students to write simple programs and functions in Python.
5. To familiarize students with basic data structures and algorithms using Python.

VB.Net

1. Create a vb.net console based application to accept a character from console and check the case of the character.
2. Create a vb.net console based application to display a star pyramid.
3. Create a vb.net console based application to accept a character from console and check whether it is a vowel or not.
4. Create a vb.net console based application to accept a number from console and reverse it.
5. Create a vb.net window based application using dialog controls like openfile, color and font.
6. Create a vb.net window based application using datetimepicker control to display a calendar.
7. Create a vb.net window based application to implement a calculator.
8. Create a vb.net window based application to sort the items in alphabetical order.

Python

1. Write a Python program to perform addition, subtraction, multiplication, division, integer division and modulo division on two integer numbers.
2. A company decides to give bonus to all its employees on Diwali. A 5% bonus on salary is given to the male workers and 10% bonus on salary to the female workers. Write a Python program to enter the salary and sex of the employee. If the salary of the employee is less than Rs. 10,000/- then the employee gets an extra 2% bonus on salary. Calculate the bonus that has to be given to the employee and display the salary that the employee will get (using if-else).
3. Write a Python program to calculate the sum of numbers from m to n. (using while).
4. Write a Python program to print the multiplication table of n, where n is entered by the user (using for loop).
5. Write a Python program to swap two numbers (using function).
6. Write a Python program that encrypts a message by adding a key value to every character (using chr() and ord()).
7. Write a Python program to extract an email address from a text (using search()).
8. Write a Python program that to store the name and marks of students. Use list to store the marks in three subjects (using classes and objects).
9. Write a Python program to demonstrate the use of hierarchical inheritance.
10. Write a Python program that prompts the user to enter a number and print its square. If no number is entered (ctrl + C is pressed), then a keyboard Interrupt is generated (Exception Handling).
11. Write a Python program that reads a file line by line. Each line read from the file is copied to another file with line numbers specified at the beginning of the line.
12. Write a Python program that generates a Quiz and uses two files – Questions.txt and Answers.txt. The program opens Questions.txt and reads a question and displays the question with options on the screen. The program then opens the Answers.txt file and displays the correct answers.

Course Outcomes

Upon completion of this course, the students will be able to:

1. Achieving proficiency in writing VB.NET programs that incorporate various programming concepts and syntax.
2. Developing skills in creating interactive user interfaces using VB.NET.
3. Developing the ability to write and execute Python programs to solve simple problems.
4. Understanding and utilizing basic programming constructs, data structures, and algorithms in Python.
5. Developing problem-solving and analytical skills using Python programming language..

Reference Books

1. Reema Thareja, Python Programming using Problem Solving Approach, Oxford University Press, 2017.
2. P. RadhaGanesan, VB.Net, Scitech Publications Pvt. Ltd., 2017.
3. Jeffrey R. Shapiro, The Complete Reference Visual Basic .Net, Mc Graw Hill Education, New Delhi, 2016.

Course	BCA	Part III	Major Elective – III (a)	Semester	VI
Major	CA	Image Editing Tool		Credit	3
				Hours	5

Course Objectives

1. To introduce students to the image editing interface and file management functions for opening, creating, saving, reverting and closing files using various tools and palettes.
2. To equip students with the skills to manage workspaces and palettes in Adobe Photoshop and perform image editing tasks, including colour adjustments using various palettes.
3. to teach students how to effectively use selection tools, make selections based on colour range and modify them using various commands for accurate and efficient image editing.
4. To teach students how to effectively use image editing tool's drawing and painting tools, work with colours and create precise and professional images using shape creation, colour filling and brushwork.
5. To teach students how to efficiently work with layers and flatten images for better management and optimization.

UNIT I

Toolbar – Menu bar – Options bar – Document Window – Toolbox – Tabbed Palettes – Opening an Existing File – Creating New Document – Saving Files – Reverting Files – Closing Files – Quitting Photoshop.

UNIT II

Selecting Workspace – Managing Palettes in Photoshop Window – Saving Workspace – Deleting Workspace – Editing images – Making Colour Adjustments: Levels Palette – Curves Palette – Brightness / Contrast Palette – Hue / Saturation Palette – Histogram Palette.

UNIT III

Selection Tools: Marquee – Lasso – Magic Wand – Making Selection based on Colour Range – Modifying Selection: Adding – Subtracting – Commands: Expand – Contrast – Grow – Similar. Refine Edges – Inverse – Transforming Selection: Scaling – Rotating – Distorting – Skewing – Changing Appearance – Warping – Saving and Loading Selection.

UNIT IV

Foreground and Background Colours – Drawing Tools: Ellipse – Custom Shape – Pen. Painting Tools: Gradient – Paint Bucket – Brush – Pencil Colour Replacement. Retouching Tools: Spot Healing Brush – Patch – Eraser – Blur – Sharpen – Smudge – Dodge – Burn – Sponge.

UNIT V

Creating New Layers – Hiding and Showing Layers – Deleting Layers – Relocating Layers – Merging Layers – Flattening Images – Linking and Unlinking Layers – Masking Layers – Type Masking – Shape Masking.

Text Book:

1. Photoshop CS4 in Simple Steps, Kogent Learning Solutions Inc. Dreamtech Press, 2010.

Reference Book:

1. Photoshop CS2 in Simple Steps, Kogent Learning Solutions Inc. Dreamtech Press, 2006.

Course Outcomes

Upon completion of this course, the students will be able to:

1. Demonstrate proficiency in utilizing interface components for file management and editing.

2. Exhibit competence in managing workspaces, palettes and performing image editing tasks.
3. Master selection tools, make accurate selections and modify them efficiently for image editing.
4. Effectively work with drawing and painting tools, colours and create professional images.
5. Demonstrate proficiency in managing layers, merging, deleting and flattening images for optimization.

Text Book

1. Photoshop CS4 in Simple Steps, Kogent Learning Solutions Inc. Dreamtech Press, 2010.

Reference Book

1. Photoshop CS2 in Simple Steps, Kogent Learning Solutions Inc. Dreamtech Press, 2006.

Course	BCA	Part III	Major Elective – III (b)	Semester	VI
Major	CA	Data Mining and Warehousing		Credit	3
				Hours	5

Course Objectives

1. To provide the knowledge on Data Mining and Warehousing concepts and techniques
2. To study the basic concepts of Data Mining, Architecture and Comparison.
3. To study a set of Mining Association Rules, Data Warehouses.
4. To study about Classification and Prediction, Classifier Accuracy
5. To study the basic concepts of cluster analysis, Cluster Methods

UNIT I

Introduction: Data mining – Functionalities – Classification – Introduction to Data Warehousing – Data Pre-processing: Pre-processing the Data – Data cleaning – Data Integration and Transformation – Data Reduction.

UNIT II

Data Mining – Primitives – Languages and System Architecture: Data Mining – Primitives – Data Mining Query Language – Architecture of Data Mining Systems. Concept Description – Characterization and Comparison: Concept Description – Data Generalization and Summarization – Analytical Characterization – Mining Class Comparison – Statistical Measures.

UNIT III

Mining Association Rules: Basic Concepts – Single Dimensional Boolean Association Rules from Transaction Databases – Multilevel Association Rules from transaction databases – Multi dimension Association Rules from Relational Database and Data Warehouses.

UNIT IV

Classification and Prediction: Introduction – Issues – Decision Tree Induction – Bayesian Classification – Classification of Back Propagation. Classification based on Concepts from Association Rule Mining – Other Methods. Prediction – Introduction – Classifier Accuracy.

UNIT V

Cluster Analysis: Introduction – Types of Data in Cluster Analysis – Partitioning Methods – Hierarchical Methods – Density Based Methods – GRID Based Method – Model based Clustering Method.

Course Outcomes

On completion of this course, students will;

1. To understand the basic concepts and the functionality of the various data mining and data warehousing component
2. To know the concepts of Data mining system architectures
3. To analyse the principles of association rules
4. To get analytical idea on Classification and prediction methods
5. To Gain knowledge on Cluster analysis and its methods.

Text Book

1. Han and M. Kamber, “Data Mining Concepts and Techniques”, 2001, Harcourt India Pvt. Ltd, New Delhi.

References Books

1. K.P. Soman, ShyamDiwakar, V. Ajay “Insight into Data Mining Theory and Practice “, Prentice Hall of India Pvt. Ltd, New Delhi
2. Parteek Bhatia, ‘Data Mining and Data Warehousing: Principles and Practical Techniques’, Cambridge University Press, 2019.

Course	BCA	Extra / Additional
Major	CA	Computer Hardware and Troubleshooting (Optional)

Course Objectives

1. Identify and describe the key components of computer hardware, including desktops, servers, laptops and tablets and understand their differences in terms of features, capabilities and intended use.
2. Understand the function and operation of motherboards, including their components, form factor, processor support and interfaces for external devices such as audio and LAN systems.
3. Explain the workings of processors, memory and storage devices and understand how they work together to execute tasks and store data in a computer system.
4. Develop the skills to assemble and configure a computer system from scratch, including setting up the cabinet, installing the power supply, CPU, memory and hard disk and connecting the motherboard to power and I/O devices.
5. Learn how to troubleshoot and maintain a computer system, including identifying and resolving issues with BIOS configuration, power-on self-test, devices and drivers and common hardware components such as motherboards and processors

UNIT I

Introduction to Computer Hardware: Basic Computer Hardware Structure – Features of Computer Systems: Desktop – Server – Laptop – Tablet. Front Panel View – Rear Panel View – Internal Arrangements. Switched Mode Power Supply (SMPS): Features – Types.

UNIT II

Mother Board: Features – Components – Form Factor – Processor Support – Mother Board Controller – Bios IDE and SATA Connectors – Power Supply Connectors – External Devices Interfaces – Audio and LAN System.

UNIT III

Processing Unit: Processor Features – Stages of CPU – Multiple Core Processors – Processor Specifications – Installing and Uninstalling CPU. Memory and Storage: Working of Computer Memory – Working and Features of Hard Disks.

UNIT IV

Assembling and Configuring Computers: Setting up the Cabinet – Installing Power Supply Unit – Installing CPU – Installing Heat Sink and Cooling Fan – Installing Memory Module – Mounting Mother Board – Installing Hard Disk – Connecting Mother Board Power Supply Cables – Connecting Front Panel – Connecting I/O Devices – Configuring BIOS.

UNIT V

Troubleshooting and Maintenance: Configure BIOS Parameters – Power on Self Test (POST) – Devices and Drivers – Preventive Maintenance – Replacing CMOS Battery – Clearing BIOS Password – Motherboard Troubleshooting – Processor Troubleshooting.

Course Outcomes

Upon completion of this course, the students will be able to:

1. Demonstrate an understanding of computer hardware components, their features and their intended use for different types of computer systems, including desktops, servers, laptops and tablets.

2. Analyze the components and features of motherboards, including form factor, processor support, BIOS, connectors and interfaces for external devices such as audio and LAN systems.
3. Apply knowledge of processors, memory and storage devices to analyze and compare their features, specifications and performance in different computer systems.
4. Develop the practical skills necessary to assemble and configure a computer system from scratch, including setting up the cabinet, installing the power supply, CPU, memory, hard disk and connecting I/O devices and front panel.
5. Develop the practical skills necessary to troubleshoot and maintain a computer system, including configuring BIOS parameters, performing power-on self-test, identifying and resolving issues with devices and drivers, replacing CMOS battery and troubleshooting common hardware components such as motherboards and processors.

Text Book

1. K. L. James, Computer Hardware Installation, Interfacing, Troubleshooting and Maintenance, PHI Learning Private Limited, 2013.

Reference Book

1. Manohar Lotia, Pradeep Nair, Payal Lotia, Modern Computer Hardware Course, BPB Publications, Second Revised Edition, 2006.