KALIABOR COLLEGE (AUTONOMOUS)

DEPARTMENT OF COMPUTER SCIENCE

SYLLABUS

(w.e.f. ACADEMIC SESSION 2026-27 vide Board of Studies Resolution No. KC-AT-COMP/BoS/26/01

Dated:)

Learning Outcome Based Curriculum Framework (LOCF)

For

BCA Programme

Under NEP 2020 Guidelines

KALIABOR COLLEGE (AUTONOMOUS)

Nagaon-782137, Assam, India

BCA Syllabus for 1st and 2nd Semester

1st semester paper details :- 1) Introduction to C Programming

2) Mathematics I

3) SEC -1 :- Computer Fundamentals and Application Software

4) AEC

5) MDC

6) VAC

2nd Semester paper details :- 1) Digital Logic

2) Data Structure & Algorithm

3) SEC -2 :- Introduction to Web Technologies

4) AEC

5) MDC

6) VAC

Introduction to C-Programming

Paper Code:

1. Prerequisites: NIL
2. Semester: 1
3. Course type: Compulsory
4. Course level: 100-199
5. Theory credit: 3
6. Practical credit: 1
7. Number of required hours:
8. Theory: 45 hrs. (45 classes)
9. Practical: 30 hrs. (15 classes)

Course Outcomes:

- Understand the fundamentals of a computer system and its components such as Input Unit, Output Unit, Storage Unit, and CPU.
- Understand different techniques of problem solving and implementation of those techniques using programming.
- Be familiar with basic features of C programming and its runtime environment.
- Understand and analyse programming constructs such as conditional statements, loops, functions, recursion, pointers, etc.
- Apply programming constructs to solve practical problems.

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

- Identify and describe the basic components of a computer system (CPU, memory, I/O units) and explain number systems and their role in computing.
- Design algorithms, flowcharts, and pseudo-code to model solutions for simple computational problems.
- Write and execute syntactically correct C programs using data types, operators, expressions, control structures, and preprocessor directives.
- Implement programs using control structures (conditionals and loops), arrays, and strings to solve structured programming problems.
- Develop modular C programs using user-defined functions, recursion, pointers, structures, unions, and dynamic memory allocation.

Suggested Readings:

Text Books:

i. H. M. Deitel, P. J. Deitel, C How to program, Pearson Education India, (8th Edition),
2018.

ii. B. W. Kernighan, D. M. Ritchie, The 'C' programming language, Pearson Education
India, (2nd Edition), 2015.

Reference Books:

i. R. G. Dromey, How to Solve It by Computer, Pearson Education India, (2nd Edition), 2008.
ii. Byron S. Gottfried, Programming in C, Tata McGraw Hill, (4th Edition), 2018.
iii. Schildt Herbert, C: the Complete Reference, Tata McGraw Hill, (4th edition), 2017.

Detailed Contents:

Unit 1: Introduction to Computer and Basics of C Programming Concepts

Basic concepts of computer and its organizations, CPU, Memory Unit, Memory Hierarchy, I/O units, Number System Representation. Problem solving approaches: Algorithm, Flowcharts, Pseudo codes, Programs, Decision Table, Bottom-up and Top-down design strategies, Design and Analysis of Algorithms. The execution process, syntax and semantic errors, instructions and values, concepts of object code, source code, executable code, operating systems, assembler, compiler, interpreters, linkers, loaders. Character set, Identifiers, Keywords, Data Types, Type Casting, Constant and Variables, Storage Class, Statements, Expressions, Operators, Precedence of operators, pre-processor directives, header files, macro, standard C library functions; Bitwise operators, Input-Output Assignments, Control structures.

Unit 2: Control Structures and Arrays

Control Structures in C. Basic programming constructs: Sequence, selection and iteration. Conditional statements – if, else, switch case. Nested conditions. Loops – for loop, while loop, do while loop. Using loop for counting iterations. Using while loop for indefinite iterations. Nested loops. Break and continue statements. Array: representation of array – one dimensional, two dimensional and multidimensional; String manipulation; passing array elements.

Unit 3: Functions, User Defined Data Types and Pointers

User defined and standard functions, Formal and Actual arguments, Functions category, function prototypes, parameter passing, Call-by-value, Call-by-reference, Recursion. Declaration and use of structure, member resolution operator and structure pointer, arrays of structures. Declaration of unions, pointer to structure & unions.

Pointer: Pointer variable and its importance, Declaring and dereferencing pointer variables. Pointer Arithmetic: Accessing arrays through pointers. Pointer types, Dynamic Memory allocation.

Practical: An indicative list of experiments is given below:

- Implementation of basic features of C: primitive data types, type casting, operators and their precedence, macro handling.
- Implementation of basic programming concepts such as Control structures, Decision making and branching.
- Decision making & looping.
- Implementation of modular programming (function and recursion), arrays.
- Implementation of pointers.
- Implementation of user defined data types (structure and unions).

Mathematics I

Paper Code:

1. Course Outcome: After completing this course, students will be able to understand and apply the concepts of sets, relations, functions, matrices, determinants, statistics and probability in problem-solving and computer science applications.

2. Learning Outcomes: After successful completion of this course, students will be able to:

(a) Learn the concepts of set, relation, and function from Computer Science point of view.

(b) Learn what a matrix is and relate it with arrays used in programming.

(c) Understand determinants and how determinants are used in solving simultaneous equations.

(d) Get familiar with statistical and probabilistic measures that are used in computation related software/packages.

3. Semester – 1

4. Course Type – Compulsory

5. Course level (CL) – 100-199

6. Theory Credit – 4

7. Practical Credit -0

8. Number of required hours:

i) Theory – 60 Hours (60 Classes)

ii) Practical – NIL

9. Reference books:

(a) Discrete Mathematics Structures with Applications to Computer Science, J. P. Tremblay and R. Manohar, Mc-Graw Hill.

(b) Discrete Mathematics, N. Ch.SN Iyengar, K.A. Venkatesh, V. M. Chandrasekaran, P. S. Arunachalam, Vikash Publishing House Pvt Ltd.

(c) Elements of Discrete Mathematics, C. L. Liu, Mc-Graw Hill International Ed.

10. Course Details:

THEORY

UNIT I: Sets, Relations and Functions (15 Lectures)

Definition of sets, their types and cardinality of sets. Basic set operations and their laws, Visual representation using Venn diagrams, Principle of inclusion and exclusion and their applications on simple problems. Definition and types of relations, properties of relations, closures of relations, equivalence relations, equivalence classes and partitions. Definition and classification of functions, Basic idea of Composite and Inverse Functions, principles of mathematical induction.

UNIT II: Matrices (15 Lectures)

Definition and classification of matrices, Matrix Operations: addition, subtraction, multiplication, Row and Column operations, Properties of matrix operations, identity and inverse of matrices. Representing relations using matrices. Transpose of a matrix and its properties. Symmetric and skew symmetric matrices, Elementary transformation of a matrix and their uses, Concept of Invertible matrices.

UNIT III: Determinants (15 Lectures)

Determinant of a square matrix, minors and cofactors, Adjoint and inverse of matrices. Inverse of a matrix using elementary transformation. Rank of a matrix and its computation. Eigen values and Eigen vectors of a matrix, Cayley-Hamilton theorem, Cramer's rule.

UNIT IV: Fundamentals of Statistics and Discrete Probability (15 Lectures)

Types of Data and variables, Frequency distribution and Cumulative frequency. Tabular and Graphical Representation of frequencies: Histogram, Frequency Polygon, Frequency Curve and Cumulative Frequency curves (Ogive curves). Diagrammatic representations: Simple bar, Subdivided bar, Pie diagrams. Measures of central tendency-Mean, Median and Mode.

Measures of variation – Range, Interquartile range, Standard Deviation and Variance. Basic probability concepts, Sample space and events, random variables, Conditional Probability and Bayes theorem.

SEC -1 Computer Fundamentals

Paper Code:

1. **Course Objective (CO) – This** subject aims to help students to understand the basic concept of computers, including what a computer is , its different parts and how it works. It also teaches students about hardware and software by helping them identify different computer devices and programs used in a computer system. Student learn to use computer application such as MS Word, PowerPoint and Excel etc. In addition to the course provide knowledge about using the internet and email for communication and information sharing. Students also gain basic skills in designing simple webpages using HTML.
2. **Learning Outcome (LO)** – a) Understanding the basic concept and components of a computer system.
   b) Identifying and understanding the function of hardware, software and computer system.
   c) Creating simple documents, presentation, and files using computer application
   d) Applying internet and email services for communication and information sharing and to design the simple webpages using HTML.
3. Prerequisites -NIL
4. Semester – 1
5. Course Type – Compulsory
6. Course level (CL) – 100-199
7. Theory Credit – 3
8. Practical Credit -1
9. Number of required hour-
   iii) Theory – 45 Hour (45 Classes)
   iv) Practical – 30 Hour (30 Classes)
10. **Contents of Syllabus**
    A. **THEORY**

    UNIT 1 -: Introduction to Computers and Information Technology. (11 Lectures)

    Brief history of development of computers, computer system concepts, capabilities and limitations, Types of Computers- Analog, Hybrid, general, special purpose, Micro, mini, mainframe, super computer, generations of computers, types of personal computer, data representation in a computer, Basic of MS – Office: -MS Word, MS-PowerPoint, MS Excel.

    UNIT 2-: Memory and Storage Devices (11 Lectures)

Definition of Memory, memory hierarchy, registers, general purpose and special purpose registers, types of memory, primary and secondary memory, volatile and non – volatile memory, semiconductor memory, types of RAM, magnetic storage devices, optical storage devices, solid state devices, flash memory.

UNIT 3 -: Input Devices and Output Devices (10 Lectures)

Input devices definition, keyboard, keyboard layouts, pointing devices, mechanical and optical mouse, scanner, hand - held and flat – bed scanners, OMR, OCR, MICR, digital camera, touch pad, track ball, joystick, digital microphone. Output Device, Monitor, LCD, LED, plasma monitor, printers, impact and non - impact printers, its types, plotters, voice output systems, projector.

UNIT 4 -: Software, Computer Network (13 Lectures)

Software, types of Software, Application Vs System Software, Example of Application Software and example of System Software, Operating System, Open-Source software, Proprietary Vs Open -Source Software, Examples of Proprietary and Open -Source Software.

Computer Network, network topologies, LAN and WAN, internet, ISP, services over internet, www, webserver, web browser, HTML, HTML Tags- headings, table, list etc., introduction to CSS, domain name, URL, DNS, E-mail ,telnet, FTP.

11. List of Reference books

(a) E Balagurusamy, Fundamentals of Computers, Mc Graw Hill Education
(b) V. Rajaraman, Neeharika Adabala, Fundamentals of Computers, PHI Learning.
(c) Anita Goel, Computer Fundamentals, Pearson Education.

B. LIST OF PRACTICALS (30 Hour)

1. Using a word processing software such as Libre Office Writer / MS Word.
2. Using a spread sheet software such as Libre Office Calc / MS Excel
3. Using a presentation software such as Libre Office Impress / MS Powerpoint
4. Designing HTML webpages.

Digital Logic Fundamentals

Paper Code:

1. **Course Objective (CO) – a)** This course aims to understand students with basic concept of digital systems and binary numbers. It introduces different logic gates and their operations, along with the design and simplification of Boolean expression using Boolean algebra and K- map. Students also study combinational logic circuits such as adders, multiplexers, encoder and decoder as well as sequential circuits like flip-flop, counters and registers. This course further explains the working of memory devices and other digital components.
2. **Learning Outcome (LO) –** a) After completing this course, students will be able to grasp the fundamental concept of digital logic that will provide students the base to understand the concept of computer architecture and organization.
   b) Able to design simple digital circuit for practical application.
   c) Improve their problem-solving skills related to digital electronics and logic design.
3. Prerequisites - NIL
4. Semester – 2
5. Course Type – Compulsory
6. Course level (CL) – 100-199
7. Theory Credit – 4
8. Number of required hour-
   v) Theory – 60 Hour (60 Classes)
9. **Contents of Syllabus**

   A. Theory

   UNIT 1: Introduction to Binary Number System

   Binary Number System and their conversions: Binary, decimal, octal and hexadecimal, positive and negative numbers, gray code, ASCII, EBCDIC, binary logic, logic gates: AND, OR, inverter, buffer, NAND, NOR, XOR and equivalence.

   UNIT 2: Arithematic operations and Character Code

   Addition, Subtraction, Multiplication, Division of numbers, 1's Complement and 2's Complement of Binary Numbers, Subtraction by using 1's complement and 2's complement, binary codes: BCD, Excess-3, error detection code: parity bit, error correction code: Hamming code.

   UNIT 3: Logic Design

   Boolean algebra & Switching function, Minimization and realization using logic gates, Representation of logic functions – SOP and POS form, K-map presentation: - 2 Variable, 3-variable, 4-Variable etc.

UNIT 4: Combinational Circuits and Sequential Circuits

Definition of combinational circuit, design procedure: Adders, subtractor, multiplexers, demultiplexers, decoders, encoders full subtractor, ROM, Sequential logic:- Latch, Flipflops, state diagram, Mealy and Moore machine, flip flop excitation tables, design procedure of clocked sequential circuit, 3-bit binary counter, shift register, ripple counter, RAM.

10. List of Reference Books:

(a) M. Morris Mano, Digital Logic and Computer Design, Pearson India.

(b) V. Rajaraman, T. Radhakrishnan, Digital Logic and Computer Organization, PHI Learning

Data Structures & Algorithms Using C

Paper Code:

1. Course Outcomes (CO):

- Explain the concept of different data types, primitive, derived and their representation and application through coding.
- Apply the concept of ADT, linear and nonlinear data types, their representation and apply these data types in various applications like arithmetic expression evaluation and conversion.
- Explain the concept of Graphs and Trees and their real-time application.
- Develop various searching, sorting and hashing techniques.
- Choose and implement efficient data structures and apply them to solve problems.

2. Learning Outcomes (LO): On successful completion of the course, students will be able to:

- Recall and describe fundamental data types (primitive and derived), data structures, and Abstract Data Types (ADT) and implement them using C.
- Implement and apply linear data structures (arrays, linked lists, stacks, queues) to solve problems such as arithmetic expression evaluation and conversion.
- Construct and traverse non-linear data structures (Trees and Graphs) including BST, AVL, Heap, B-Tree, DFS and BFS, and relate them to real-world applications.
- Design and implement searching, sorting (linear, binary, bubble, merge, heap, quick, radix), and hashing techniques and evaluate their time and space complexity.
- Select and justify the most appropriate data structure for a given computational problem and implement a solution using C programming.

3. No. of Credits: 4 (L+P=3+1)

4. Prerequisites - NIL

5. Semester – 2

6. Course Type – Compulsory

7. Course level (CL) – 100-199

8. Number of required hour-

a) Theory: 45 hrs. (45 classes)
b) Practical: 30 hrs. (15 classes)

Suggested Readings: Textbooks: 1. S. Lipschutz: Data Structures, McGraw Hill Education; (Revised 1st Edition), 2014.

Recommended Books: 1. M.T. Goodrich, R. Tamassia and D. Mount: Data Structures and Algorithms in C++, John Wiley & Sons, Inc, (2nd Edition), 2011.
2. Aho, Hopcroft and Ullman: Data Structures and Algorithms, Addison Wesley Publishing Co, (1st Edition), 1982.

Detailed Contents:

UNIT 1: ADT Overview, Array and Linked List: Data Definition, Data Object, Data Types, Built-in Data Type, Derived Data Type, Data Structure and Implementation, Abstract data types. Array: Data Structure, insertion and deletion in an array, Storage Representation of Arrays, Applications of Arrays, Polynomial Representation Using Arrays, insertion and deletion operations in an array. Searching and Sorting in Array. Linked List: Insertion, deletion, searching in Singly (sorted/unsorted) Linked List, circular and doubly Linked List.

UNIT 2: Stack, Queue and Trees: Stack: Representation using array & linked list; operations - push, pop, Applications of stack: recursion. Conversion from infix: to postfix, evaluation of postfix expression. Queue: Representation using array & linked list; operation - insert, delete; circular queue, dequeue, priority queue implementation, Applications of Queue. Trees: Terminologies, traversal algorithms (preorder, post order, in order). Implementation of binary search tree, threaded tree (one way & two way), AVL tree balancing, Heap tree, B tree -introduction, operation - insertion, deletion, Introduction to red black tree.

UNIT 3: Graph and Hashing: Introduction, Graph Representation, Adjacency Matrix, Adjacency List, 6 Graph Traversals, Depth First Search, Breadth First Search, Spanning trees. Hashing Types and Applications, collision resolution: open addressing and chaining.

UNIT 4: Searching, Sorting and analysis of Time and Space Complexity: Linear search, Binary search, Insertion sort, selection sort, Bubble sort, Merge sort, Heap sort, Quick sort, Radix sort, Complexity analysis of sorting and searching algorithms.) Complexity measures of an algorithm – Time and space complexity. Average case and worst-case analysis. Asymptotic notation as a measure of algorithm complexity, O and θ notations. Analysis of sorting algorithms and Searching algorithms in terms of time and space complexity in best, average and worst case.

Practical: An indicative list of experiments is given below:
- Program to insert a set of elements in an array.
- Program to insert elements in a sorted array.
- Program to delete an element in an array.
- Sorting a list of numbers in ascending order.
- Searching an element using linear search.

- Searching an element using binary search.
- Program to create and display a linked list.
- Insert data at the beginning of a linked list.
- Insert data at the end of a linked list.
- Insert data in a sorted linked list.
- Insert an element at a specific position in a linked list.
- Search an element in a two-dimensional array.
- Implement Stacks and Stack application, Queues
- Implement all traversals in a Binary tree.
- Create, insert and delete in a Binary search tree.

SEC 2: WEB TECHNOLOGIES

Paper Code:

1. Course Outcome: This course provides basic knowledge of web technologies and web application development. Students learn to create web pages using HTML and CSS, and interactively using JavaScript, and understand server-side scripting with PHP for developing simple web applications.

2. Learning Outcome: At the end of the course, students will be able to:

a) Understand the basic concept of web applications and web services.

b) Design basic well-structured web page using HTML and CSS.

c) Develop the ability to implement interactive elements and dynamic content using basic JavaScript.

d) Develop a foundational understanding of server-side scripting using PHP.

3. Semester: 2

4. Course Type: Compulsory

5. Course level (CL): 100-199

6. Theory Credit: 3

7. Practical Credit: 1

8. Number of required hours:

i) Theory – 45 Hours

ii) Practical – 35 Hours

9. Reference books:

a) Jackson J.C. (2007). Web Technologies: A Computer Science Perspective. Pearson.

b) Duckett, J. (2011). HTML and CSS: Design and Build Websites. John Wiley & Sons.

c) Robbins, J. N. (2018). A Beginner's Guide to HTML, CSS, JavaScript, and Web Graphics. O'Reilly Media.

10. Course Details:

A. Theory

UNIT 1: Introduction to Web Technologies

Basic Concepts of Web, Overview of web browsers and web servers, Client-Server Architecture, Communication Protocols, Basics of DNS, Introduction to ports, URLs, caching and cookies, Overview of data privacy and protection laws (including GDPR), web hosting services and control panels.

UNIT 2: Front End Development using HTML

Introduction to websites and webpages, Markup languages and structure of HTML documents, Overview of HTML elements and document layout, Text formatting in HTML: headings, paragraph, line break, horizontal lines. Creating hyperlinks and navigation using anchor elements. Working with lists: ordered, unordered, descriptive lists. Adding images and multimedia content to web pages. Tables in HTML. Forms in HTML. Semantic HTML for meaningful page structure. Sectioning elements in HTML.

UNIT 3: Front End Design using CSS

Introduction to CSS and its role in web design, CSS syntax and rule structure, Inline, Internal and External CSS, CSS selectors, CSS combinators, CSS Box Model, CSS colors and background properties, Typography in CSS.

UNIT 4: Client-Side Scripting with JavaScript

Introduction to JavaScript, JS Syntax and Datatypes, JS variables and their scope, JS operators, Control Structures in JS (Conditional and Looping mechanisms), Functions in JS, Introduction to DOM, DOM manipulation techniques, Selecting and Accessing HTML elements, Inner Text and Inner HTML, JS Events and Event Handling, Client-side form validation using JavaScript.

UNIT 5: Server-Side Programming with PHP

Introduction to PHP, PHP syntax and variables, Predefined PHP functions, Handling HTML forms using PHP, Form submission methods – GET and POST, Processing and validating user input on the server-side, Introduction to the tech-stack (Frontend, Backend, Database, Infrastructure), Establishing database connections using PHP, performing database operations: inserting, fetching, updating and deleting records.

B. List of Practical

1. Create a basic HTML webpage structure with a heading, paragraph, and an image.

2. Build a navigation menu using an unordered list (<ul>) with clickable links.

3. Design an image gallery using HTML and CSS with proper padding and border.

4. Embed a YouTube video on a webpage using the <iframe> tag.

5. Create a dropdown select menu (<select>) with multiple options.

6. Use HTML5 semantic tags (such as <header>, <nav>, <section>, <article>, <footer>) to structure and organize content on a webpage.

7. Build a registration form with fields for name, email, password, date of birth, address and other such fields with a submit button. Include appropriate input types, labels and placeholders.

8. Build a simple JavaScript calculator that can perform basic arithmetic operations.

9. Implement a function that changes the inner Text and inner HTML of a paragraph on button clicks.

10. Build a form with input fields for email, password and confirm password. When the form is submitted, use an alert to display a success message if the password and confirm password values matches, otherwise show an error alert. Use JavaScript for the validation.

11. Create a list of items. Add a click event listener to each item so that when clicked, the background color of the clicked item changes.

12. Write a PHP script to display the current date and time on a webpage.

13. Write a PHP script to connect to a MySQL database and fetch data from a table.

COMMON PAPER SYLLABUS DETAILS

Papers :- 1) SEC :- ITB(Information Technology to Business)

2) VAC (Artificial Intelligence)

COURSE NAME – Information Technology in Business (SEC)

PAPER CODE –

PAPER CREDIT – 3 Credit

SEMESTER – 1st

THEORY

UNIT 1-: Fundamentals of Computer and Information Technology

Definition of Computer, different types of computer system, I/O devices, storage device & its units of measurement, other peripheral devices, CPU its functions, communication among various parts of a computer system, data, information, role of IT in information technology, management and decision technology.

UNIT 2-: Introduction to the Computer Networks

Fundamentals of Computer Networks and internet, WAN, LAN, MAN, PAN, wireless technologies, information ethics and security, hackers, computer viruses, www, search engines and their business prospective.

UNIT 3-: Introduction to MS Office

MS Office and its tools, working with MS Word, creating, editing, formatting and printing document, working with MS – Excel, data sorting and formulas, graph creation. MS - Powerpoint -creating simple and animated presentation.

PRACTICALS

1. Using word processing software MS -Word.
2. Preparing a presentation using MS-Powerpoint.
3. Basic calculation in MS - Excel.

Introduction to Artificial Intelligence
VAC

No. of Credits: 2
Level: 100-199
Number of required hours: 30 hrs.

Course Description

This course introduces students to the basic ideas of Artificial Intelligence (AI) and how it is used in everyday life. It is designed for beginners and covers simple concepts of AI, Machine Learning, Deep Learning, and Natural Language Processing in an easy-to-understand way, without focusing on complex mathematics or technical details.

Course Objectives

a) To help students understand the basic concepts and scope of Artificial Intelligence.
b) To build a simple understanding of Machine Learning and Deep Learning.
c) To introduce students to how AI works with human language through Natural Language Processing.
d) To make students aware of real-world applications of AI and the ethical issues related to it.

Learning Outcomes Upon successful completion of the course, students will be able to:

a) Explain the basic concepts and evolution of Artificial Intelligence
b) Differentiate between AI, Machine Learning, and Deep Learning
c) Describe various types of Machine Learning approaches
d) Understand the role of AI in processing human language
e) Identify real-world applications and implications of AI technologies

Suggested Readings and Resources

- Artificial Intelligence: A Modern Approach – Stuart Russell & Peter Norvig
- Artificial Intelligence Basics: A Non-Technical Introduction – Tom Taulli
- Machine Learning for Absolute Beginners – Oliver Theobald
- IBM AI Learning Resources – Beginner-friendly explanations
- Microsoft Learn – Basic AI concepts and tutorials
- Coursera – Introductory AI courses

Course Structure

Unit I: Foundations of Artificial Intelligence (10 Hours)

Definition, scope, and objectives of Artificial Intelligence, historical development and evolution of AI, characteristics and goals of AI systems, types of AI: Narrow AI, General AI, and Super AI, comparison between Human Intelligence and Artificial Intelligence, applications of AI in various sectors (Healthcare, Education, Finance, etc).

Unit II: Fundamentals of Machine Learning and Deep Learning (10 Hours)

Introduction to Machine Learning and its significance, relationship between AI, Machine Learning, and Deep Learning, types of Machine Learning: Supervised Learning, Unsupervised Learning and Semi-supervised, Basic concepts: data, features, training, and models. introduction to Deep Learning Concept of Artificial Neural Networks (ANN), introduction to Convolutional Neural Networks (CNN) and their applications.

Unit III: Natural Language Processing and AI Applications (10 Hours)

Introduction to Natural Language Processing (NLP), Applications of NLP: chatbots, language translation, sentiment analysis, Basic NLP processes: tokenization and text analysis, Introduction to Generative AI, Case studies: recommendation systems, virtual assistants, and chatbots, Emerging trends and future scope of AI), ethical issues in gen ai (bias, privacy, and societal impact)