VIDYASAGAR UNIVERSITY

Midnapore, West Bengal

PROPOSED CURRICULUM & SYLLABUS (DRAFT) OF

BACHELOR OF COMPUTER APPLICATION- BCA (HONOURS)

4-YEAR UNDERGRADUATE PROGRAMME

(w.e.f. Academic Year 2023-2024)

Based on

Curriculum & Credit Framework for Undergraduate Programmes (CCFUP), 2023 & NEP, 2020

VIDYASAGAR UNIVERSITY
BACHELOR OF COMPUTER APPLICATION- BCA(HONOURS)
(under CCFUP, 2023)

Level	YR.	SEM	Course Type	Course Code	Course Title	Credit	L-T-P	Marks		
								CA	ESE	TOTAL
BCA (Hons.)	2nd	III	**SEMESTER-III**							
			Major-3	**BCAHMJ03**	**T: Computer Architecture**	**4**	**3-0-1**	**15**	**60**	**75**
			Major-4	**BCAHMJ04**	**T: Discrete Mathematics**	**4**	**3-1-0**	**15**	**60**	**75**
			SEC	**BCASEC03**	**P: PYTHON (Practical)**	**3**	**0-0-3**	**10**	**40**	**50**
			AEC	**AEC03**	Communicative English -2 (***common for all programmes***)	**2**	**2-0-0**	**10**	**40**	**50**
			MDC	**MDC03**	Multidisciplinary Course -3 (***to be chosen from the list***)	**3**	**3-0-0**	**10**	**40**	**50**
			Minor-3	**BCAMIN03**	**T: Data Structure and Algorithm**	**4**	**3-0-1**	**15**	**60**	**75**
					Semester-III Total	**20**				**375**
		IV	**SEMESTER-IV**							
			Major-5	**BCAHMJ05**	**T: Database Management Systems; P: Practical**	**4**	**3-0-1**	**15**	**60**	**75**
			Major-6	**BCAHMJ06**	**T: Operating Systems; P: Practical**	**4**	**3-1-0**	**15**	**60**	**75**
			Major-7	**BCAHMJ07**	**T: Computer Network**	**4**	**3-1-0**	**15**	**60**	**75**
			AEC	**AEC04**	MIL-2 (***common for all programmes***)	**2**	**2-0-0**	**10**	**40**	**50**
			Minor-4	**BCAMIN04**	**T: Artificial Intelligence**	**4**	**3-0-1**	**15**	**60**	**75**
			Summer Intern.	**INT**	**Internship/ Apprenticeship - activities to be decided by the Colleges following the guidelines to be given later**	**4**	**0-0-4**	**-**	**-**	**50**
					Semester-IV Total	**22**				**400**
					TOTAL of YEAR-2	**42**				**775**

MJ = Major, MI = Minor Course, SEC = Skill Enhancement Course, AEC = Ability Enhancement Course, MDC = Multidisciplinary Course, CA= Continuous Assessment, ESE= End Semester Examination, T = Theory, P= Practical, L-T-P = Lecture-Tutorial-Practical, MIL = Modern Indian Language

MAJOR (MJ)

MJ-3: Computer Architecture **Credits 04**

OBJECTIVE OF THE COURSE

This course offers a deep understanding of computer system design, covering processor architecture, memory hierarchies, input/output systems, and instruction set architectures (ISA). It explores data paths, control units, pipelining, parallel processing, and multi-core architectures, highlighting advancements in computing. Students gain practical experience through labs and programming in assembly language and hardware simulation tools. The course emphasizes the impact of architectural choices on software performance, preparing students to analyze, optimize, and pursue advanced studies in computer system design and engineering.

MJ-3T: Computer Architecture **Credits 03**

Course Content:

Module I: Fundamentals of Computer Architecture **20 Hrs.**

Boolean algebra, Logic gates, combinational circuits; circuit simplification, decoders, multiplexers, sequential circuits; flip-flops and, registers, counters and memory units.

Module II: Data Representation and Arithmetic **10 Hrs.**

Number systems, complements, fixed and floating point representation, character representation, addition, subtraction, magnitude comparison, multiplication and division algorithms for integers

Module III: Basic Computer Organization and Design **8 Hrs.**

Computer registers, bus system, instruction set, timing and control, instruction cycle, memory reference, Organization of a basic single-bus computer system.

Module IV: Central Processing Unit **10 Hrs.**

Register organization, arithmetic and logical operations, Instruction formats, addressing modes, instruction codes, machine language, assembly language, RISC, CISC architectures, Hardwired and microprogrammed control unit design.

Module V: Memory Organization **6 Hrs.**

Memory interfacing and addressing, cache memory organization.

Module VI: Input-Output Organization **6 Hrs.**

Input / Output: External Devices, I/O Modules, Programmed I/O, Interrupt-Driven I/O, Direct Memory Access, I/O Channels.

Suggested Readings:

1. 2. W. Stallings, Computer Organization and Architecture Designing for Performance, 8 Edition, Prentice Hall of India,2009
2. M. Mano, Computer System Architecture, Pearson Education 1992
3. Carl Hamacher, Computer Organization, Fifth edition, McGrawHill, 2012.
4. M.M. Mano , Digital Design, Pearson Education Asia,2013

MJ-3P: Computer Architecture Lab **Credits 01**

All laboratory assignments focus on Hardware Description Language (HDL) using VHDL or Verilog for simulation or implementation with hardware kits.

[Pre-requisite: Prior experience in hardware-based design from Analog & Digital Electronics and Computer Organization laboratories.]

1. Introduction to HDL: Overview and basic syntax of VHDL/Verilog.
2. Digital Logic Programming: Implement basic digital logic circuits using HDL.
3. Arithmetic Operations: Design and simulate 8-bit addition, multiplication, and division circuits.
4. 8-bit ALU Design: Simulate an Arithmetic Logic Unit capable of performing basic operations.
5. 8-bit CPU Design: Implement a simple CPU with basic instruction set architecture.
6. 8-bit Register Design: Create a register and perform read/write operations.
7. Memory Unit Design: Design a memory unit and demonstrate basic memory operations.
8. CPU and Memory Interfacing: Simulate and test the interaction between a CPU and a memory unit.

9. Create the micro operations and associate with instructions as given in the chapter (except interrupts). Design the register set, memory and the instruction set. Use this machine for the assignments of this section.

10. Design a fetch routine for the instruction cycle.

11. Simulate the machine for the memory-reference instructions referred in above question with I= 1 and address part = 082. The instruction to be stored at address 026 in RAM. Initialize the memory word at address 082 with the value 298. Initialize the memory word at address 298 with operand B8F2 and AC with A937. Determine the contents of AC, DR, PC, AR and IR in hexadecimal after the execution.

12. Simulate the machine to determine the hexadecimal contents of the AC, E, PC, AR, and IR registers after executing each of the following register reference instructions:

a. CMA e. CIR i. SNA

b. CLE f. CIL j. SZA

c. CLA g. INC k. SZE

d. CME h. SPA l. HLT

Initialize the contents of AC to $(A937)_{16}$, that of PC to $(022)_{16}$ and E to 1.

13. Modify the machine created in Practical 1 according to the following instruction format:

Instruction format

0 2 3 4 15

Opcode I Address

a. The instruction format contains a 3-bit opcode, a 1-bit addressing mode and a 12-bit address. There are only two addressing modes, I = 0 (direct addressing) and I = 1 (indirect addressing).

b. Create a new register I of 1 bit.

c. Create two new microinstructions as follows:

i. Check the opcode of instruction to determine type of instruction (Memory

Reference/Register Reference/Input-Output) and then jump accordingly.

ii. Check the I bit to determine the addressing mode and then jump accordingly.

14. Simulate the machine for the following memory-reference instructions with I= 0 and address part = 082. The instruction to be stored at address 022 in RAM. Initialize the memory word at address 082 with the operand B8F2 and AC with A937. Determine the contents of AC, DR, PC, AR and IR in hexadecimal after the execution.

a. STA f. BSA

b. ISZ g. AND

c. LDA d. ADD

e. BUN

MJ-4: Discrete Mathematics **Credits 04(Full Marks: 75)**

OBJECTIVE OF THE COURSE

The objective of the Discrete Mathematics course is to introduce students to essential mathematical concepts and techniques used in computer science, engineering, and mathematics, focusing on logical reasoning, problem-solving, and the study of discrete structures like sets, functions, relations, combinatorics, and graph theory. The course aims to enhance students' ability to apply mathematical principles to real-world problems, develop skills in mathematical proof techniques, and establish a strong foundation for advanced topics in computer science, such as algorithms, cryptography, and network theory. By the end of the course, students will be equipped to approach complex problems with rigorous mathematical tools and techniques.

MJ-4T: Discrete Mathematics **Credits 04**

Module 1: Introduction **(12 Lectures)**

Sets - finite and Infinite sets, uncountably Infinite Sets; functions, relations, Properties of Binary Relations, Closure, Partial Ordering Relations; counting - Pigeonhole Principle, Permutation and Combination; Mathematical Induction, Principle of Inclusion and Exclusion.

Module 2: Growth of Functions **(8 Lectures)**

Asymptotic Notations, Summation formulas and properties, Bounding Summations, approximation by Integrals

Module 3: Recurrences **(8 Lectures)**

Recurrence Relations, generating functions, Linear Recurrence Relations with constant coefficients and their solution, Substitution Method, Recurrence Trees, Master Theorem

Module 4: Graph Theory **(10 Lectures)**

Basic Terminology, Models and Types, multigraphs and weighted graphs, Graph Representation, Graph Isomorphism, Connectivity, Euler and Hamiltonian Paths and Circuits, Planar Graphs, Graph Coloring, Trees, Basic Terminology and properties of Trees, Introduction to Spanning Trees

Module 5: Prepositional Logic **(12 Lectures)**

Logical Connectives, Well-formed Formulas, Tautologies, Equivalences, Inference Theory

Module 6: Matrix Algebra **(10 Lectures)**

Types of matrices, algebra of matrices–addition, subtraction, and multiplication of matrices, determinant of a matrix, symmetric and skew-symmetric matrices, orthogonal matrix, rank of a matrix,

inverse of a matrix, applications of matrices to solve system of linear equations, Eigen values and Eigen vectors, Caley-Hamilton theorem.

Suggested Readings:

1. Discrete Mathematics and Graph Theory, Satyanarayana & Prasad
2. Discrete Mathematics with Graph Theory, 3rd ed., Goodaire & Parmenter
3. Discrete Mathematics and Graph Theory, 2nd ed., Biswal
4. Discrete Mathematics, Rajendra Akerkar and Rupali Akerkar
5. Discrete Mathematics, Babu Ram

MJ-5: Database Management Systems **Credits 04(Full Marks: 75)**

OBJECTIVE OF THE COURSE

- Understanding core concepts of DBMS.
- Gaining proficiency in database design and SQL.
- Applying advanced database techniques.

OUTCOME OF THE COURSE:

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

- Understand database concepts and design principles.
- Apply SQL for data retrieval and manipulation.
- Implement transactions and concurrency control.
- Explore NoSQL databases and big data technologies.
- Develop secure and efficient database solutions.

MJ-5T: Database Management Systems **Credits 04 (45 hrs)**

Course Contents

Introduction **(12 hours)**

Definition of Data, Database, and DBMS, Overview of Database Applications, Advantages, and Disadvantages of DBMS, Roles of Database Users and Administrators Data Models: Introduction to Data Models, Types of Data Models: Hierarchical, Network, Relational, Object-Oriented, Importance of Data Models in DBMS, Database Design: Keys: Primary Key, Candidate Key, Super Key, Foreign Key, Composite Key, Alternate Key, Unique Key, Surrogate Key, Constraints: Primary Key, Foreign Key, Unique Key, NOT NULL, CHECK Entity-Relationship (ER) Model: Entities and Entity Sets, Attributes and Relationships, ER Diagrams, Key Constraints, Weak Entity Sets, Extended ER Features: Introduction to Relational Model and Relational Schema.

Relational Algebra, SQL, and Normalization (16 hours)

Relational Algebra and Calculus: Introduction to Relational Algebra, Operations: Selection, Projection, Set Operations, Join Operations, Division, Tuple and Domain Relational Calculus
Structured Query Language (SQL): SQL Basics: DDL and DML, Aggregate Functions (Min(), Max(), Sum(), Avg(), Count()), Logical Operators (AND, OR, NOT), Predicates (LIKE, BETWEEN, DISTINCT), Clauses: GROUP BY, HAVING, ORDER BY, Joins: Inner Join, Natural Join, Outer Joins (Left, Right, Full), Equi Join, Advanced SQL: Analytical Queries, Hierarchical Queries, Recursive Queries, Views, Cursors, Stored Procedures and Functions, Triggers, Dynamic SQL, Normalization and Database Design: Functional Dependencies: Armstrong's Axioms, Reflexivity, Augmentation, Transitivity, Types of Functional Dependencies: Trivial, Non-Trivial, Partial, Full, Normal Forms: 1NF, 2NF, 3NF, BCNF, Denormalization

Transaction Management and Indexing (8 hours)

Transaction Management: ACID Properties, Transactions and Schedules, Concurrent Execution of Transactions, Lock-Based Concurrency Control, Performance of Locking, Transaction Support in SQL, Crash Recovery, Two-Phase Locking (2PL), Serializability, Recoverability, Lock Management and Deadlock Resolution, Database Storage and Indexing: Data on External Storage, File Organizations and Indexing, Index Data Structures and Comparison of File Organizations, Indexes and Performance Tuning, Guidelines for Index Selection.

Advanced Databases and Security (9 hours)

NoSQL Databases and Big Data: Introduction to NoSQL Databases, Data Models: Document, Key-Value, Column Family, Graph, Features of NoSQL Databases, CAP Theorem, BASE vs. ACID, CRUD Operations in MongoDB, MongoDB Operators, Big Data Technologies: Overview of Hadoop, MongoDB, Cassandra, Database Security and Advanced Topics: Introduction to Database Security, Access Control, Discretionary Access Control, Introduction to Data Warehousing, OLAP, and Data Mining.

Suggested Readings:

Text Books

1. Raghu Ramakrishnan, Johannes Gehrke, “Database Management Systems,” Third Edition, McGraw–Hill, 2018.
2. Benjamin Rosenzweig, Elena Rakhimov, “Oracle PL/SQL by Example,” Fifth Edition, Prentice Hall, 2015.
3. Brad Dayley, “NoSQL with MongoDB in 24 Hours,” First Edition, Sams Publishing, 2024.

Reference Books

1. Korth, Silbertz, Sudarshan, “Database System Concepts,” Seventh Edition, McGraw-Hill, 2019.
2. R.P. Mahapatra, Govind Verma, “Database Management Systems,” Khanna Publishing House, 2025.

Web Resources

1. Oracle Base Articles : https://oracle-base.com/articles/articles
2. Oracle Forums: https://forums.oracle.com/ords/r/apexds/community/home
3. Ask Tom Oracle: https://asktom.oracle.com/ords/f?p=100:1:0

MJ-5P: Database Management Systems- Lab **Credits 01 (30 hrs)**

List of practical exercises:

1. Draw an ER Diagram for a Hospital Management System.
2. Convert the ER diagram in question 1 into tables.
3. Create tables for Supplier, Part, and Supplies schemas.
4. (a) Add a new attribute to the Supplier table.
   (b) Modify the datatype of the Phone attribute.
   (c) Rename the attribute City to Address in the Supplier table.
   (d) Use Truncate and Drop commands.
5. (a) Insert records into Supplier, Part, and Supplies tables.
   (b) Query data using various conditions (e.g., city starts with 'A').
   (c) Perform updates and deletions based on specific conditions.
6. Analytical queries (e.g., fifth highest earning employee), Recursive queries (e.g., hierarchy paths), Queries using operators and advanced functions.
7. Write PL/SQL programs for basic calculations.
8. Use cursors for data retrieval and aggregation.
9. Implement triggers, procedures, and functions for database operations.
10. MongoDB Queries

Suggested Readings:

1. The complete reference-By Coach and loney
2. A Beginners guide- By Abbey and corney
3. Database System-Elmasri and Navathe
4. Database system concepts- Silberschatz Abraham Korth Henry F. Jt. auth. Sudarshan S. Jt. Auth.
5. Database management system oracle SQL and PL/SQL- Das Gupta Pranab Kumar

MJ-6: Operating Systems **Credits 04(Full Marks: 75)**

OBJECTIVE OF THE COURSE

- To provide knowledge about the services rendered by operating systems
- To provide a detailed discussion of the various memory management techniques
- To discuss the various file-system design and implementation issues
- To discuss how the protection domains help to achieve security in a system

OUTCOME OF THE COURSE:

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

- Ability to comprehend the techniques used to implement the process manager
- Ability to comprehend virtual memory abstractions in operating systems
- Ability to design and develop file system interfaces, etc.
- Technical knowhow of the working principle of various types of operating systems

MJ-6T: Operating Systems **Credits 04 (45 hrs)**

Course Contents

Unit – I **(10 Lectures)**

Operating Systems –Definition – Types - Functions - Abstract view of OS - System Structures – operating systems generations – System Calls - Virtual Machines – Process Concepts – Threads – Multithreading - protection and security - distributed systems.

Unit – II **(15 Lectures)**

Process concepts – process state, – process control block – Process Scheduling - Process Co-ordination – Synchronization – Semaphores – Monitors Hardware Synchronization – Deadlocks – Methods for Handling Deadlocks

Unit – III **(15 Lectures)**

Memory Management Strategies –Contiguous and Non-Contiguous allocation –Virtual memory Management – Swapping – contiguous memory allocation – paging – structure of the page table – Demand Paging - Page Placement and Replacement Policies - thrashing – case study - UNIX.

Unit – IV **(12 Lectures)**

File System –Basic concepts - File System design and Implementation - file system mounting - file sharing - protection - Mass Storage Structure – free-space management - Disk Scheduling – Disk Management – System Protection and Security- Case Study: Linux File Systems.

Unit – V **(8 Lectures)**

I/O Management - Principles of I/O Hardware - Disk structure - Disk scheduling algorithms.

Suggested Books:

1. Silberschatz, Galvin, Gagne, "Operating System Concepts", John Wiley and Sons.
2. William Stallings, “Operating Systems –Internals and Design Principles”, 8/E, Pearson Publications.
3. Andrew S. Tanenbaum, “Modern Operating Systems”, 4/E, Pearson Publications.

MJ-7: Computer Networking **Credits 04 (Full Marks: 75)**

OBJECTIVE OF THE COURSE

The objective of the **Computer Networks** course is to provide students with a solid understanding of the principles, protocols, and technologies used in modern computer networks. The course aims to teach students how data is transmitted across different network types, including local area networks (LANs), wide area networks (WANs), and the internet. Topics such as network architecture, communication protocols, routing, switching, network security, and wireless networks will be covered. Students will learn to design, configure, and troubleshoot network systems while understanding key concepts like IP addressing, TCP/IP, and network-layer protocols. By the end of the course, students will be equipped with the knowledge and skills necessary to analyze, implement, and manage network systems effectively in both academic and real-world settings.

MJ-7: Computer Networking **[Theory 60 Hours]**

Corse Content:

Unit I : Physical Layer: Introduction to Computer Communication and Network : Network Topologies, Types of Network, OSI Model, Protocol Stack, Network Protocols. Analog Signals & Digital Signals. Data Transmission: Sampling, Transmission Mode. Analog Transmission: Modulation (Analog & Digital Signals). Multiplexing: FDM, WDM & TDM. Transmission Media: Guided Media, Unguided Media (Wireless). Circuit Switching.

Unit II : Data Link Layer: Error detection and correction: - Type of Errors, Detection, Error Correction, Framing. Data Link Control and Protocols: - Flow and Error control, CRC, REC, FEC, Hamming Code, Stop-and Wait ARQ, Go-Back, N ARQ, Selective Repeat ARQ, HDLC. ALOHA, CSMA, CSMA/CD. Multiple Access: Random Access, Controlled Access, Area Network: Ethernet, Wireless LANS: IEEE 802-11, Frame Relay, ATM

Unit-III: Network Layer: Host to Host Delivery: IP Addressing and Routing, Gateway, N/W Layer Protocols: ARP, IPV4, ICMP, IPV6, Transport Layer: Process-to-Process Delivery: UDP, TCP Congestion Control & Quality of Service.

Unit-IV: Application Layer: Client Server Model, Domain Name System (DNS), E-mail (SMTP), File Transfer (FTP) HTTP, WWW.

Reference Books:

1. Data Communication & Networking – Behuouz A. Forouzan, TMH
2. Computer Network – A.S Tanenbaum, Pearson Education
3. Computer Newtworks- kundu – PHI
4. Computer Network – Rajesh – Vikash

MINOR (MI)

Minor (MI) – 3: Data Structure and Algorithm **Credits 04 (Full Marks: 75)**

OBJECTIVE OF THE COURSE

The objective of the **Data Structures and Algorithms** course is to provide students with a comprehensive understanding of the fundamental data structures and algorithms that form the backbone of computer programming and software development. The course aims to equip students with the skills to design, analyze, and implement efficient algorithms, using appropriate data structures to solve complex computational problems. Students will learn how to evaluate algorithm performance through time and space complexity analysis and gain hands-on experience with techniques such as searching, sorting, recursion, dynamic programming, and graph algorithms. By the end of the course, students will be capable of selecting and applying the right data structures and algorithms to optimize performance and solve real-world challenges in computer science and software engineering.

MI – 3T: Data Structure and Algorithm **[Theory: 60 Hrs]**

Course contents:

1. **Analysis of Algorithms:** Characteristics of an algorithm, Steps in Designing of Algorithms, Growth of function, Recurrence, Substitution Method, Iteration Method, Asymptotic Notations, Concept of efficiency of analysis of an algorithm ,Comparative efficiencies of algorithms: Linear, Quadratic, Polynomial and Exponential, Calculation of Storage Complexity, Calculation of Run Time Complexity.

2. **Arrays:** Single and Multi-dimensional Arrays, Sparse Matrices, Row major and column major representation.

3. **Linked Lists:** Abstract Data Type-List, Singly, Doubly and Circular Lists (Array and Linked representation); Normal and Circular representation of Stack in Lists;

4. **Stacks:** Abstract Data Type-Stack, Implementation of stack using array and linked list, Prefix, Infix and Postfix expressions, Algorithmic Implementation of Multiple Stacks , Applications of stack.

5. **Queues:** Abstract Data Type-Queue, Array and Linked representation of Queue and circular Queue, De-queue, Priority Queues.

6. **Trees:** Abstract Data Type-Tree, Implementation of Tree, Tree Traversals, Binary Trees , Implementation of Binary Tree , Binary Tree Traversals(Recursive Implementation and Non Recursive Implementations),Binary Search Trees, AVL tree (Various operations on AVL Trees), Heap Tree.

7. **Searching and Sorting:** Linear Search, Selection Sort, Insertion Sort, Bubble Sort, Divide and Conquer approach(Binary Search, Quick Sort and Merge sort), Comparison of Sorting Algorithm Techniques.

8. **Hashing:** Hashing technique, Different types of hash function ,Collision resolution, chaining.

9. **Graph:** Definitions, Shortest Path Algorithms ,Dijkstra's Algorithm , Minimum cost Spanning Trees , Kruskal's Algorithm , Prims's Algorithm , Breadth First Search , Depth First Search.

10. **Greedy Technique:** Some Examples to understand Greedy Techniques, Formalization of Greedy Techniques, Knapsack problem, Huffman Codes, A task scheduling algorithm

Suggested Readings:

1. Data Structure using C – Rajni Jindal – Umesh Publication
2. Data Structure using C – B. Baluja Dhanpatrai Publication
3. Classic Data Structures, 2nd ed., Samanta
4. Data Structures Using C and C++, 2nd ed., Langsam, Augenstein & Tenenbaum

Minor (MI) – 4: Artificial Intelligence **Credits 04(Full Marks: 75)**

OBJECTIVE OF THE COURSE

The objective of the **Artificial Intelligence** course is to introduce students to the fundamental concepts, techniques, and applications of AI in solving complex problems. The course aims to provide an understanding of key AI topics such as machine learning, search algorithms, natural language processing, knowledge representation, and robotics. Students will learn how to design and implement intelligent systems that can perform tasks typically requiring human intelligence, such as reasoning, decision-making, and pattern recognition. By the end of the course, students will be able to apply AI algorithms to real-world problems, evaluate their performance, and explore emerging trends in AI technologies, preparing them for advanced studies and careers in the field.

Minor (MI) – 4: Artificial Intelligence **[Theory : 60 Hrs.]**

Course Content:

Unit 1: Introduction to Artificial Intelligence (12 Lectures)
Definition and scope of AI
Historical overview and key milestones
Differentiating AI from human intelligence

Unit 2: AI Subfields and Technologies (12 Lectures)
Machine learning: Supervised, unsupervised, and reinforcement learning
Deep learning and neural networks
Natural language processing (NLP) and computer vision

Unit 3: Applications of AI (12 Lectures)
AI in healthcare: Diagnosis, treatment, and medical imaging
AI in finance: Fraud detection, algorithmic trading, and risk assessment
AI in transportation: Autonomous vehicles and traffic optimization
AI in customer service and chatbots
AI in education: Personalized learning and intelligent tutoring systems

Unit 4: Ethical and Social Implications of AI (12 Lectures)
Bias and fairness in AI systems
Privacy and data protection concerns
Impact of AI on employment and the workforce
AI and social inequality

Unit 5: Other Important Issues (12 Lectures)
Ethical guidelines and responsible AI practices
AI and Innovation
Emerging trends and future directions in AI
AI and creativity: Generative models and artistic applications

Suggested Readings:

1. AI for All – Biswapati Jana, Debkumar Bera, Sharmistha Jana – Amitrakshar™ Publishers Kolkata
2. AI for Everyone- S Goswami, AK Das, A Chakraborty- Pearson

SKILL ENHANCEMENT COURSE (SEC)

SEC 3: PYTHON **Credits 03 (Full Marks: 50)**

OBJECTIVE OF THE COURSE -

The objectives of this course are to make the student understand programming language, programming, concepts of Loops, reading a set of Data, stepwise refinement, Functions, Control structure, Arrays. After completion of this course the student is expected to analyze the real-life problem and write a program in 'Python' language to solve the problem. The main emphasis of the course will be on problem solving aspect i.e., developing proper algorithms.

- After completion of the course the student will be able to
- Develop efficient algorithms for solving a problem.
- Use the various constructs of a programming language viz. conditional, iteration and recursion.
- Implement the algorithms in "Python" language.
- Use simple data structures like arrays, stacks and list in solving problems.

SEC3P: PYTHON **Credits 03**

Course Outline:

Planning the Computer Program: Concept of problem solving, Problem definition, Problem design, Debugging, Types of Errors in programing, Documentation

Techniques of Problem Solving: Flowcharting, decision table, algorithms, Structured programing concepts, Programing methodologies viz. top-down and bottom-up

Overview to Python Programming: Structure of Python Program, Elements of Python

Introduction to Python: Python Interpreter, Python shell, Indentation, Atoms, Identifiers and keywords, literals, Strings, Operator (Arithmetic Operator, Relational Operator, Logical or Boolean Operator, Assignment Operator, Ternary operator, Bitwise Operator)

Creating Python Programs: Input and Output Statements, Control Statements (Branching, Looping, Conditional Statement, Exit, Function, Difference, between break, continue and pass). Defining Functions, Default arguments and Exception handling

Iterations and Recursions: Conditional execution, Alternative execution, Nested conditionals, Return statements, Recursion, Stack diagrams for recursive functions, Multiple assignment, While statement, For statement.

String and List: String as a compound data type, Length, Traversal and the for loop, String slices, String Comparison, A find function, Looping and counting, List values, Accessing elements, List length, List membership, List and for loops, List operations, List deletion, Cloning lists, Nested Lists

Object Oriented Programing: Introduction to Classes, Objects and Methods, Standard Libraries

Suggested Readings:

1. Jhon V. Guttag, "Introduction to Computation and Programming Using Python", MIT Press
2. Allen Downey, "Think Python: How to Think a Computer Scientist", O'Reilly
3. Mark Lutz, "Learning Python, 5th Edition", O'Reilly

INTERNSHIP/APPRENTICESHIP (INT)

Credit-04 Marks: 50 **(120 hours, 8 weeks)**

Guideline for internship/apprenticeship:

The internship program will commence at the beginning of the third semester and will be evaluated upon its completion at the end of the fourth semester.

1. A student may visit an industry for industry-related issues or a research institution, laboratory, or academic institute to engage in internship under the guidance of an industry official, scientist, or academician.
2. A student may work at a company's outlet or similar type of office, Professional bodies, etc. to develop programming / process etc. under the supervision of the respective official or a faculty of his/her own college teacher or a teacher from another college/university/industry person.
3. Interns may engage in advanced learning in topics beyond their course curriculum, under the guidance of their respective mentor.
4. Interns may be assigned a problem to solve using any programming language.
5. Interns may be assigned to design a webpage for college /department/ Entrepreneur/ start-up under the mentor's guidance.
6. Interns may be allowed to work as quantitative researchers in advance topics in Computer Science and applications from the reputed institute/ organization.

General instructions:

a) Each intern must maintain a daily logbook of activities.
b) At the end of the internship, a completion certificate must be obtained from the mentor, supervisor, or concerned authority.
c) Interns are expected to strictly adhere to the assigned tasks and deadlines.