Mahakaushal University, Jabalpur (M.P.)

Syllabus For B.Sc
(Artificial Intelligence and Machine Learning)
2021-22

Course Code : (BSC AI&ML)

Department of Computer Science Faculty of Computer Science Application

Duration of Course: 3 Years

Examination Mode: Year Examination

System: Non Grading

Mahakaushal University
Village-Aithakheda ,Mukunwara Road ,Post-Tilwara
Jabalpur (M.P.)482003

Mahakaushal University, Jabalpur (M.P.)

Class : B.A./B.Sc./B.Com./BBA/BCA/B.A --- **I Year**
Subject : Foundation Course
Paper : I
Paper Name : (Hindi Language & Moral Values) –I
Paper code : FC102-T

Particulars

Unit-I	हिन्दी भाषा स्वतंत्रता पुकारती (कविता) – जयशंकर प्रसाद पुष्प की अभिलाषा (कविता) – माखनलाल चतुर्वेदी वाक्य संरचना और अशुद्धियां (संकलित)
Unit-II	हिन्दी भाषा नमक का दरोगा (कहानी) – प्रेमचंद एक थे राजा भोज (निबंध) – डॉ त्रिभुवननाथ शुक्ल पर्यायवाची, विलोम, एकार्थी, अनेकार्थी एवं शब्दयुग्म शब्द (संकलित)
Unit-III	हिन्दी भाषा भगवान बुद्ध (निबंध) – स्वामी विवेकानंद लोकतंत्र एक धर्म है (निबंध) – डॉ. सर्वपल्ली राधाकृष्णन नही रूकती है नदी – हीरालाल बाछोतिया पल्लवन
Unit-IV	हिन्दी भाषा अफसर (निबंध) – शरद जोशी हमारी सांस्कृतिक एकता (निबंध) – रामधारी सिंह दिनकर (एक भारत श्रेष्ठ भारत के अन्तर्गत) संक्षेपण (संकलित)
Unit-V	नैतिक मूल्य नैतिक मूल्य परिचय एवं वर्गीकरण (आलेख) – डॉ. शशि राय आचरण की सभ्यता (निबंध) – सरदार पूर्णसिंह अंतर्ज्ञान और नैतिक जीवन (लेख) – स्वामी श्रद्धानंद

Mahakaushal University, Jabalpur (M.P.)

Class : B.A./B.Sc./B.Com./BBA/BCA/B.A --- **I Year**
Subject : Foundation Course
Paper : II
Paper Name : English Language-I
Paper code : FC102-T

Particulars

Unit-I	Where the mind is without fear : Rabindranath Tagore. The Hero: R.K. Narayan. Tryst with Density Jawaharlaal Nehru. Indian weavers : Sarojnini Naidu. The portrait of a lady : Khushwant Singh. The Solitary Reaper : Willian Wordsworth.
Unit-II	Basic Language Skills : Vocabulary, Synonyms, Antonyms, Word formation, Prefixes Suffixes.
Unit-III	Basic Language Skills : Uncountable nouns, verbs, tenses, adverbs.
Unit-IV	Comprehension / Unseen Passage.
Unit-V	Composition and Paragraph writing.

Class : B.A./B.Sc./B.Com./BBA/BCA/B.A --- **I Year**
Subject : Foundation Course
Paper : III
Paper Name : (Entrepreneurship Development)
Paper code : FC103-T

Particulars

Unit-I	Entrepreneurship Development - Concept and importance, function of enterpriser, Goal determination- Problems Challenges and solutions.
Unit-II	Project Proposal - need and Objects – Nature of organization, Production Management, Financial Management, Marketing Management, Consumer Management.
Unit-III	Role of regulatory Institutions, Role of development Organizations and self employment oriented schemes, Various growth schemes.
Unit-IV	Financial Management for Project – Financial institution and their role, Capital estimation and arrangement, cost and price determination, accounting management
Unit-V	Problem of entrepreneur – Problem relating capital, Problem relating Registration, administration problem and how to overcome

BAIM101-T
Introduction to Artificial Intelligence

UNIT-I **Marks -10**

Introduction to AI - Intelligent Agents, Problem-Solving Agents, Searching for Solutions - Breadth-first search, Depth-first search, Hill-climbing search, Simulated annealing search, Local Search in Continuous Spaces.

UNIT-II **Marks -10**

Games - Optimal Decisions in Games,Alpha–Beta Pruning, Defining Constraint Satisfaction Problems, Constraint Propagation, Backtracking Search for CSPs, Knowledge-BasedAgents, Logic Propositional Logic, Propositional Theorem Proving: Inference and proofs, Proof by resolution, Horn clauses and definite clauses.

UNIT-III **Marks -10**

First-Order Logic - Syntax and Semantics of First-Order Logic, Using First Order Logic, KnowledgeEngineeringinFirst-OrderLogic.InferenceinFirst-OrderLogic:Propositionalvs. First-Order Inference, Unification, Forward Chaining, Backward Chaining, Resolution. Knowledge Representation: Ontological Engineering, Categories and Objects, Events.

UNIT-IV **Marks -10**

PlanningDefinitionofClassicalPlanning,AlgorithmsforPlanningwithStateSpaceSearch, Planning Graphs, other Classical Planning Approaches, Analysis of Planning approaches. Hierarchical Planning.

UNIT-V **Marks -10**

ProbabilisticReasoning:Acting under Uncertainty, Basic Probability Notation Bayes'Rule and Its Use, Probabilistic Reasoning, Representing Knowledge in an Uncertain Domain, The Semantics of Bayesian Networks, Efficient Representation of Conditional Distributions, Approximate Inference in Bayesian Networks, Relational and First- Order Probability.

Reference Books:

- Artificial Intelligence: A Modern Approach, Third Edition, Stuart Russell and Peter Norvig, Pearson Education.
- Artificial Intelligence,3rd Edn., E.Rich andK. Knight (TMH)
- Artificial Intelligence,3rdEdn.,PatrickHennyWinston,Pearson Education.
- Artificial Intelligence, Shivani Goel, Pearson Education.
- Artificial Intelligence and Expert systems–Patterson, Pearson Education.

BAIM101-P Artificial Intelligence LAB

Machine Learning Model Building

- **Description**: Learn how to build and train various types of machine learning models, such as linear regression, decision trees, support vector machines, and neural networks.
- **Practical Activities**: Data preprocessing, feature engineering, model selection, training, and evaluation using tools like Scikit-learn and Tensor Flow.

2. Natural Language Processing (NLP)

- **Description**: Understand and implement techniques for processing and analyzing textual data.
- **Practical Activities**: Text preprocessing, sentiment analysis, named entity recognition (NER), and text generation using libraries like NLTK, SpaCy, and Hugging Face Transformers.

3. Computer Vision

- **Description**: Explore techniques for image and video analysis.
- **Practical Activities**: Image classification, object detection, and image segmentation using OpenCV, Tensor Flow, and Py Torch.

4. Reinforcement Learning

- **Description**: Study how agents can learn to make decisions by interacting with their environment.
- **Practical Activities**: Implementing algorithms like Q-learning, Deep Q Networks (DQN), and Policy Gradients using environments from Open AI Gym.

5. Speech Recognition

- **Description**: Learn to convert spoken language into text.
- **Practical Activities**: Building and training speech-to-text models using datasets like Libri Speech and tools like Google Cloud Speech-to-Text API or Mozilla DeepSpeech.

6. Recommendation Systems

- **Description**: Develop systems that provide personalized recommendations.
- **Practical Activities**: Implementing collaborative filtering, content-based filtering, and hybrid recommendation systems using libraries like Surprise and TensorFlow.

7. AI for Robotics

- **Description**: Integrate AI with robotics for autonomous decision-making.
- **Practical Activities**: Building autonomous navigation systems, path planning, and robotic manipulation using ROS (Robot Operating System) and machine learning models.

8. Generative Adversarial Networks (GANs)

- **Description**: Learn how to generate new data samples similar to a given dataset.
- **Practical Activities**: Implementing GANs to create realistic images, videos, or audio using TensorFlow or PyTorch.

9. Anomaly Detection

- **Description**: Identify unusual patterns that do not conform to expected behavior.
- **Practical Activities**: Implementing anomaly detection algorithms in domains like fraud detection, network security, and predictive maintenance using machine learning techniques.

10. AI Ethics and Fairness

- **Description**: Understand the ethical considerations and ensure fairness in AI systems.
- **Practical Activities**: Conducting bias detection and mitigation in machine learning models, ensuring transparency and accountability, and using frameworks like IBM's AI Fairness 360.

BAIM102-T
Linear algebra and Calculus

UNIT-I **Marks -10**

Fundamentals of Calculus:- Limits and continuity, Differentiation and derivatives, Applications of derivatives, Integration and antiderivatives.

UNIT-II **Marks -10**

Multivariable Calculus:- Partial derivatives, Multiple integrals, Vector calculus (gradient, divergence, curl), Line and surface integrals.

UNIT-III **Marks -10**

Linear Algebra Fundamentals:- Vectors and vector spaces, Matrices and matrix operations, Determinants and their properties, Eigenvalues and eigenvectors.

UNIT-IV **Marks -10**

Linear Transformations and Matrix Decompositions:- Linear transformations and their properties, Orthogonalization methods (Gram-Schmidt process), Matrix decompositions (LU, QR, eigenvalue decompositions).

UNIT-V **Marks -10**

Applications of Linear Algebra and Calculus:- Optimization problems and techniques, Differential equations and their solutions, Applications in physics, engineering, and computer science

Reference Books:

"Calculus: Early Transcendentals" by James Stewart

"Calculus" by Michael Spivak

"Thomas' Calculus" by Joel Hass, Christopher Heil, and Maurice D. Weir

"Multivariable Calculus" by James Stewart

"Vector Calculus, Linear Algebra, and Differential Forms: A Unified Approach" by John Hubbard and Barbara Burke Hubbard

"Calculus on Manifolds: A Modern Approach to Classical Theorems of Advanced Calculus" by Michael Spivak.

"Linear Algebra and Its Applications" by David C. Lay, Steven R. Lay, and Judi J. McDonald

"Introduction to Linear Algebra" by Gilbert Strang

"Linear Algebra Done Right" by Sheldon Axler

"Matrix Analysis and Applied Linear Algebra" by Carl D. Meyer

"Numerical Linear Algebra" by Lloyd N. Trefethen and David Bau III

"Applied Linear Algebra" by Olver and Shakiban

"Introduction to the Theory of Neural Computation" by John A. Hertz, Anders Krogh, and Richard G. Palmer

"Introduction to Applied Linear Algebra: Vectors, Matrices, and Least Squares" by Stephen Boyd and Lieven Vandenberghe

"Introduction to Partial Differential Equations with Applications" by E. C. Zachmanoglou and Dale W. Thoe

BAIM103-T

Digital Logic and system design

UNIT-I **Marks -10**

Introduction to Digital Logic:- Binary number systems, Boolean algebra and logic gates, Combinational logic circuits, Boolean function simplification techniques.

UNIT-II **Marks -10**

Combinational Logic Design:- Designing combinational circuits using logic gates, Arithmetic circuits (adders, subtractors), Multiplexers, demultiplexers, and encoders, Decoders and priority encoders.

UNIT-III **Marks -10**

Sequential Logic Design:- Flip-flops and latches, State diagrams and state tables, Analysis and design of synchronous sequential circuits, Finite State Machines (FSMs).

UNIT-IV **Marks -10**

Memory and Programmable Logic:- Memory devices (RAM, ROM, EEPROM), Programmable Logic Devices (PLDs), Field-Programmable Gate Arrays (FPGAs), Introduction to Hardware Description Languages (HDLs) such as Verilog and VHDL.

UNIT-V **Marks -10**

Advanced Topics in Digital Logic Design:- High-level synthesis, Design for Testability (DFT), Hardware description and synthesis, Application-specific integrated circuit (ASIC) design basics.

Reference Books:

"Digital Design" by M. Morris Mano and Michael D. Ciletti

"Fundamentals of Digital Logic with Verilog Design" by Stephen Brown and Zvonko Vranesic

"Digital Logic and Computer Design" by M. Morris Mano

"Logic and Computer Design Fundamentals" by M. Morris Mano and Charles R. Kime

"Digital Systems Design with FPGAs and CPLDs" by Ian Grout

"Digital Design: Principles and Practices" by John F. Wakerly

"Sequential Logic and Verilog HDL Fundamentals" by Joseph Cavanagh

"Digital Design: Principles and Practices" by John F. Wakerly

"Digital Systems: Principles and Applications" by Ronald J. Tocci et al.

"Digital Design Using Digilent FPGA Boards: Verilog / Vivado Edition" by Richard E. Haskell and Darrin M. Hanna

"FPGA Prototyping by VHDL Examples: Xilinx MicroBlaze MCS SoC" by Pong P. Chu

"FPGA-based Implementation of Signal Processing Systems" by Roger Woods et al.

"ASIC Design in the Silicon Sandbox: A Complete Guide to Building Mixed-Signal Integrated Circuits" by Keith Barr

"Digital Systems Engineering" by William J. Dally and John W. Poulton

"Advanced Digital Logic Design Using VHDL, State Machines, and Synthesis for FPGA's" by Sunggu Lee

BAIM104-T

Problem solving using C

UNIT-I **Marks -10**

Introduction to C Programming:- Overview of the C programming language, Basic syntax and structure of C programs, Variables, data types, and operators, Input and outpu operations in C.

UNIT-II **Marks -10**

Control Flow and Functions:- Decision-making statements (if-else, switch), Looping statements (for, while, do-while), Functions and parameter passing, Recursion.

UNIT-III **Marks -10**

Arrays and Pointers:- Arrays and strings in C, Pointer basics and pointer arithmetic, Dynamic memory allocation, Multi-dimensional arrays.

UNIT-IV **Marks -10**

File Handling and Data Structures:- File operations in C, Basics of data structures (linked lists, stacks, queues), Implementation of data structures in C, Sorting and searching algorithms.

UNIT-V **Marks -10**

Advanced Topics in C Programming:- Preprocessor directives,Bit manipulation and bitwise operators, Introduction to low-level programming concepts, Debugging and optimization techniques.

Reference Books:

"C Programming: A Modern Approach" by K. N. King
"The C Programming Language" by Brian W. Kernighan and Dennis M. Ritchie
"Programming in C" by Stephen G. Kochan
"Pointers in C: A Hands-on Approach" by Hrishikesh Dewan and Nidhi Agarwal
"Data Structures Using C" by Aaron M. Tenenbaum et al.
"Modern C: Programming with Testing and Debugging" by Jens Gustedt

BAIM104-P Problem solving using C LAB

- **Description**: Learn how to handle basic input and output operations in C.
- **Practical Activities**: Writing programs to read and print different data types using `scanf` and `printf`.

2. Control Structures

- **Description**: Understand and implement decision-making and looping constructs.
- **Practical Activities**: Writing programs using `if`, `else if`, `else`, `switch`, `for`, `while`, and `do-while` loops.

3. Functions

- **Description**: Learn how to create and use functions in C.
- **Practical Activities**: Writing programs that define and call user-defined functions, understand scope and lifetime of variables, and pass arguments by value and by reference.

4. Arrays and Strings

- **Description**: Work with arrays and strings for data storage and manipulation.
- **Practical Activities**: Writing programs to perform operations like searching, sorting, and manipulating arrays and strings (e.g., string concatenation, comparison).

5. Pointers and Dynamic Memory Allocation

- **Description**: Understand pointers and their usage in dynamic memory management.
- **Practical Activities**: Writing programs to demonstrate pointer arithmetic, pointer to functions, and dynamic memory allocation using `malloc`, `calloc`, `realloc`, and `free`.

6. Structures and Unions

- **Description**: Learn to define and use structures and unions to create complex data types.
- **Practical Activities**: Writing programs to define structures and unions, access their members, and use them in functions.

7. File Handling

- **Description**: Understand file operations for reading and writing data to files.
- **Practical Activities**: Writing programs to open, read, write, and close files using `fopen`, `fscanf`, `fprintf`, `fgets`, and `fputs`.

8. Command Line Arguments

- **Description**: Learn to handle command line arguments passed to a program.
- **Practical Activities**: Writing programs that take arguments from the command line and process them (e.g., counting the number of arguments, performing operations based on arguments).

9. Preprocessor Directives

- **Description**: Understand the use of preprocessor directives for macro definition, file inclusion, and conditional compilation.
- **Practical Activities**: Writing programs to define macros, include header files, and use `#ifdef`, `#ifndef`, and `#endif` for conditional compilation.

10. Data Structures (Linked Lists, Stacks, Queues)

- **Description**: Implement and use basic data structures like linked lists, stacks, and queues.
- **Practical Activities**: Writing programs to create, traverse, insert, and delete nodes in linked lists; implement stack operations (push, pop) and queue operations (enqueue, dequeue).

BAIM105-T
Ethics and Human Values

UNIT-I **Marks -10**

Introduction to Ethics and Privacy in Data Science, Understanding the ethical implications of data science, Introduction to privacy concerns in data collection, storage, and analysis, Case studies illustrating ethical dilemmas and privacy breaches.

UNIT-II **Marks -10**

Legal and Regulatory Frameworks, Overview of relevant laws and regulations governing data privacy (e.g., GDPR, CCPA), Ethical guidelines and professional codes of conduct in data science, Impact of legal and regulatory compliance on data science practices.

UNIT-III **Marks -10**

Ethical Data Collection and Handling, Principles of ethical data collection and management, Bias and fairness in data collection and algorithmic decision-making, Techniques for ensuring data integrity and protecting sensitive information.

UNIT-IV **Marks -10**

Ethical Analysis and Interpretation of Data, Ethical considerations in data analysis and interpretation, Transparency and accountability in algorithmic decision-making, Evaluating the ethical implications of data-driven insights.

UNIT-V **Marks -10**

Privacy Preservation and Ethical Data Sharing, Techniques for preserving privacy

in data sharing and dissemination, Ethical guidelines for responsible data sharing and collaboration, Emerging technologies and approaches for privacy-enhanced data analytics.

Reference Book:

"Ethics of Big Data: Balancing Risk and Innovation" by Kord Davis

"Data and Goliath: The Hidden Battles to Collect Your Data and Control Your World" by Bruce Schneier

"Weapons of Math Destruction: How Big Data Increases Inequality and Threatens Democracy" by Cathy O'Neil

"Automating Inequality: How High-Tech Tools Profile, Police, and Punish the Poor" by Virginia Eubanks.

BAIM106-T

Computer Architecture

UNIT-I **Marks -10**

Introduction to Computer Architecture:- Overview of computer architecture, Evolution of computer architecture, Basic concepts and terminology.

UNIT-II **Marks -10**

Instruction Set Architecture (ISA):- Types of ISA (RISC, CISC, etc.), Instruction formats and types, Addressing modes, Assembly language programming.

UNIT-III **Marks -10**

Processor Design:- Processor components (ALU, Control Unit, Registers, etc.), Pipelining and its stages, Superscalar and VLIW architectures, Hazards and their solutions.

UNIT-IV **Marks -10**

Memory Hierarchy:- Memory types (RAM, ROM, Cache, Virtual Memory), Cache memory organization and optimization, Memory management techniques.

UNIT-V **Marks -10**

Parallelism and Multiprocessing:- Flynn's taxonomy, Multi-core and multi-processor systems, Synchronization and concurrency, Parallel programming paradigms.

Reference Books:

"Computer Organization and Design: The Hardware/Software Interface" by David A. Patterson and John L. Hennessy

"Computer Architecture: A Quantitative Approach" by John L. Hennessy and David A. Patterso

"Computer Organization and Design: The Hardware/Software Interface" by David A. Patterson and John L. Hennessy

"Computer Architecture: A Quantitative Approach" by John L. Hennessy and David A. Patterson

"Computer Organization and Design: The Hardware/Software Interface" by David A. Patterson and John L. Hennessy.

"Computer Organization and Design: The Hardware/Software Interface" by David A. Patterson and John L. Hennessy.

BAIM106P Computer Architecture LAB

1. **Study of Basic Computer Organization**
   - Block diagram of computer system (Input, Output, CPU, Memory)
2. **Number System Conversion Program**
   - Conversion between Binary, Decimal, Octal, and Hexadecimal
3. **Binary Arithmetic Operations**
   - Binary addition, subtraction (2's complement method)
4. **Implementation of Logic Gates**
   - AND, OR, NOT, NAND, NOR, XOR using truth tables
5. **Half Adder and Full Adder Circuit**
   - Design and verification using logic gates
6. **Half Subtractor and Full Subtractor Circuit**
7. **Study of Flip-Flops**
   - SR, JK, D, T Flip-Flops (truth table & timing diagram)
8. **Design of Counters**
   - Synchronous and Asynchronous counters
9. **Study of Registers and Shift Registers**
10. **Design of Multiplexer and Demultiplexer**
11. **Study of Encoder and Decoder Circuits**
12. **Memory Organization Practical**

- Study of RAM, ROM, Cache memory

13. **Simulation of Instruction Cycle**

- Fetch–Decode–Execute cycle demonstration

14. **Study of Microprocessor Architecture**

- Architecture of **Intel 8086**

15. **Assembly Language Programming**

- Simple programs (Addition, Subtraction, Looping, Data Transfer)

Class : B.A./B.Sc./B.Com./BBA/BCA/B.A--- **II Year**
Subject : Foundation Course
Paper : I
Paper Name : (Hindi Language & Moral Values) –II
Paper code : FC201-T

Particulars

Unit-I	हिन्दी भाषा वह तोड़ती पत्थर (कविता) – सूर्यकान्त त्रिपाठी निराला दिमागी गुलामी (निबंध) – राहुल सांकृत्यायन वर्ण – (स्वर– व्यंजन, वर्गीकरण, उच्चारण स्थान)
Unit-II	हिन्दी भाषा नारीत्व का अभिशाप (निबंध) – महादेवी वर्मा चीफ की दावत (कहानी) – भीष्म साहनी विराम चिन्ह – (संकलित)
Unit-III	हिन्दी भाषा चली फगुनाहट बौरे आम (ललित निबंध) – विवेकी राय इन्द्रधनुष का रहस्य (वैज्ञानिक लेख) – डॉ. कपूरमल जैन संधि – (संकलित) पल्लवन
Unit-IV	हिन्दी भाषा सपनों की उड़ान (प्रेरक निबंध) – ए.पी.जे. अब्दुल कलाम हमारा सौरमण्डल (संकलित) समास (संकलित)
Unit-V	नैतिक मूल्य शिकागो व्याख्यान (व्याख्यान) – स्वामी विवेकानंद धर्म और राष्ट्रवाद (लेख) – महर्षि अरविन्द सादगी (आत्मकथा) – महात्मा गांधी चित्त जहाँ भय शून्य (कविता) – रवीद्रनाथ टैगोर

Class : B.A./B.Sc./B.Com./BBA/BCA/B.A--- **II Year**
Subject : Foundation Course
Paper : II
Paper Name : English Language - II
Paper code : FC202-T

Particulars

Unit-I	Tree: Tina Morris. Night of the Scorpion: Nissim Ezekiel. Idgah : Premchand (translated by khushwant Singh). Letter of God : G.L. Swanteh (translated by Donald A. Yates). My Bank Account : Stephen Leacock. God sees the truth but waits : Leo Tolstoy.
Unit-II	Basic English Language : Idioms, Proverbs and Phrasal Verbs, Tenses, Prepositions, Determiners, Verbs Articles, Nouns & Pronouns.
Unit-III	Short Essay on given topics. Correspondence Skills (Formal & Informal letters and Application)
Unit-IV	Translation of sentences / passage English to Hindi and Hindi to English.
Unit-V	Drafting CV.

Class	:	B.A./B.Sc./B.Com./BBA/BCA/B.A--- **II Year**
Subject	:	Foundation Course
Paper	:	III
Paper Name	:	(Environmental Studies)
Paper code	:	FC203-T

Particulars

Unit-I	**Study of Environment and Ecology :** (a.) Definition and importance. (b.) Public participation and public awareness. (c.) Ecology – Introduction. (d.)Ecosystem – Concepts, components, structure & function, energy flow. food chain. food web, ecological pyramids and types.
Unit-II	**Environmental Pollution and Population :** (a.) Air, water, noise, Heat and nuclear pollution. definition, causes, effect and prevention of pollution. (b.)Population growth, disparities between countries. (c.) Population explosion, family welfare programme. (d.)Environment and human health. (e.) Cleanliness and disposal of domestic waste.
Unit-III	**Natural Resources, Problems and Conservation :** (a.) Water Resources. (b.)Forest Resources. (c.) Land Resources. (d.)Food Resources. (e.) Energy Resources.
Unit-IV	Bio-diversity and its Protection (a.) Introduction- Genetic, species and ecosystem diversity. (b.) Value of bio-diversity − Consumable use : Productive use. Social, Moral and Aesthetic values (c.) India as a nation of mega bio-diversity centre. bio-diversity at national and local levels. (d.) Threats to bio-diversity − Loss of habitat, poaching of wildlife, man and wildlife conflicts.
Unit-V	Disaster Management and Environment and Environmental laws : (a.) Disaster Management − flood, earthquake, cyclones and landslides. (b.)Conservation of laws for air and water pollution. (c.) Wildlife conservation laws. (d.)Role of information technology in protecting environment and health.

BAIM201-T

Data structure

UNIT-I **Marks -10**

Introduction to Data Structures:- Overview of data structures and their importance in computer science, Basic terminology and concepts, Introduction to arrays and linked lists.

UNIT-II **Marks -10**

Stacks and Queues:- Understanding the stack and queue data structures,

Operations on stacks and queues, Applications of stacks and queues in solving

real-world problems.

UNIT-III **Marks -10**

Trees and Binary Trees:- Introduction to tree data structure, Binary trees: properties, traversal techniques, Binary search trees and their operations.

UNIT-IV **Marks -10**

Graphs and Graph Algorithms:- Basics of graph theory, Representation of

graphs and graph traversal algorithms, Shortest path algorithms, minimum spanning

tree algorithms.

UNIT-V **Marks -10**

Advanced Data Structures:- Introduction to advanced data structures such as heaps, hash tables, and advanced trees (AVL trees, B-trees), Comparison of different data structures and their performance characteristics, Applications and case studies demonstrating the use of advanced data structures.

Reference Book:

"Data Structures and Algorithms in Java "by Robert Lafore.
"Algorithms in C, Parts 1-4: Fundamentals, Data Structures, Sorting, Searching" by Robert Sedgewick
"Introduction to Algorithms" by Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein
"Algorithms" by Robert Sedgewick and Kevin Wayne
"Introduction to Algorithms" by Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein

BAIM202-T

Object Oriented Programming Using C++

UNIT-I **Marks -10**

Introduction to object oriented programming, user defined types, structures, unions, polymorphism, encapsulation. Getting started with C++ syntax, data-type, variables, strings, functions, default values in functions, recursion, namespaces, operators, flow control, arrays and pointers.

UNIT-II **Marks -10**

Abstraction mechanism: Classes, private, public, constructors, destructors, member data, member functions, inline function, friend functions, static members, and references. Inheritance: Class hierarchy, derived classes, single inheritance, multiple, multilevel, hybrid inheritance, role of virtual base class, constructor and destructor execution, base initialization using derived class constructors.

UNIT-III **Marks -10**

Polymorphism: Binding, Static binding, Dynamic binding, Static polymorphism: Function Overloading, Ambiguity in function overloading, Dynamic polymorphism: Base class pointer, object slicing, late binding, method overriding with virtual functions, pure virtual functions, abstract classes.

UNIT-IV **Marks -10**

Operator Overloading: This pointer, applications of this pointer, Operator function, member and non member operator function, operator overloading, I/O operators. Exception handling: Try, throw, and catch, exceptions and derived classes, function exception declaration, unexpected exceptions, exception when handling exceptions, resource capture and release.

UNIT-V **Marks -10**

Dynamic memory management, new and delete operators, object copying, copy constructor, assignment operator, virtual destructor.
Template: template classes, template functions.
Standard Template Library: Fundamental idea about string, iterators, hashes, iostreams and other types.
Namespaces: user defined namespaces, namespaces provided by library.
Object Oriented Design, design and programming, role of classes.

Text Books:

1. Object Oriented Programming with C++ by E. Balagurusamy, McGraw-HillEducation (India)
2. ANSI and Turbo C++ by Ashoke N. Kamthane, Pearson Education

Reference Books:

1. Big C++ - Wiley India
2. C++: The Complete Reference- Schildt, McGraw-Hill Education (India)
3. C++ and Object Oriented Programming – Jana, PHI Learning.
4. Object Oriented Programming with C++ - Rajiv Sahay, Oxford
5. Mastering C++ - Venugopal, McGraw-Hill Education (India)

BAIM202-P C++ LAB

Basic Input and Output

- **Description**: Learn how to handle basic input and output operations in C++.
- **Practical Activities**: Writing programs to read and print different data types using `cin` and `cout`, and using manipulators for formatting output.

2. Control Structures

- **Description**: Understand and implement decision-making and looping constructs.
- **Practical Activities**: Writing programs using `if`, `else if`, `else`, `switch`, `for`, `while`, and `do-while` loops to solve various problems.

3. Functions

- **Description**: Learn how to create and use functions in C++.
- **Practical Activities**: Writing programs that define and call user-defined functions, use function overloading, and understand default arguments and inline functions.

4. Classes and Objects

- **Description**: Understand the principles of object-oriented programming by creating classes and objects.
- **Practical Activities**: Writing programs to define classes, create objects, and implement member functions. Demonstrating concepts of encapsulation, constructors, and destructors.

5. Operator Overloading

- **Description**: Learn to overload operators to work with user-defined data types.
- **Practical Activities**: Writing programs to overload operators such as +, –, *, ==, and << for custom classes.

6. Inheritance and Polymorphism

- **Description**: Explore the concepts of inheritance and polymorphism in C++.
- **Practical Activities**: Writing programs to create base and derived classes, use virtual functions, and demonstrate runtime polymorphism through method overriding.

7. Templates

- **Description**: Implement generic programming using templates.
- **Practical Activities**: Writing programs that use function templates and class templates to handle different data types with a single code base.

8. Exception Handling

- **Description**: Learn how to handle runtime errors using exception handling.
- **Practical Activities**: Writing programs that use `try`, `catch`, and `throw` to handle exceptions. Implementing custom exception classes.

9. File Handling

- **Description**: Understand file operations for reading and writing data to files.
- **Practical Activities**: Writing programs to open, read, write, and close files using file streams (`ifstream`, `ofstream`, and `fstream`).

10. STL (Standard Template Library)

- **Description**: Utilize the components of the Standard Template Library to write efficient and effective code.
- **Practical Activities**: Writing programs that use STL components like vectors, lists, queues, stacks, maps, and algorithms such as `sort`, `find`, and `accumulate`.

BAIM203-T
Web Technology

UNIT-I **Marks -10**

Introduction to Web Technologies:- Overview of the Internet and World Wide Web, Client-server architecture and web protocols (HTTP, HTTPS), Introduction to HTML, CSS, and JavaScript.

UNIT-II **Marks -10**

Frontend Development:- Advanced HTML5 and CSS3 features, Responsive web design principles and techniques, Introduction to frontend frameworks/libraries (e.g., Bootstrap, React, Vue.js).

UNIT-III **Marks -10**

Backend Development:- Introduction to server-side scripting languages (e.g., PHP, Node.js, Python), Database management systems and SQL fundamentals, Building dynamic web applications with server-side frameworks (e.g., Express.js, Django).

UNIT-IV **Marks -10**

Web Security and Performance Optimization:- Understanding common web security threats and vulnerabilities, Techniques for securing web applications (e.g., HTTPS, input validation, authentication, authorization), Performance optimization strategies for web applications (e.g., minification, caching, CDN).

UNIT-V **Marks -10**

Emerging Web Technologies and Trends:- Introduction to progressive web apps (PWAs) and single-page applications (SPAs) Exploring the Internet of Things (IoT) and its impact on web development Discussion on current trends and future directions in web technology.

Reference Book:

"Web Development and Design Foundations with HTML5" by Terry Felke-Morris
"Learning Web Design: A Beginner's Guide to HTML, CSS, JavaScript, and Web Graphics" by Jennifer Robbins.
"Node.js Web Development: Server-side Development with Node 10 made easy" by David Herron.
"Web Application Security: A Beginner's Guide" by Bryan Sullivan.
"Building Progressive Web Apps: Bringing the Power of Native to the Browser" by Tal Ater.

BAIM204-T
Java Programming

UNIT-I Marks -10

Introduction to Java, history, Characteristics, abject oriented programming, data types, variables arrays, difference between java and C++

UNIT-II Marks -10

Control Statements: Selection, iteration, jump statements operators, Introduction to Classes, fundamentals Constructor methods, Stack, Class, inheritance, creating multilevel hierarchy, method over riding packages and interfaces, exception handling, multithreaded programming I/o applets.

UNIT-III Marks -10

Java Library, String handling, string comparison, string buffer, utility Classer, vector stack dictionary applet Class, introduction to AWT, working with frame windows.

UNIT-IV Marks -10

Java beans, beans architecture, AWT components, Advantages of Java beans, beans Serialization, JDBC Class & Methods, API Components, IDBC Components Driver, Connectivity to database.

UNIT-V Marks -10

Processing result and interfaces, RMI, Comparison of distributed and Non- distributed Java programs, Interfaces, RMI Architecture layer, ODBC, CORBA, CORBA Services and Products, CGI, Structure of CGI.

Reference Books

JAVA THE COMPLETE REFERENCE BY PATRICK NAUGHTON AND HERBERT SCHILDT. TMH PUBLICATION ISBN 0-07-463769-X
PROGRAMMING WITH JAVA BY E. BALAGURUSWAMY TMH PUBLICATIONS ISBN 0-07-463542-5 USING JAVA 1.2 BY JOSEPH WEBER. PHI – ISBN-81-203-1558-9

BAIM204-P Java LAB

Basic Input and Output

- **Description**: Learn how to handle basic input and output operations in Java.
- **Practical Activities**: Writing programs to read and print different data types using `Scanner` for input and `System.out` for output.

2. Control Structures

- **Description**: Understand and implement decision-making and looping constructs.
- **Practical Activities**: Writing programs using `if`, `else if`, `else`, `switch`, `for`, `while`, and `do-while` loops to solve various problems.

3. Classes and Objects

- **Description**: Understand the principles of object-oriented programming by creating classes and objects.
- **Practical Activities**: Writing programs to define classes, create objects, and implement member functions. Demonstrating concepts of encapsulation, constructors, and destructors.

4. Inheritance and Polymorphism

- **Description**: Explore the concepts of inheritance and polymorphism in Java.
- **Practical Activities**: Writing programs to create base and derived classes, use method overriding, and demonstrate runtime polymorphism using abstract classes and interfaces.

5. Exception Handling

- **Description**: Learn how to handle runtime errors using exception handling.
- **Practical Activities**: Writing programs that use `try`, `catch`, `finally`, and `throw` to handle exceptions. Implementing custom exception classes.

6. Collections Framework

- **Description**: Utilize Java's Collections Framework to manage groups of objects.
- **Practical Activities**: Writing programs that use collections like `ArrayList`, `LinkedList`, `HashMap`, `HashSet`, and `TreeSet`. Performing operations like adding, removing, and iterating over elements.

7. File Handling

- **Description**: Understand file operations for reading and writing data to files.
- **Practical Activities**: Writing programs to open, read, write, and close files using `FileReader`, `FileWriter`, `BufferedReader`, and `BufferedWriter`.

8. Multithreading

- **Description**: Learn to create and manage multiple threads to perform concurrent tasks.
- **Practical Activities**: Writing programs to create threads by extending `Thread` class and implementing `Runnable` interface. Synchronizing threads to avoid race conditions.

9. Graphical User Interface (GUI)

- **Description**: Design and implement GUI applications using Java's Swing library.
- **Practical Activities**: Writing programs to create windows, dialogs, buttons, text fields, and other GUI components. Handling events like button clicks.

10. Networking

- **Description**: Implement network programming using Java's networking classes.
- **Practical Activities**: Writing programs to create client-server applications using `Socket` and `ServerSocket` classes. Implementing simple chat applications or file transfer programs.

BAIM205-T
Cloud Computing

UNIT-I **Marks -10**

Introduction to Cloud Computing:- Overview of cloud computing concepts, models, and deployment types, Evolution of cloud computing and its impact on businesses and IT infrastructure, Cloud service providers and their offerings: AWS, Azure, Google Cloud Platform.

UNIT-II **Marks -10**

Cloud Infrastructure and Virtualization:-Virtualization technologies: hypervisors, containers (Docker, Kubernetes), Infrastructure as a Service (IaaS) offerings and architecture, Cloud storage solutions: object storage, block storage, file storage.

UNIT-III **Marks -10**

Cloud Platforms and Services :-Platform as a Service (PaaS) offerings: database services, application hosting, serverless computing, Software as a Service (SaaS) applications and their advantages, Hybrid cloud and multi-cloud architectures.

UNIT-IV **Marks -10**

Cloud Security and Compliance:- Security challenges in cloud computing: data breaches, identity management, compliance requirements, Cloud security best practices: encryption, access control, network security, Compliance standards and certifications: GDPR, HIPAA, SOC 2.

UNIT-V **Marks -10**

Cloud Migration and Management:- Strategies for cloud migration: lift-and-shift, re-platforming, refactoring, Cloud management tools and platforms: monitoring, orchestration, automation, Cost optimization techniques and tools.

Reference Book:

"Cloud Computing: Concepts, Technology & Architecture" by Thomas Erl, Ricardo Puttini, and Zaigham Mahmood, "Virtualization Essentials" by Matthew Portnoy, "Architecting the Cloud: Design Decisions for Cloud Computing Service Models" by Michael J. Kavis, "Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance" by Tim Mather, Subra Kumaraswamy, and Shahed Latif, "Migrating to the Cloud: A Practical Guide for Business Decision Makers" by Jason Nash and Darren Guckenheimer.

BAIM205-P Cloud Computing LAB

Virtual Machine Provisioning

- **Description**: Learn to provision and manage virtual machines (VMs) on cloud platforms.
- **Practical Activities**: Deploying VM instances on cloud providers like AWS EC2, Google Cloud Compute Engine, or Microsoft Azure Virtual Machines. Configuring and managing VMs using cloud management consoles or APIs.

2. Containerization with Docker

- **Description**: Understand containerization technology and how to use Docker for packaging, distributing, and running applications.
- **Practical Activities**: Building Docker images, creating containers, managing container lifecycles, and deploying containerized applications on cloud platforms.

3. Infrastructure as Code (IaC)

- **Description**: Learn to automate infrastructure provisioning and management using tools like Terraform or AWS CloudFormation.
- **Practical Activities**: Writing infrastructure code to define and deploy cloud resources such as virtual networks, storage, compute instances, and security groups.

4. Serverless Computing

- **Description**: Explore serverless computing models such as AWS Lambda, Google Cloud Functions, or Azure Functions for executing code without managing servers.
- **Practical Activities**: Developing and deploying serverless functions to perform tasks like data processing, event handling, or API endpoints on cloud platforms.

5. Cloud Storage Solutions

- **Description**: Understand different cloud storage options like object storage, block storage, and file storage services.
- **Practical Activities**: Using cloud storage services such as Amazon S3, Google Cloud Storage, or Azure Blob Storage to store and manage data. Configuring access controls, versioning, and lifecycle policies.

6. Database Services

- **Description**: Learn to use managed database services provided by cloud providers for storing and querying structured data.
- **Practical Activities**: Setting up and configuring databases like Amazon RDS, Google Cloud SQL, or Azure Database for MySQL/PostgreSQL/SQL Server. Importing data, creating tables, and executing queries.

7. Load Balancing and Auto-scaling

- **Description**: Implement load balancing and auto-scaling mechanisms to distribute traffic and handle varying workloads efficiently.
- **Practical Activities**: Configuring load balancers like AWS ELB, Google Cloud Load Balancing, or Azure Load Balancer to distribute incoming requests. Setting up auto-scaling policies based on CPU utilization or other metrics.

8. Monitoring and Logging

- **Description**: Learn to monitor and analyze cloud infrastructure and applications for performance, availability, and security.
- **Practical Activities**: Setting up monitoring dashboards using tools like AWS CloudWatch, Google Cloud Monitoring, or Azure Monitor. Configuring logging and alerting for detecting and responding to issues.

9. Network Security

- **Description**: Understand cloud networking concepts and implement security measures to protect data and resources.
- **Practical Activities**: Configuring network access controls, firewalls, VPNs, and security groups on cloud platforms. Implementing encryption, authentication, and authorization mechanisms.

10. Cloud Native Application Development

- **Description**: Develop applications designed to run natively on cloud platforms, leveraging cloud-native technologies and principles.
- **Practical Activities**: Building microservices architectures using container orchestration platforms like Kubernetes. Implementing cloud-native design patterns for scalability, resilience, and agility.

BAIM206-T

Natural Language processing

UNIT-I **Marks -10**

Introduction to Natural Language Processing:- Overview of NLP and its applications, Basics of linguistics and language processing, Text preprocessing techniques: tokenization, stemming, lemmatization.

UNIT-II **Marks -10**

Text Representation and Feature Engineering:- Bag-of-Words (BoW) model and term frequency-inverse document frequency (TF-IDF) representation, Word embeddings: Word2Vec, GloVe, and fast Text, Document embeddings: Doc2Vec, Paragraph Vectors.

UNIT-III **Marks -10**

Text Classification and Sentiment Analysis:- Supervised learning algorithms for text classification: Naive Bayes, Support Vector Machines (SVM), and logistic regression, Sentiment analysis techniques: lexicon-based methods, machine learning approaches, Evaluation metrics for text classification and sentiment analysis tasks.

UNIT-IV **Marks -10**

Sequence Modeling and Language Understanding;-Basics of sequence modeling: Hidden Markov Models (HMMs) and Conditional Random Fields (CRFs), Introduction to Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) networks, Applications of sequence models in part-of-speech tagging, named entity recognition, and dependency parsing.

UNIT-V **Marks -10**

Advanced Topics in Natural Language Processing:-Attention mechanisms in neural networks: Transformers and the Transformer architecture,Transfer learning and pre-trained language models: BERT, GPT, and their variants, Ethical considerations and challenges in NLP: bias, fairness, and privacy concerns.

Reference Book:

"Speech and Language Processing" by Daniel Jurafsky and James H. Martin."Natural Language Processing in Action" by Lane, Howard, and Hapke."Text Analytics with Python" by Dipanjan Sarkar. "Neural Network Methods in Natural Language Processing" by Goldberg, Yoav. "Natural Language Processing with Python and spaCy" by Yuli Vasiliev.

Class : B.A./B.Sc./B.Com./BBA/BCA/B.A --- **III Year**
Subject : Foundation Course
Paper : I
Paper Name : (Hindi Language & Moral Values) – III
Paper code : FC301-T

Particulars

न्दपज.ए	हिन्दी भाषा मेरे सहयात्री (यात्रा वृतांत) – अमृतलाल बेगड़ । मध्यप्रदेश की लोक कलाएं (संकलित) लोकोक्तियॉ एवं मुहावरे (संकलित)
न्दपज.ब	हिन्दी भाषा जनसंचार माध्यम (प्रिन्ट, इलै, एवं सोशल मीडिया). टूटते हुए (एकांकी) – सुरेश शुक्ल चंद्र संक्षिप्तियां
न्दपज.ब्ब	हिन्दी भाषा पत्रकारिता के विभिन्न आयाम (संकलित) मध्यप्रदेश का लोक साहित्य (संकलित) पत्र लेखन – आवेदन, प्रारूपण, आदेश परिपत्र ज्ञापन, अनुस्मारक (संकलित)
न्दपज.प्ट	हिन्दी भाषा राजभाषा, हिन्दी (संकलित) हिन्दी की संवैधानिक एवं व्यावहारिक स्थिति दूरभाष और मोबाइल (संकलित) हिन्दी की शब्द सम्पदा (संकलित) अनुवाद : अर्थ प्रकार एवं अभ्यास
न्दपज.ट	नैतिक मूल्य विश्व के प्रमुख धर्म एवं महत्वपूर्ण विशेषताएं (हिन्दू धर्म, जैन धर्म, बौद्ध धर्म, सिक्ख धर्म, ईसाई धर्म, इस्लाम धर्म) सत्य के साथ मेरे प्रयोग (महात्मा गाँधी की आत्म कथा का संक्षिप्त संस्करण)

Class : B.A./B.Sc./B.Com./BBA/BCA/B.A --- **III Year**
Subject : Foundation Course
Paper : II
Paper Name : English Language-III
Paper code : FC302-T

Particulars

Unit-I	Stopping by Woods on a snowy Evening: Robert Frost. Cherry Tree : Ruskin Bond. The Axe : R.K. Narayan. The Selfish Giant : Oscar Whilde On The Rule of the Road : A.G. Gardiner. The song of Kabir : Translated by Tagore
Unit-II	Basic Language Skills : Transformation of sentences, Direct-Indirect Speech, Active. Passive Voice, Confusing Words, Misused words, Similar words with different meaning.
Unit-III	Report Writing, Narration Skills, Narration of events and situations.
Unit-IV	Drafting of E-mails.
Unit-V	Drafting CV.

Class : B.A./B.Sc./B.Com./BBA/BCA/B.A --- **III Year**
Subject : Foundation Course
Paper : III
Paper Name : (Basics of Computer & Information technology)
Paper code : FC303-T

Particulars

Unit-I	**Introduction to Computer :** Basic Organization of Computer system: Block diagram & Functions (Central Processing Unit, Input/ Output Unit, Storage Unit); Characteristics: Capabilities & Limitations. Types of Computing Devices: Desktop Laptop & Notebook smart-Phone, Tablet PC, Server, Workstation & their Characteristics. Primary Memory & Their Types: RAM, ROM, PROM, EPROM EEPROM, Cache Memory. **Peripheral Devices :** Input Devices : Keyboard Mouse, Trackball, Joystick, Digitizer or Graphic tablet, Scanners, Digital Camera, Web Camera, MICR, OCR,OMR, Bar-code Reader, Voice Recognition device Light pen & Touch Screen. Output Devices : Display Devices (CRT, TFT, LCD, LED, Multimedia Projectors): Video Standard : VGA, SVGA, XGA etc, Impact Printers (Daisy Wheel, Dot Matrix & Line Printer); Non impact printer (Inkjet, Laser, Thermal); **Storage Devices :** Magnetic Tape, Cartridge, Data Drives, Hard Disk Drives (Internal & External), Floppy Disks, CD, VCD, CD-RW, Zip Drive DVD, DVD-RW, USB Flash Drive, Blue Ray Disc & Memory cards.
Unit-II	**Operating System (OS)**
	Dos Basic: FAT, File & Directory Structure and naming rules, Booting process, DOS system files, internal & External Dos Commands. Windows Basics (only elementary ideas): Windows 7 & 8: Desktop, Control Panel: saving, remaining, moving copying and searching files & folders, restoring from recycle Bin. Creating shortcut, Establishing Network Connections.
Unit-III	**MS Word** Text editing and formatting using Word file in various file formats: Previewing documents, Printing document to file / page: Protecting document Editing of selected text, Inserting Deleting and Moving text. Formatting documents: page Layout, Paragraph format, Aligning text and Paragraph, Borders and Shading, Headers and Footers.
Unit-IV	**MS Power Point & MS Excel :** Creating presentation using slide master and template in various themes & variants. Working with slide, move, copy, delete, duplicate, slide layouts, presentation views. Format menu: Font, Paragraph, drawing & editing. Printing presentation: Print slides, notes, handouts and outlines. Saving presentation in different file formats. Workbook & worksheet: Entering data into worksheet (General, Number, Currency, Date, Time, Text, Accounting etc.); Renaming, Copying, Inserting, deleting & protecting worksheet.

	Working with Row & Column (Inserting, deleting, Pasting, Resizing & Hiding), Cell & Cell formatting, and Concept of range.
Unit-V	Internet: World Wide Web Dial up connectivity, leased line, VSAT, Broad Band, WI-FI, URL, Domain name, Web Dial up Browser (Internet Explorer, Firebox, Google Chrome, opera, UC Browser etc.) Search Engine (Google, Bing, Ask etc); Website: Static & Dynamic; Difference between website & Portal. E-mail: Account opening. Sending & Receiving Mails, Managing Contacts & Folders.
	E-mail, Internet & Social Networking Ethics. Types of viruses & antivirus. Computer security Issues & its protection through firewall & antivirus. Making secured online transactions.

TextBoks :

1. PC Software for Windows by R.K. Taxali.
2. Fundamental of Computers by P.K. Sinha.
3. Computer Today by Suresh K. Basandra
4. Computer fundamental s and Architechture by B.Ram.
5. Internet Security by Kenneth Einar Himma, 2007.
6. Internet Security Secrets by John R. Vacca, 2007.

BAIM301-T
Reinforcement Learning Applications

UNIT-I **Marks -10**

Introduction to Reinforcement Learning (RL):- Overview of reinforcement learning, Components of an RL system (Agent, Environment, Actions, Rewards), Markov Decision Processes (MDPs), Exploration vs. Exploitation trade-off.

UNIT-II **Marks -10**

Deep Q-Networks (DQN):- Introduction to deep Q-learning, DQN architecture and training process, Experience replay and target networks, Applications of DQN in gaming and control tasks.

UNIT-III **Marks -10**

Policy Gradient Methods:- Policy gradients and REINFORCE algorithm, Actor-Critic methods, Advantage Actor-Critic (A2C) and Proximal Policy Optimization (PPO), Applications of policy gradient methods in robotics and autonomous systems.

UNIT-IV **Marks -10**

Model-Based Reinforcement Learning:- Model-based vs. model-free RL, Dynamics models and planning, Model-based RL algorithms (e.g., Model Predictive Control, Dyna-Q), Applications of model-based RL in healthcare and finance.

UNIT-V **Marks -10**

Advanced Topics in Reinforcement Learning Applications:-Multi-agent reinforcement learning,Hierarchical RL and options frameworks, Transfer learning and meta-learning, Case studies and real-world applications in diverse domains.

Reference Books:

"Reinforcement Learning: An Introduction" by Richard S. Sutton and Andrew G. Barto
"Deep Reinforcement Learning Hands-On" by Maxim Lapan

BAIM301-P Reinforcement Learning LAB

Q-Learning

- **Description**: Implement the Q-learning algorithm to learn action values for an agent in a discrete state space environment.
- **Practical Activities**: Writing programs to solve classic RL problems such as the grid world or mountain car using Q-learning.

2. Deep Q-Networks (DQN)

- **Description**: Train a neural network to approximate the Q-function for handling high-dimensional state spaces.
- **Practical Activities**: Implementing DQN algorithm using deep learning frameworks like TensorFlow or PyTorch to solve Atari games or robotic control tasks.

3. Policy Gradient Methods

- **Description**: Implement policy gradient methods such as REINFORCE or Actor-Critic for learning policy directly.
- **Practical Activities**: Writing programs to train agents using policy gradient methods to play games, control robotic arms, or navigate complex environments.

4. Proximal Policy Optimization (PPO)

- **Description**: Learn to implement the PPO algorithm, which addresses some of the challenges of policy gradient methods.
- **Practical Activities**: Implementing PPO algorithm to train agents for continuous control tasks or complex environments with sparse rewards.

5. Model-Based RL

- **Description**: Explore methods for learning a model of the environment to facilitate planning and decision-making.
- **Practical Activities**: Implementing model-based RL algorithms like Dyna-Q or Monte Carlo Tree Search (MCTS) for learning a model and using it for planning.

6. Multi-Agent RL

- **Description**: Learn to train multiple agents to collaborate or compete in a shared environment.
- **Practical Activities**: Writing programs to implement multi-agent RL algorithms like multi-agent DDPG or QMIX to solve coordination problems or competitive games.

7. Exploration Strategies

- **Description**: Explore techniques for balancing exploration and exploitation to discover optimal policies efficiently.
- **Practical Activities**: Implementing exploration strategies like ε-greedy, softmax, or UCB1 in RL algorithms to encourage exploration while learning.

8. Transfer Learning in RL

- **Description**: Learn to transfer knowledge from one task to another to accelerate learning in new environments.
- **Practical Activities**: Implementing transfer learning techniques like fine-tuning, distillation, or domain adaptation in RL algorithms to improve sample efficiency.

9. Continuous Action Spaces

- **Description**: Handle continuous action spaces using methods like actor-critic architectures or deterministic policy gradients.
- **Practical Activities**: Implementing algorithms like DDPG, SAC, or TD3 to solve continuous control tasks like robotic manipulation or locomotion.

10. RL in Real-World Applications

- **Description**: Apply RL techniques to solve real-world problems in domains like robotics, finance, healthcare, or recommendation systems.
- **Practical Activities**: Designing and implementing RL solutions for specific applications, such as autonomous driving, algorithmic trading, personalized medicine, or content recommendation.

BAIM302-T
Data Ethics and Privacy

UNIT-I **Marks -10**

Introduction to Data Ethics and Privacy:- Understanding the concepts of data ethics and privacy, Importance and significance of ethical considerations in data science, Historical context and evolution of data ethics and privacy regulations.

UNIT-II **Marks -10**

Ethical Issues in Data Collection and Processing:- Ethical considerations in data collection practices, Privacy implications of data processing techniques (e.g., machine learning, data mining), Case studies illustrating ethical dilemmas in data collection and processing.

UNIT-III **Marks -10**

Legal and Regulatory Frameworks:- Overview of data privacy laws and regulations (e.g., GDPR, CCPA), Ethical guidelines and professional codes of conduct in data science,

Impact of legal and regulatory compliance on data ethics and privacy practices.

UNIT-IV **Marks -10**

Ethical Data Analysis and Interpretation:- Ethical considerations in data analysis and interpretation, Bias and fairness issues in algorithmic decision-making, Techniques for ensuring transparency and accountability in data analysis.

UNIT-V **Marks -10**

Privacy-Enhancing Technologies and Practices:-Overview of privacy-enhancing technologies (PETs),Best practices for privacy-preserving data sharing and collaboration, Emerging trends and developments in data ethics and privacy.

Reference Book:

"Data and Ethics: A Guidebook for Data Scientists" by Mike Loukides and Hilary Mason.
"Ethics of Big Data: Balancing Risk and Innovation" by Kord Davis.
"Privacy, Big Data, and the Public Good: Frameworks for Engagement" by Julia Lane, Victoria Stodden, Stefan Bender, and Helen Nissenbaum.
"Weapons of Math Destruction: How Big Data Increases Inequality and Threatens Democracy" by Cathy O'Neil.
"Privacy in Context: Technology, Policy, and the Integrity of Social Life" by Helen Nissenbaum.

BAIM303-T
Model Deployment and Serving

UNIT-I **Marks -10**

Introduction to Model Deployment:- Overview of model deployment and serving, Importance and challenges of deploying machine learning models, Introduction to deployment architectures and technologies.

UNIT-II **Marks -10**

Containerization and Orchestration:- Introduction to containerization technologies (e.g., Docker), Container orchestration with Kubernetes, Deploying machine learning models in containers.

UNIT-III **Marks -10**

Model Deployment Platforms:-

Overview of model deployment platforms (e.g., TensorFlow Serving, PyTorch Serve), Hands-on experience with deploying models using platform-specific tools, Comparison of different deployment platforms and their features.

UNIT-IV **Marks -10**

Scalability and Performance Optimization:- Techniques for scaling model deployments to handle large workloads, Performance optimization strategies for deployed models, Monitoring and debugging deployed models.

UNIT-V **Marks -10**

DevOps for Machine Learning:- Introduction to DevOps principles and practices, DevOps tools and techniques for machine learning model deployment, Continuous integration and continuous deployment (CI/CD) pipelines for machine learning projects.

Reference Book:

"Building Machine Learning Powered Applications: Going from Idea to Product" by Emmanuel Ameisen
"Kubernetes: Up and Running: Dive into the Future of Infrastructure" by Kelsey Hightower, Brendan Burns, and Joe Beda
"Practical Deep Learning for Cloud, Mobile, and Edge: Real-World AI & Computer-Vision Projects Using Python, Keras & TensorFlow" by Anirudh Koul, Siddha Ganju, and Meher Kasam
"Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems" by Martin Kleppmann
"Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation" by Jez Humble and David Farley.

BAIM303-P Model Deployment LAB

Containerization with Docker

- **Description**: Learn to containerize machine learning models for easy deployment and scalability.
- **Practical Activities**: Dockerizing machine learning models using Dockerfiles and deploying them as Docker containers.

2. RESTful API Development

- **Description**: Develop RESTful APIs to expose machine learning models for consumption by other applications.
- **Practical Activities**: Building APIs using frameworks like Flask or FastAPI to serve predictions from machine learning models.

3. Model Serving with TensorFlow Serving

- **Description**: Deploy TensorFlow models for inference using TensorFlow Serving.
- **Practical Activities**: Setting up TensorFlow Serving and deploying TensorFlow models as RESTful APIs for scalable model serving.

4. Model Deployment on Kubernetes

- **Description**: Learn to deploy machine learning models on Kubernetes clusters for scalability and high availability.
- **Practical Activities**: Deploying machine learning models as microservices on Kubernetes using tools like Kubernetes Deployment or Helm charts.

5. Serverless Model Deployment

- **Description**: Deploy machine learning models as serverless functions for on-demand scaling and cost efficiency.
- **Practical Activities**: Deploying machine learning models on serverless platforms like AWS Lambda, Google Cloud Functions, or Azure Functions.

6. Model Deployment on Edge Devices

- **Description**: Deploy machine learning models on edge devices for real-time inference at the edge.
- **Practical Activities**: Optimizing machine learning models for deployment on edge devices like Raspberry Pi or NVIDIA Jetson.

7. Model Deployment with AWS SageMaker

- **Description**: Utilize AWS SageMaker for end-to-end model deployment and management on AWS.
- **Practical Activities**: Deploying machine learning models on AWS SageMaker and creating scalable endpoints for inference.

8. Model Deployment with Azure ML

- **Description**: Use Azure Machine Learning for deploying and managing machine learning models on Azure.
- **Practical Activities**: Deploying machine learning models on Azure ML as web services or containers and consuming them from other applications.

9. Model Deployment with TensorFlow.js

- **Description**: Deploy machine learning models for inference in web browsers or Node.js environments using TensorFlow.js.
- **Practical Activities**: Converting TensorFlow models to TensorFlow.js format and deploying them for inference in web applications.

10. Continuous Integration and Deployment (CI/CD) for Models

- **Description**: Implement CI/CD pipelines for automating model deployment and monitoring.
- **Practical Activities**: Setting up CI/CD pipelines using tools like Jenkins, GitLab CI/CD, or GitHub Actions to automate model deployment workflows.

BAIM304-T
Bayesian Machine Learning

UNIT-I **Marks -10**

Introduction to Bayesian Machine Learning:- Overview of Bayesian inference and probabilistic modeling, Introduction to Bayesian decision theory, Comparison of Bayesian and frequentist approaches in machine learning.

UNIT-II **Marks -10**

Bayesian Linear Regression:- Bayesian formulation of linear regression, Prior specification and posterior inference, Comparison with traditional linear regression techniques.

UNIT-III **Marks -10**

Bayesian Classification:- Bayesian formulation of classification algorithms (e.g., Naive Bayes, Bayesian logistic regression), Bayesian model averaging and ensemble methods, Applications of Bayesian classification in practice.

UNIT-IV **Marks -10**

Bayesian Non-parametric Methods:- Introduction to Dirichlet Process and Gaussian Process models, Bayesian inference in non-parametric settings, Applications of Bayesian non-parametric methods in clustering and regression.

UNIT-V **Marks -10**

Advanced Topics in Bayesian Machine Learning:- Bayesian optimization and decision-making,, Variational inference and Monte Carlo methods, Emerging trends and applications in Bayesian machine learning.

Reference Book:

"Bayesian Methods for Hackers: Probabilistic Programming and Bayesian Inference" by Cameron Davidson-Pilon. "Pattern Recognition and Machine Learning" by Christopher M. Bishop. "Probabilistic Graphical Models: Principles and Techniques" by Daphne Koller and Nir Friedman. "Gaussian Processes for Machine Learning" by Carl Edward Rasmussen and Christopher K. I. Williams, "Bayesian Reasoning and Machine Learning" by David Barber.

BAIM305-T

Big Data Handling and Distributed Computing

UNIT-I **Marks -10**

Introduction to Big Data and Distributed Computing:- Definition and characteristics of big data, Challenges in handling big data, Introduction to distributed computing, Distributed systems architectures.

UNIT-II **Marks -10**

MapReduce and Hadoop:- Overview of MapReduce programming model, Introduction to Hadoop ecosystem, Hadoop Distributed File System (HDFS), MapReduce framework and its components.

UNIT-III **Marks -10**

Spark and In-Memory Computing:- Introduction to Apache Spark, RDDs (Resilient Distributed Datasets) and data transformations, Spark SQL and DataFrame API, Spark Streaming and real-time processing.

UNIT-IV **Marks -10**

Distributed Data Storage:- Distributed storage systems (e.g., Cassandra, HBase),

CAP theorem and its implications, Consistency models and trade-offs, Scalability and fault tolerance in distributed storage.

UNIT-V **Marks -10**

Advanced Topics in Distributed Computing:- Stream processing frameworks (e.g., Apache Flink, Kafka Streams), Containerization and orchestration (e.g., Docker, Kubernetes),

Machine learning at scale, Case studies and practical applications.

Reference Books:

"Big Data: Principles and best practices of scalable real-time data systems" by Nathan Marz and James Warren

"Hadoop: The Definitive Guide" by Tom White

"Spark: The Definitive Guide" by Bill Chambers and Matei Zaharia

"HBase: The Definitive Guide" by Lars George.

BAIM306-T
Python programming

UNIT-I **Marks -10**

Introduction to Python:- Overview of Python programming language, Basic syntax, data types, and control structures, Introduction to Python scripting and development environments.

UNIT-II **Marks -10**

Functions and Modules:- Defining and calling functions in Python, Scope and namespaces, Working with modules and packages.

UNIT-III **Marks -10**

Data Structures and Algorithms in Python:- Introduction to data structures (lists, tuples, dictionaries, sets), Implementing common algorithms in Python (sorting, searching, recursion), Analyzing algorithm efficiency (big O notation).

UNIT-IV **Marks -10**

Object-Oriented Programming (OOP) in Python:- Understanding OOP concepts (classes, objects, inheritance, polymorphism), Implementing OOP principles in Python, Design patterns and best practices in Python OOP.

UNIT-V **Marks -10**

Advanced Python Topics:- File I/O operations, Exception handling and debugging techniques, Introduction to libraries and frameworks (e.g., NumPy, Pandas, Flask).

Reference Book:

"Python Crash Course" by Eric Matthes, "Fluent Python: Clear, Concise, and Effective Programming" by Luciano Ramalho, "Problem Solving with Algorithms and Data Structures using Python" by Bradley N. Miller and David L. Ranum, "Python 3 Object-Oriented Programming" by Dusty Phillips, "Python for Data Analysis" by Wes McKinney.

BAIM306-P Python LAB

- **Description**: Learn how to extract data from websites using libraries like BeautifulSoup or Scrapy.
- **Practical Activities**: Writing scripts to scrape data from websites, parse HTML/XML documents, and save the extracted data to a file or database.

2. Data Analysis with Pandas

- **Description**: Understand data manipulation and analysis using the Pandas library.
- **Practical Activities**: Loading and exploring datasets, performing data cleaning, filtering, aggregation, and visualization using Pandas and Matplotlib/Seaborn.

3. Machine Learning with Scikit-learn

- **Description**: Explore machine learning algorithms and techniques using the Scikit-learn library.
- **Practical Activities**: Building and training machine learning models for classification, regression, clustering, and dimensionality reduction tasks using Scikit-learn.

4. Web Development with Django or Flask

- **Description**: Learn web development by building web applications using Django or Flask frameworks.
- **Practical Activities**: Creating web applications with user authentication, CRUD operations, and RESTful APIs using Django or Flask.

5. GUI Development with Tkinter or PyQt

- **Description**: Develop graphical user interfaces (GUIs) for desktop applications using Tkinter or PyQt libraries.
- **Practical Activities**: Designing and building GUI applications with buttons, labels, input fields, and other widgets using Tkinter or PyQt.

6. Data Visualization with Matplotlib and Seaborn

- **Description**: Visualize data and create plots for exploration and presentation.
- **Practical Activities**: Creating line plots, scatter plots, histograms, heatmaps, and other visualizations using Matplotlib and Seaborn libraries.

7. Natural Language Processing (NLP) with NLTK or SpaCy

- **Description**: Process and analyze textual data using natural language processing libraries.
- **Practical Activities**: Tokenization, stemming, lemmatization, part-of-speech tagging, named entity recognition, and sentiment analysis using NLTK or SpaCy.

8. Image Processing with OpenCV

- **Description**: Perform image processing tasks such as object detection, image segmentation, and feature extraction.
- **Practical Activities**: Reading and displaying images, applying filters, detecting edges, detecting and recognizing objects in images using OpenCV.

9. Data Science Projects

- **Description**: Work on data science projects to solve real-world problems using Python.
- **Practical Activities**: Identifying and defining a problem, collecting and exploring data, performing analysis, building models, and presenting insights using Jupyter Notebooks or other tools.

10. Automation Scripts

- **Description**: Develop scripts to automate repetitive tasks and improve productivity.
- **Practical Activities**: Writing scripts for file manipulation, data processing, web automation, system administration, or any other task that can be automated using Python.