SCHEME AND SYLLABUS

FOR

B.TECH

IN

ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

Offered by the Department of Computer Engineering,
National Institute of Technology, Kurukshetra

w.e.f. 2023-24 (1st & 2nd year)

VISION AND MISSION OF THE INSTITUTE

VISION

To be a role-model in technical education and research, responsive to global challenges.

MISSION

To impart technical education that develops innovative professionals and entrepreneurs and to undertake research that generates cutting-edge technologies and futuristic knowledge, focusing on the socio-economic needs.

VISION AND MISSION OF THE DEPARTMENT

VISION

To address societal needs and global industry challenges in the field of Computer & IT with state-of-art education & research.

MISSION

M-1: To create a platform for education, research and development by providing sound theoretical knowledge and practical skills in Computer Engineering and Information Technology.

M-2: To produce motivated professional technocrats capable of generating solutions for industry and society.

M-3: To develop the ability to work ethically at individual and team level and be responsive towards socio-economic needs.

VISION AND MISSION OF THE PROGRAM

VISION

To disseminate state-of-the-art education to develop competent professionals in Computer Engineering with capability to serve the global society**.**

MISSION

To educate and train manpower engaged in cutting-edge research by offering latest in the field of Computer Engineering for sustainable development of society.

Scheme for B.Tech. in
Artificial Intelligence and Machine Learning
w.e.f. Academic Year 2023-24

SEMESTER - I

Sr. No.	Course Code	Course Title	L_T_P	Credits	Contact Hrs
1.	-	Communication Skills in English OR Financial Education	2 0 2	3	4
2.	-	Differential Calculus and Differential Equations	3 0 0	3	3
3.	-	Engineering Physics	3 0 2	4	5
4.	CSIC 101	Problem solving & Programming Skills I (For CO, IT & AI&ML, AI&DS, M&C)	3 0 2	4	5
5.	CSIC 103	Problem solving & Programming Skills II (for EE, ECE, CE, ME, PIE, IIOT)	3 0 2	4	5
6.	-	Energy & Environmental Science	2 0 2	3	4
7.	-	Engineering Practices (P)	1 0 3	2	4
8.	-	AU1	2 0 0	2	2
9.	-	Sports	0 0 2	1	2
10.	-	NCC/NSS/Yoga	0 0 2	1	2
Total				**21**	**31**

SEMESTER - II

Sr. No.	Course Code	Course Title	L_T_P	Credits	Contact Hrs
1.	-	Economics for Engineers OR Business Studies	3 0 0	3	3
2.	-	Integral Calculus and Difference Equations	3 0 0	3	3
3.	CSIC 100	Digital System Design	3 0 0	3	3
4.	CSIC 102	Web Design*	1 0 3	2	4
5.	CSIC 104	Programming Using Python	3 0 2	4	5
6.	AIPC 100	Data Structures	3 0 2	4	5
7.	-	AU2	2 0 0	2	2
8.	-	Sports	0 0 2	1	2
9.	-	NCC/Yoga/Clubs/NSS	0 0 2	1	2
Total				**21**	**29**

*** Treated as a practical course (not integrated), evaluation will be as per practical course.**

14-3-6-24

Scheme for
B Tech in Artificial Intelligence and Machine Learning
w.e.f. Academic Year 2024-25

SEMESTER-III

Course Category	SN	Course Code	Course Title	Load Per Week			Credits	Contact Hrs
				L	T	P		
IC	1.	MAIC ***	Statistical Distribution	3	0	0	3	3
	2.	MAIC ***	Optimization Techniques	3	0	0	3	3
PC	3.	AIPC 201	Knowledge Representation and Problem Searching	3	0	2	4	5
	4.	AIPC 203	Design and Analysis of Algorithms	3	0	2	4	5
	5.	AIPC 205	Database Management Systems	3	0	2	4	5
NC	6.	SWNC 101	NCC/ Sports /Yoga	0	0	2	1**	2
	7.	SWNC 102	NSS /Club/Technical Societies	0	0	2		2
			Total Credits				**18**	**25**

SEMESTER -IV

Course Category	SN	Course Code	Course Title	Load Per Week			Credits	Contact Hrs
				L	T	P		
IC	1	MAIC ***	Applied Linear Algebra	3	0	0	3	3
PC	2	AIPC 200	Decision Making Using AI	3	0	2	4	5
	3	AIPC 202	Operating Systems	3	0	2	4	5
	4	AIPC 204	Neural Networks	3	0	2	4	5
	5	AIPC 206	Fuzzy Systems and Evolutionary Computing	3	0	2	4	5
NC	6	SWNC 101	NCC/ Sports /Yoga	0	0	2	1**	2
	7	SWNC 102	NSS /Club/Technical Societies	0	0	2		2
Total Credits							**19**	**27**

** Continuous Evaluation Model as per guidelines and the credit to be awarded at the end of 6th Semester based on Cumulative performance up to 6th Semester.

14-6-24

Course Code	**:**	**CSIC 101**
Course Title	**:**	**Problems Solving and Programming Skills-I (for CO, IT & AI &ML)**
Number of Credits & L/T/P Scheme	**:**	**4 and 3/0/2**
Prerequisites	**:**	
Course Type	**:**	**IC**

Course Learning Objectives:
The objective of this course is to provide fundamentals of problem-solving using C language programming.

Course Content:
Unit 1 Programming Fundamentals & Control Statements:
Block Diagram of Computer, Hardware vs software, concept of operating system and compiler, Introduction to C programming, basic programming using input and output operators and expressions, programming using if and if-else, Programming using looping-for, while, do-while; use of switch and break.

Unit 2 Arrays based Programming:
Defining and processing 1-D and 2-D Arrays for Problem Solving and Strings.

Unit 3 Modular programming using Functions:
Structured Programming, Defining and calling a function, modular programming using functions, passing arguments and arrays to functions, functions of void and returning values.

Unit 4 Programming using pointers, structures and unions:
Pointers in C: Pointer declaration, Passing Pointer to functions, pointers vs arrays, dynamic memory allocation. Structures and Unions, Programming Using Array of Structures and Unions, Memory Requirements for Unions.

Reference Books:
1. Byron S. Gottfried, Programming with C Language, Schaum Series, Tata McGraw Hill, 2015.
2. E Balaguruswamy, Programming with C, Tata McGraw Hill, 2015.
3. Kernighan & Richie, C Programming, Prentice Hall of India, 2002.

Course Outcomes:
1. Understand the use of software and programming for problem solving.
2. Learn programming using simple concepts of input, output and control statements.
3. Use arrays, functions, strings, structures and pointers for problem solving.

Course Code	**:**	**CSIC 102**
Course Title	**:**	**Digital System Design**
Number of Credits & L/T/P Scheme	**:**	**3 and 3/0/0**
Prerequisites	**:**	
Course Type	**:**	**IC**

Course Learning Objectives:

1. Awareness of intricate design details of components in any digital system.
2. Knowledge of number system and conceptual understanding of different codes.
3. Design fundamentals of computing machinery.
4. Introduction of computational automation process.

Course Content:

Unit 1 Number Systems and Coding Schemes:

Number Systems and Codes Introduction to the positional number system, signed magnitude numbers, floating point numbers, binary arithmetic: addition, subtraction, multiplication and division, Base conversion, conversion formulas with examples, one's and two's compliment arithmetic, Computer codes
– BCD codes, gray codes, excess-3 codes, parity checks, Hamming and alphanumeric codes.

Unit 2 Combinational Logic:

Design Introduction, standard representations for logical functions, Karnaugh map representation, simplification of logical functions using K-map, minimization of logical functions specified in minterms/maxterms or Truth Table, minimization of logical functions not specified in minterms/maxterms, Don't care conditions, design examples, Ex-or and Ex-nor simplification of Kmaps, five and six-variable K-maps, QM method, MEV method, Introduction of multiplexers and their use in combinational logic design, demultiplexers/decoders and their use in combinational logic design, adders and their use as subtractors, digital comparators, parity generators/checkers, code converters, priority encoders.

Unit 3 Synchronous Sequential Circuits:

Introduction, FSM model, memory elements and their excitation functions. Synthesis of synchronous sequential circuits, capabilities and limitation of FSM, state equivalence and minimization, simplification of incompletely specified machines, registers and counters, RAM design, ROM design and programmablelogic array.

Unit 4 Asynchronous Sequential Circuits:

Fundamental mode and Pulse mode Circuits Analysis and Design.

Books:

1. M. Morris Mano and Michael D. Ciletti: Digital Logic Design, Sixth Edition, Pearson Education.
2. R.P. Jain: Modern Digital Electronics, Fifth Edition, TMH.
3. Z Kohavi and Niraj. K. Jha: Switching And Finite Automata Theory, Third Edition, Cambridge UniversityPress.
4. Kumar A. Anand: Fundamentals of Digital Circuits, Fourth Edition, PHI.
5. James Bignell and Robert Donovan: Digital Electronics, Fifth Edition, Cengage Learning.

Course Outcomes:

1. Clarity of application of different number system and coding schemes.
2. Proficiency in design and analysis of combinational and sequential circuits.
3. Circuit level understanding of computer addressing and memory layouts.
4. Application of digital circuits for design of finite automaton.

Course Code	**:**	**CSIC 104**
Course Title	**:**	**Engineering Graphics (Web Design)**
Number of Credits & L/T/P Scheme	**:**	**2 and 1/0/3**
Prerequisites	**:**	
Course Type	**:**	**IC**

Course Learning Objectives:

1. Introduction and brief history of World Wide Web (WWW).
2. Web essentials: HTML, XHTML, CSS.
3. Addressing web standards, client requirements and principles of web page design.
4. Introduction of Web architecture.

Course Content:

Unit 1 Introduction: Introduction to world wide web, Web Browsers, Web Servers, Hypertext Transfer Protocol, URLs, Domain Names, Internet Service Provider, Basic steps for Developing Website, Choosing the Contents, Planning and Designing Web Site, Creating a Website, Web Publishing, Hosting Site, Types of hosting packages, Five Golden rules of web designing.

Unit 2 Web essentials and standards: Clients, servers, introduction to Markup languages, scripting languages, Introduction to elements of HTML, XHTML and CSS, Introduction to Document object model (DOM), working with text, list, tables, frames, hyperlinks, Images, forms and controls. CSS properties, Id and Class, Box Model.

Unit 3 Javascript: Javascript as programming language, Data types, Values, Variables, Expressions and Operators. JavaScript Statements, loops, arrays, strings, methods, Defining and Invoking functions and their closure, random functions and maths library, representing dates, Pattern Matching and Regular Expressions, difference between server side and client side javascript, embedding javascript in HTML, hiding HTML elements, showing hidden HTML elements. DOM and event handling, error handling, mouse, text, and keyboard events and cookies.

Unit 4 XML: XML: Introduction – benefits of XML, well formed XML documents, XML syntax, XML declaration ,XML schema , XML with CSS, Document Type Definition (DTD),creating DTD – Types(internal DTD, external DTD),XSL.

Reference Books:

1. Thomas A Powell, HTML: The Complete Reference, Tata McGraw Hill Publications.
2. Scott Guelich, Shishir Gundavaram, Gunther Birzniek; CGI Programming with Perl 2/e,O'Reilly
3. Doug Tidwell, James Snell, Pavel Kulchenko; Programming Web Services with SOAP, O'Reilly
4. Robert. W. Sebesta, "Programming the World Wide Web", Fourth Edition, Pearson Education,2007.
5. Yong, XML Step by Step, PHI.
6. Chris Bales, "Web programming- Building Internet Application".
7. Deitel, Deitel, Goldberg, "Internet & World Wide Web How To Program", Third Edition,Pearson Education, 2006.
8. Marty Hall and Larry Brown, "Core Web Programming" Second Edition, Volume I and II,Pearson Education, 2001.
9. Bayross Ivan, "Web Enabled Commercial Applications Development using HTML, Javascript, DHTML & PHP", BPB Publication, 2005.

Course outcomes

1. Knowledge of basic principles of web site design.
2. Design proficiency of websites adhering to current web standards (HTML, XML, CSS).
3. Knowledge of various scripting languages.

Course Code	**:**	**CSIC 106**
Course Title	**:**	**Programming using Python**
Number of Credits & L/T/P Scheme	**:**	**4 and 3/0/2**
Prerequisites	**:**	
Course Type	**:**	**IC**

Course Learning Objectives:

1. Building robust applications using Python programming.
2. Building multithreaded, platform-independent and GUI based python applications for business problems.

Course Content:

Unit 1 The concept of data types:

Variables, Assignments; Immutable Variables; Numerical Types; Arithmetic Operators And Expressions; comments in the program; understanding error messages; Conditions, boolean logic, logical operators; ranges; Control statements: if-else, loops (for, while); short-circuit (lazy) evaluation; Strings and text files; manipulating files and directories, os and sys modules; text files: reading/writing text and numbers from/to a file; creating and reading a formatted file (csv or tab-separated); String manipulations: subscript operator, indexing, slicing a string.

Unit 2 Lists, tuples, and dictionaries:

Basic list operators, replacing, inserting, removing an element; searching and sorting lists; dictionary literals, adding and removing keys, accessing and replacing values; traversing dictionaries; Design with functions: hiding redundancy, complexity; arguments and return values; formal vs actual arguments, named arguments.

Unit 3 Simple Graphics and Image Processing:

"turtle" module; simple 2d drawing - colors, shapes; digital images, image file formats, image processing: Simple image manipulations with 'image' module (convert to bw, greyscale, blur, etc). Classes and OOP: classes, objects, attributes and methods; defining classes; design with classes, data modeling; persistent storage of objects; inheritance, polymorphism, operator overloading (_eq_, _str_, etc); abstract classes; exception handling, try block

Unit 4 Graphical user interfaces:

Event-driven programming paradigm; tkinter module, creating simple GUI; buttons, labels, entry fields, dialogs; widget attributes - sizes, fonts, colors layouts, nested frames.

Reference Books:

1. T.R. Padmanabhan, Programming with Python, Springer, 1st Ed., 2016.
2. Kenneth Lambert, Fundamentals of Python: First Programs, Cengage Learning,, 1st Ed., 2012.

Course outcomes:

1. Programming ability for solving simple business problems.
2. Design of robust and multithreaded python applications.
3. Familiarity of simple GUI interfaces.

Course Code	**:**	**AIPC 100**
Course Title	**:**	**Data Structures**
Number of Credits & L/T/P Scheme	**:**	**4 and 3/0/2**
Prerequisites	**:**	**Problems Solving and Programming Skills-I**
Course Type	**:**	**PC**

Course Learning Objectives

1. This course introduces the concept of Data Structures used in various computer science applications
2. The students are introduced to understand and efficiently apply various data structures such as stacks, queues, linked lists, trees and graphs for solving various computing problems using C programming language.

Course Content

Unit 1 Pointers & File Handling:

Revision of Pointers and Dynamic Memory, Files and related operations.
Searching techniques: Linear and Binary, Sorting techniques: Selection, Bubble, Insertion, Merge sort, Quicksort.

Unit 2 Simple Data Structures

Arrays based Linear Data Structures: Array storage, sparse arrays; Transpose, addition, and multiplication of sparse matrices, Stacks and Queues and their applications, multiple stacks and queues in an array.

Unit 3 Linked Data Structures

Singly, Doubly & Circular Linked Lists; representation, operations and applications, linked stacks and queues. linked lists based polynomial addition.

Unit 4 Advanced Data Structures

Trees, Basic concepts and definitions of a tree and binary tree and associated terminology, Binary tree traversal techniques, some more operations on binary trees, Heaps and heapsort.

Reference Books:

1. E Horowitz and S. Sahni: Fundamentals of Data Structures in C, Second Edition, Universities Press, Hyderabad.
2. R.L. Kruse: Data Structures & Program Design in C, PHI.
3. D.F. Knuth: The art of Computer Programming Vol-1, Narosa Publications, 1985.
4. Byron S. Gottfried & J K Chhabra: Theory and Problems of Programming with C Language, Schaum's Outlines Series, TMH, 2005.

Course Outcomes

1. Develop skill to identify and determine the usage of various data structures, operations, associated algorithms and implement their applications.
2. Apply knowledge of pointers, memory allocation and string handling for solving programming problems.
3. Understand the concept of trees and graphs, their implementation and applications.
4. Able to implement standard algorithms for searching and sorting.
5. Analyze efficiency of different algorithms using time and space complexity.

Course Code	**:**	**MAIC*** **
Course Title	**:**	**Statistical Distribution**
Number of Credits and L/T/P scheme	**:**	**3 and 3/0/0**
Prerequisites (Course code)	**:**	**Calculus**
Course Category	**:**	**IC**

Course Learning Objectives

1. To identify a random variable that describes randomness or an uncertainty in certain realistic situation
2. To learn important probability distributions like: in the discrete case, study of the Binomial and the Poisson Distributions and in the continuous case the Normal Distributions.
3. To check the relationship between two variables and also to predict behaviour of variable changes based on other variable.
4. To interpret the types of sampling, sampling distribution of means and variance, Estimations of statistical parameters.

Course Content:

Unit 1 Random Variables Single Random Variables -Discrete and Continuous, Probability distribution function, Probability mass and density functions, mathematical expectation and variance. Multiple Random variables: Discrete and Continuous, Joint probability distribution, Marginal probability density functions, conditional probability distribution function and density functions.

Unit 2 Probability Distributions Bernoulli distribution, Binomial distribution – properties, mean, variance and recurrence formula for Binomial distribution, Poisson distribution – Poisson distribution as Limiting case of Binomial distribution, properties, mean variance and recurrence formula for Poisson distribution, Normal distribution, Exponential distribution – mean, variance, median, mode and characteristics of Normal distribution.

Unit 3 Correlation and Regression Bivariate and multivariate distributions, Joint and marginal distributions, Covariance and correlation, Regression- Regression coefficients, Lines of regression.

Unit 4 Sampling and Testing of Hypothesis for Large Samples Sampling: Definitions - Types of sampling - Expected values of sample mean and variance, Standard error - Sampling distribution of means and variance. Estimation - Point estimation and Interval estimation. Testing of hypothesis: Null and Alternative hypothesis - Type I and Type II errors, Critical region - confidence interval - Level of significance, One tailed and Two tailed test.

Text Books:

1. Steven F.Arnold, Mathematical Statistics, Prentice Hall.
2. R.V. Hogg, J. W. McKean and A. T. Craig, An Introduction to Mathematical Statistics, Sixth Edition, Pearson Education

Reference Books:

1. V.K. Rohatgi and Md.E. Saleh, An Introduction to Probability and Statistics, Second Edition, Wiley.
2. E. J. Dudewicz and S. N. Mishra, John, Modern Mathematical Statistics, Wiley & Sons.

Course outcomes

1. Evaluate randomness in certain realistic situation, which can be either discrete or continuous type and compute statistical constants of these random variables.
2. Higher up thinking skills to make objective, data-driven decisions by using correlation and regression.
3. Analyze and interpret statistical inference using samples of a given size, which is taken from a population.

Course Code	**:**	**MAIC*** **
Course Title	**:**	**Optimization Techniques**
Number of Credits and L/T/P scheme	**:**	**3 and 3/0/0**
Prerequisites (Course code)	**:**	**Background in differential calculus and basic maths.**
Course Category	**:**	**IC**

Course Learning Objectives:

1. Analyze various methods of solving the unconstrained minimization problem.
2. Understand the concept of multivariable optimization technique.
3. Analyze the optimality criteria for various optimization techniques.
4. Understand the concept of single variable optimization problems.
5. Analyze solution of nonlinear programming problems..
6. Analyze various optimization methodologies.

Course Content

Unit 1 Introduction and Basic Concepts : Historical Development; Engineering applications of Optimization; Art of Modeling; Objective function; Constraints and Constraint surface; Formulation of design problems as mathematical programming problems; Classification of optimization problems; Optimization techniques – classical and advanced techniques.

Unit 2 Optimization using Calculus : Stationary points; Functions of single and two variables; Global Optimum; Convexity and concavity of functions of one and two variables; Optimization of function of one variable and multiple variables; Gradient vectors; Examples; Optimization of function of multiple variables subject to equality constraints; Lagrangian function; Optimization of function of multiple variables subject to equality constraints; Hessian matrix formulation; Eigen values.

Unit 3 Linear Programming : Standard form of linear programming (LP) problem; Canonical form of LP problem; Assumptions in LP Models; Elementary operations; Graphical method for two variable optimization problem; Examples; Motivation of simplex method, Simplex algorithm and construction of simplex tableau; Simplex criterion; Minimization versus maximization problems.

Unit 4 Dynamic Programming : Sequential optimization; Representation of multistage decision process; Types of multistage decision problems; Concept of sub optimization and the principle of optimality; Recursive equations – Forward and backward recursions; Computational procedure in dynamic programming (DP) ; Discrete versus continuous dynamic programming; Multiple state variables; curse of dimensionality in DP

Text Books:

1. S.S. Rao,"Engineering Optimization: Theory and Practice", New Age International P)Ltd., New Delhi, 2000.
2. G. Hadley,"Linear programming", Narosa Publishing House, New Delhi, 1990.
3. H.A. Taha,"Operations Research:An Introduction", 5th Edition, Macmillan, New York, 1992.
4. K. Deb,"Optimization for Engineering DesignAlgorithms and Examples",Prentice-Hall of India Pvt. Ltd., New Delhi, 1995.
5. K.Srinivasa Raju and D. Nagesh Kumar, "Multicriterion Analysis in Engineering and Management",PHI Learning Pvt. Ltd., New Delhi, India, ISBN 978-81-203-3976-7, pp.288.

Reference Books:

1. D. G. Luenberger and Y. Ye, Linear and Nonlinear Programming, 3rd Ed., Springer India, 2010.
2. M. S. Bazarra, J.J. Jarvis, and H.D. Sherali, Linear Programming and Network Flows, 4th Ed., 2010. (3nd ed. Wiley India 2008).
3. U. Faigle, W. Kern, and G. Still, Algorithmic Principles of Mathematical Programming, Kluwe, 2002.
4. D.P. Bertsekas, Nonlinear Programming, 2nd Ed., Athena Scientific, 1999.
5. M. S. Bazarra, H.D. Sherali, and C. M. Shetty, Nonlinear Programming: Theory and Algorithms, 3rd Ed., Wiley, 2006. (2nd Edn., Wiley India, 2004).

Course Outcomes

At the end of the course student will be able to:

1. Comprehend the techniques and applications of Engineering optimization.
2. Analyze characteristics of a general linear programming problem
3. Apply basic concepts of mathematics to formulate an optimization problem
4. Analyse various methods of solving the unconstrained minimization problem
5. Analyze and appreciate variety of performance measures for various optimization problems

Course Code	**:**	**AIPC 201**
Course Title	**:**	**Knowledge Representation and Problem Searching**
Number of Credits and L/T/P scheme	**:**	**4 and 3/0/2**
Prerequisites (Course code)	**:**	**Some exposure to formal languages, logic and programming**
Course Category	**:**	**PC**

Course Learning Objectives

1. Be able to model simple application domains in a logic-based language;
2. Understand the notion of a reasoning service;
3. Master the fundamentals of the reasoning algorithms underlying current systems;
4. Understand the fundamental trade-off between representation power and computational properties of a logic-based representation language;
5. Be conversant with several widely used knowledge representation languages; and
6. Understand how the theoretical material covered in the course is currently being applied in practice**.**

Course Content:

Unit-1 Problem Solving: Intelligent Agent: Sensors, Actuators, Agent Program; Solving Problems by Searching: problem-solving agent, Blind Search Strategies: Breadth-first Search, Depth-first Search (completeness, optimality, complexity); Informed Search Strategies: Greedy Best-first Search, A* Search (completeness, optimality, complexity)

Unit-2 Knowledge and Reasoning - Logical Agents: Knowledge Base, Models, and Knowledge-Based Agents; Propositional Logic Knowledge Representation Language, Syntax and Semantics; Logical Reasoning: Entailment and Inference (soundness, completeness); Propositional Theorem Proving: Validity, Satisfiability, Reduction to Absurd; MP Inference Rule, Resolution Inference Rule, Horn Form, CNF

Unit-3 Knowledge and Reasoning - First-Order Logic: Propositional Logic vs. First-Order Logic: objects, relations (unary, n-ary), functions; First-Order Logic: Syntax and Semantics (predicates, variables, quantifiers); First-Order Logic Knowledge Representation Language, Model, Interpretation; First-Order Logic Knowledge Base

Unit-4 Uncertain Knowledge and Reasoning - Quantifying Uncertainty: Acting under Uncertainty; Probability (model, atomic event, conditional), Random Variables (propositional, discrete, continuous); Syntax and Semantics: probability distribution, joint probability distribution; Inference by Enumeration, Normalization; Independence, Conditional Independence, and Bayes' Rule

Unit-5 Learning - Machine Learning: Inductive Learning; Decision Tree Learning; Unsupervised Learning; Supervised Learning

Text Books:

1. S. Russell and P. Norvig, Artificial Intelligence: A Modern Approach, 3rd edition, Prentice Hall, 2010.
2. Frank Van Harmelen, Vladimir Lifschitz, and Bruce Porter, eds. Handbook of knowledge representation. Elsevier, 2008. Ch1.

Reference Books:

1. Martin Gebser, Roland Kaminski, Benjamin Kaufmann, and Torsten Schaub. Answer set solving in practice. Synthesis Lectures on Artificial Intelligence and Machine Learning. 2012.
2. Grigoris Antoniou and Frank Van Harmelen. A semantic web primer. 3rd ed. 2012.
3. Pedro Domingos and Daniel Lowd. Markov logic: An interface layer for artificial intelligence. 2009.

Course outcomes

At the end of the course student will be able to

1. Discuss the foundations of KRR
2. Explain different categories of representation and reasoning tasks
3. Assess the tradeoff between representation and reasoning
4. Identify which knowledge-based techniques are appropriate for which tasks
5. Apply KRR systems to challenging real-world problems.

Course Code	**:**	**AIPC 203**
Course Title	**:**	**Design and Analysis of Algorithms**
Number of Credits and L/T/P scheme	**:**	**4 & 3/0/2**
Prerequisites (Course code)	**:**	
Course Category	**:**	**PC**

Course Learning Objectives:

1. Able to design, implement and analysis of standard searching and sorting algorithms.
2. Implement standard divide and conquer, Dynamic programming, Greedy and backtracking algorithms.
3. Able to implement between different data structures i.e., trees, heaps etc. also, able to pick an appropriate data structure for any given design situation.
4. Able to implement the major graph algorithms and their analysis.

Course Content:

Unit-1 Introduction

Concept of Time and space complexity, analysis of algorithms, asymptotic notation, recurrence relations, design and analysis of D & C problems like quick sort etc, heap sort, priority queues, sorting in linear time, hashing, binary search trees.

Unit-2 Graph Algorithms

Graph representation & traversal (search), topological sort, strongly connected components, minimum spanning trees – Kruskal and Prim's, Single source shortest paths, relaxation, Dijkstra's algorithm, Bellman-Ford algorithm, single source shortest paths for directed acyclic graphs, all-pairs shortest path.

Unit-3 B-Trees and Dynamic programming

B-Trees: representation and operations; Elements of Dynamic Programming, structure and steps, Matrix-chain multiplication, longest common subsequence.

Unit-4 Greedy & Backtracking Approaches:

Greedy algorithms – Elements, activity-selection problem, Huffman codes, task scheduling problem, Knapsack Problem, Backtracking – Elements, 8 – Queens, Graph Coloring, Hamiltonian Cycles.

Text Books:

1. Cormen, Leiserson and Rivest: Introduction to Algorithms, 3/e, PHI.
2. Horowitz, Sahni, and Rajasekaran: Fundamentals of Computer Algorithms, Second Edition, Universities Press, Hyderabad.
3. Aho, Hopcroft, and Ullman: The Design and Analysis of Computer Algorithms, Addison Wesley.

Course Outcomes

1. Able to design, implement and analysis of standard searching and sorting algorithms.
2. Implement standard divide and conquer, Dynamic programming, Greedy and backtracking algorithms.
3. Able to implement between different data structures i.e., trees, heaps etc. also, able to pick an appropriate data structure for any given design situation.
4. Able to implement the major graph algorithms and their analysis.

Course Code	**:**	**AIPC 205**
Course Title	**:**	**Database Management Systems**
Number of Credits and L/T/P scheme	**:**	**4 and 3/0/2**
Prerequisites (Course code)	**:**	**Set Theory**
Course Category	**:**	**PC**

Course Learning Objectives:

1. To understand data models and database systems.
2. To understand relational database theory,
3. To learn database design and its normalization.
4. To understand concepts of transaction, its processing, and management.
5. To learn concurrency control techniques.
6. To develop practical skill in using MySQL DBMS.

Course Content:

Unit-1 (Introduction)

Data, Database, Database management system, Historical background from file systems to Database Systems, Data Models, Relational Data Model, ER model, Schemas and Instances, Database users, DBMS architecture.

Unit-2 (Relational Model and Query Language)

Theory of Relational Database, Key Integrity constraint, Relational Algebra, Relational Calculus: Domain Relational Calculus, Tuple Relational Calculus, SQL, queries writing in SQL.

Unit-3 (Database Design)

Normalization, Database Anomalies, Functional Dependencies, Candidate and Super Key, Non-loss Decomposition, Dependency Preservation, First, Second, Third Normal, BCNF, etc.

Unit-4 (Transaction Processing)

Transaction concept, a simple transaction model, states, ACID Properties, implementation of ACID properties, Serializability.

Unit-5 (Concurrency Control)

Need for Concurrency, Lock-based protocols, Deadlock, Starvation, deadlock handling, time-stamp based protocols, validation-based protocols.

Text Books:

1. A Silberschatz, H.F. Korth & S. Sudarshan: Data Base System Concepts, Mc Graw Hill, 4^{th}, 5^{th} or 6^{th} edition.
2. Elmasri & Navathe : Fundamentals of Database Systems, 5^{th}, 6^{th}, or 7^{th} edition Pearson.

Course Outcomes

1. To Design and Implement a small database for applications using MySQL DBMS.
2. To apply the concepts of normalization to database design.

Course Code	**:**	**MAIC*** **
Course Title	**:**	**Applied Linear Algebra**
Number of Credits and L/T/P scheme	**:**	**3 and 3/0/0**
Prerequisites (Course code)	**:**	**MAIC 102**
Course Category	**:**	**IC**

Course Objective:

l. To introduce the application of linear algebra and matrices in the different branches of mathematics.
2. To introduce theoretical aspects of linear algebra required for recent evolving branches like machine learning and data analysis.
3. Extensively introduce students with generalized inverse, QR decomposition and SVI).
4. Introduces Perron—Frobenius theorem, Quadratic form. Sylvester inertia theorem, Linear functional Bilinear mapping.

Course Content:

Unit 1:

Vector spaces, Bases and dimensions, Sums and direct sums, Quotient spaces.

Unit 2:

Linear transformations, Kernel and Image of a linear transformation, Rank-nullity theorem, Representation of linear transformations by matrices, Change of bases for linear transformations, Bases-change Matrices, Orthonormal bases, Gram-Schmidt process.

Unit 3:

Invariant subspaces, Cayley-Hamilton theorem, Minimal polynomial, Adjoint operators (matrix), Normal, unitary, and self-adjoint operators (matrix), Schur's Lemma, Spectral theorem for normal operators (matrix) (Unitary diagonalization, and triangulation of a matrix), Direct-sum decomposition, Cyclic subspaces, and Annihilators, Rational and Jordan canonical forms, LU and Cholesky decomposition, Householder's Reflection, QR and Polar Decomposition.

Unit 4:

Tridiagonal Matrix, Strum's Sequence, Projection Matrix, Singular value decomposition, Generalized inverse of the matrix, Perron—Frobenius theorem, Quadratic form. Sylvester inertia theorem, Linear functional, Bilinear mapping and inner product spaces.

Text Books:

1. K. Hoffman and R. A. Kunze, Linear Algebra, Prentice Hall of India.
2. H. Dym, Linear algebra in Action (Graduate studies in Mathematics, American Mathematical Society

Reference Books:

1. R A Horn, C R Johnson, Matrix Analysis, Cambridge.
2. J H Kwak, S Hong, Linear Algebra, Birkhauser

Course Outcomes:

At end of the course, the student will be able to:

1. Understand the basic concepts of linear algebra related to stability, accuracy, etc.
2. Find QR factorization of a matrix using Householder transformation and study its applications.
3. Write various algorithms to solve system of linear equations to understand computational issues.
4. Apply the SVD of a matrix in solving real life problems.

Course Code	**:**	**AIPC 200**
Course Title	**:**	**Decision Making Using AI**
Number of Credits and L/T/P scheme	**:**	**4 and 3/0/2**
Prerequisites (Course code)	**:**	**Basic understanding of computer science concepts and algorithms. Familiarity with introductory statistics and probability is beneficial but not mandatory.**
Course Category	**:**	**PC**

Course Learning Objectives

1. Understand the fundamentals of AI and its potential for decision making.
2. To introduce the concept of artificial intelligence, methods, techniques and applications
3. Gain practical experience through case studies and hands-on projects.

Course Content:

Unit-1 Introduction to AI and Decision Making: Fundamentals of AI: Definitions, history, key concepts. Decision-making frameworks: Rationality, bounded rationality, intuition. The role of AI in decision making: Opportunities and challenges.

Unit-2 Problem solving: State space search; production systems, search space control; depth first search, breadth-first search. Heuristic Based Search: Heuristic search, Hill climbing, best-first search, A*Algorithm and AO* algorithm, Min-max algorithms, game playing – Alpha beta pruning branch and bound, Problem Reduction, Constraint Satisfaction End, Means-End Analysis.

Unit-3 Representing and Reasoning with Uncertain Knowledge: probability, connection to logic, independence, Bayes rule, Bayesian networks, probabilistic inference, sample applications.

Unit-4 Decision-Making: basics of utility theory, decision theory, sequential decision problems, elementary game theory, sample applications.

Unit-5 Machine Learning: Introduction, Machine Learning Process, Machine learning: Supervised, unsupervised, and reinforcement learning, Feature Engineering-Feature Extraction, Feature Selection, Feature Engineering Methods, Feature Engineering, Data Visualization, Line Chart, Bar Chart, Pie Chart, Histograms, Scatter Plot, Seaborn-Distplot, joint plot. Learning nearest neighbour, naive Bayes, and decision tree classifiers, Q-learning for learning action policies, applications.

Unit-6 Applications of AI: 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.

Text Books:

1. Rich E., Knight K. and Nair B. S., Artificial Intelligence, Tata McGraw Hills (2009) 3rded.
2. Luger F. G., Artificial Intelligence: Structures and Strategies for Complex Problem Solving, Pearson Education Asia (2009) 6thed.

Reference Books:

1. Patterson W. D., Introduction to Artificial Intelligence and Expert Systems, Pearson (2015) 1st ed.
2. Russel S., Norvig P., Artificial Intelligence: A Modern Approach, Prentice Hall (2014) 3rd ed.

Course outcomes

1. Demonstrate fundamental understanding of Artificial Intelligence (AI) and its foundation.
2. Analyse basic and advanced search techniques.
3. Illustrate the concept of decision-making and machine learning.
4. Apply basic principles of ML Algorithms and Models in various application domain.

Course Code	**:**	**AIPC 202**
Course Title	**:**	**Operating Systems**
Number of Credits and L/T/P scheme	**:**	**4 and 3/ 0/ 2**
Prerequisites (Course code)	**:**	
Course Category	**:**	**IC/PC/PE/OE/AU**

Course Learning Objectives

1. To understand the services and design of an operating system.
2. To understand the structure and organization of file system
3. To understand the process states and various concepts such as scheduling and synchronization related with it.
4. To understand different memory management approaches.
5. Students should be able to use system calls for managing processes, memory and file system.
6. students should understand the data structures and algorithms for implementation of OS.

Course Content

Unit-1: Computer system architecture and organization, Introduction and evolution of OS, Introduction to distributed OS, Real time systems and multimedia systems. OS structures: OS services, system calls and programs, OS design and implementation. Processes: Process concept, scheduling policies, algorithms, multilevel queuing, operations on process, Inter-process communication. Threads: multithreading models and threading issues. CPU scheduling: Criteria and algorithms, multiprocessor and thread scheduling.

Unit-2: Process synchronization: critical sections, classical two process and n-process solutions, hardware primitives for synchronization, semaphores, monitors, classical problems in synchronization (producer-consumer, readers-writer, dining philosophers, etc.).

Deadlocks: modelling, resource allocation, characterization, prevention and avoidance, detection and recovery.

Unit-3: Memory management: Swapping, contiguous memory allocation, paging, multilevel paging, segmentation, demand paging, page replacement algorithms, allocation of frames, thrashing, working set model. Input/Output: I/O system and services, device controllers and device drivers, disks, scheduling algorithms and management.

Unit-4: File system interface: access methods, access control, directory structures, file organization, file sharing and protection. system performance, protection and security, , OS design considerations for security, access control lists and OS support, internet and general network security. Operating system as service provider: Access control matrix, access control list, capability matrix, encryption and access permissions in Linux and Windows, Operating system based security threats: Viruses and worms, Trojan horses, trapdoors and keyloggers, Buffer overflow attacks, phishing and social engineering, Methods to prevent buffer overflows, logic bomb, Denial of service, virtual machines: history, benefits, building blocks, types of virtual machines and their implementation, virtualization and operating system components.

Reference Books:

1. A. Silberschatz, Peter B. Galvin and G. Gagne, "Operating System Concepts," (6th or newer edition), Wiley.
2. H. Brinch, "Operating System Principles," Prentice Hall of India.
3. Dhamdhere, "Systems programming & Operating systems," TataMcGraw Hil
4. A. N. Habermann, "Introduction to Operating System Design," Galgotia publication, New Delhi.
5. A.S. Tanenbaum, "Modern Operating Systems," Prentice Hall of India.

Course Outcomes:

At the end of the course student will be able to

1. Understand functions, structures and history of operating systems
2. Able to know the design issues associated with operating systems
3. Master various process management concepts such as scheduling, synchronization, multithreading and deadlocks
4. Understand the various concepts associated with memory management such as virtual memory, demand paging, page replacements algorithms
5. Be familiar with various protection and security mechanisms
6. Be familiar with virtualization and operating system components

Course Code	**:**	**AIPC 204**
Course Title	**:**	**Neural Networks**
Number of Credits and L/T/P scheme	**:**	**4 and 3/0/2**
Prerequisites (Course code)	**:**	**Probability and Random Processes, Linear Algebra**
Course Category	**:**	**IC/PC/PE/OE/AU**

Course Learning Objectives

The objective of this course is Students can able to:

1. Understand the difference between a biological neuron and artificial neuron.
2. Understand building blocks of Neural Networks.
3. Understand recent advances made in applying neural network or deep learning models.
4. Understand the application areas of neural networks

Course Content

Unit-1 Basics: Biological Neuron, Idea of computational units, McCulloch–Pitts unit and Thresholding logic, Linear Perceptron, Perceptron Learning Algorithm, Linear separability. Convergence theorem for Perceptron Learning Algorithm, Historical Development of Neural Networks, Biological Neural Networks, Comparison Between them and the Computer, Comparison Between Artificial and Biological Neural Network.

Unit-2 Artificial Neural Networks: Basic concepts of artificial neurons, single and multi layer perceptrons, perceptron learning algorithm, its convergence proof, different activation functions, softmax cross entropy loss function, Gradient Descent, Backpropagation, Empirical Risk Minimization, autoencoders.

Unit-3 Deep Neural Networks: Difficulty of training deep neural networks, Greedy layerwise training, Newer optimization methods for neural networks (Adagrad, adadelta, rmsprop, adam, NAG), second order methods for training, Saddle point problem in neural networks, Regularization methods (dropout, drop connect, batch normalization), Types of errors, bias-variance trade-off, overfitting-underfitting.

Unit-4 Convolution neural networks (CNNs): Introduction to CNNs – convolution, pooling, Deep CNNs, Different deep CNN architectures – LeNet, AlexNet, VGG, PlacesNet, Training a CNNs: weights initialization, batch normalization, hyperparameter optimization, regularization, dropout, etc, Understanding and visualizing CNNs.

Unit-5 Recurrent neural networks (RNNs), Generative models and Application of neural networks: Recurrent Architectures, Transformers, Vision Transformers, Discussion on Recurrent Neural Networks (RNNs), Long-Short Term Memory (LSTM) architectures, Restrictive Boltzmann Machines (RBMs), Stacking RBMs, Belief nets, Learning sigmoid belief nets, Deep belief nets. Applications in vision, speech and natural language processing.

Text Books:

1. Ian Goodfellow, Yoshua Bengio and Aaron Courville, Deep learning, In preparation for MIT Press 2016.

Reference Books:

1. Raúl Rojas, Neural Networks: A Systematic Introduction, 1996.
2. S. Haykin, Neural Networks and Learning Machines, Prentice Hall of India, 2010.
3. Satish Kumar, Neural Networks - A Class Room Approach, Second Edition, Tata McGraw-Hill, 2013.
4. B. Yegnanarayana, Artificial Neural Networks, Prentice- Hall of India, 1999 4. C.M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.

Course outcomes

1. Able to demonstrate knowledge of understanding of neural networks.
2. Understand various deep learning models such CNN, Autoencoders, RNN, transformer, etc.
3. Analyze various applications solved through the use of deep learning models.
4. Design and implement their own deep learning models for the problem of their choice.

Course Code	**:**	**AIPC 206**
Course Title	**:**	**Fuzzy Systems and Evolutionary Computing**
Number of Credits and L/T/P scheme	**:**	**4 and 3/0/2**
Prerequisites (Course code)	**:**	**Basics of Calculus and classical set theory**
Course Category	**:**	**PC**

Course Learning Objectives

The objective of this course is Students can able to:

1. To understand the fundamental theory and concept of fuzzy logic and sets.
2. To introduce concepts of various fuzzy systems and their functions, and fuzzy rules for reasoning.
3. To understand the biologically inspired evolutionary algorithm such as neural networks, genetic algorithms, ant colony optimization, and bee colony optimization

Course Content

Unit-1 Introduction to Fuzzy Logic, Classical Sets and Fuzzy sets: Classical sets, Fuzzy sets. Classical Relations and Fuzzy Relations: Cartesian Product of relation, classical relation, fuzzy relations, tolerance and equivalence relations, non-iterative fuzzy sets. Membership Function: features of the membership functions, fuzzification, methods of membership value assignments. Defuzzification: Lambda-cuts for fuzzy sets, Lambda-cuts for fuzzy relations, Defuzzification methods. Fuzzy Arithmetic and Fuzzy measures: fuzzy arithmetic, fuzzy measures, measures of fuzziness, fuzzy integrals.

Unit-2 Fuzzy Rule base and Approximate reasoning: Basic Definition and Terminology, Set-theoretic Operations, Member Function Formulation and Parameterization, Fuzzy Rules and Fuzzy Reasoning, Extension Principle and Fuzzy Relations, Fuzzy If-Then Rules, Fuzzy Reasoning , Fuzzy Inference Systems, Mamdani Fuzzy Models, Sugeno Fuzzy Models, Tsukamoto Fuzzy Models, Input Space Partitioning and Fuzzy Modeling.

Unit-3 Neural networks: Single layer networks, Perceptrons: Adaline, Multilayer Perceptrons Supervised Learning, Back-propagation, LM Method, Radial Basis Function Networks, Unsupervised Learning Neural Networks, Competitive Learning Networks, Kohonen Self-Organizing Networks, Associative Memory, Learning Vector Quantization, Hebbian Learning. Recurrent neural networks. Adaptive neuro-fuzzy information; systems (ANFIS), Hybrid Learning Algorithm, Applications to control and pattern recognition.

Unit-4 Derivative-free Optimization Genetic algorithms: Basic Concepts, Basic Operators for Genetic Algorithms, encoding, Crossover and Mutation Properties, Genetic Algorithm Cycle, Fitness Function, Reproduction, Applications of Genetic Algorithm, Differences of GA and traditional optimization methods. Basic genetic programming concepts Applications.

Evolutionary Computing, Simulated Annealing, Random Search, Downhill Simplex Search, Swarm optimization, ant colony optimization, bee colony optimization.

Text Books:

1. Soft Computing – Advances and Applications - Jan 2015 by B.K. Tripathy and J. Anuradha – Cengage Learning
2. Timothy J. Ross, “Fuzzy Logic with Engineering Applications”, Third Edition

Reference Books:

1. Principles of Soft Computing by S.N. Sivanandam, S. N. Deepa, John Wiley & Sons.
2. Neural Networks and Learning Machines by S. Haykin, Prentice Hall of India.
3. Genetic Algorithms in Search, Optimization and Machine Learning by D.E. Goldberg, Pearson.
4. Fuzzy Logic with Engineering Applications by T.J. Ross, John Wiley & Sons.
5. Foundations of Neural Networks, Fuzzy systems and Knowledge Engineering by N.K. Kasabov, MIT Press.
6. An Introduction to Genetic Algorithms by M. Mitchell, MIT Press.

Course Outcomes

1. Understand the concepts of classical and fuzzy sets.
2. Understand the basic ideas of fuzzy logic, operations and properties of fuzzy sets and also about fuzzy relations.
3. Understand the neural networks and its applications.
4. Solve single objective optimization problems using Genetic algorithm.
5. Understand the biologically inspired evolutionary algorithm such as neural networks, genetic algorithms, ant colony optimization, and bee colony optimization.