Centre for Artificial Intelligence

DATABASE MANAGEMENT SYSTEM

(31242201/ 24242203)

COURSE OBJECTIVES

- To understand the different issues involved in the design and implementation of a database system.
- To study the physical and logical database designs, database modelling, relational, hierarchical and network models.
- To understand and use data manipulation language to query, update and manage a database.

UNIT-I: DBMS: Database Approach v/s Traditional File Approach, Advantages of Database System, Database Users and Administrator, Database System Environment, Application Architectures, Schemas, Instances, Data Independence, Data Models: Hierarchical Data Model, Network Data Model & Relational Data Model, Comparison between Models. Entities and Relationship Model: Entity types, Entity sets, Attributes and Keys, Relationship Types and Sets, Constraints, Design issue, E-R Diagram, Weak Entity Sets.

UNIT-II: Relational Model: Structure of Relational Databases: Relation, Attribute, Domain, Tuples, Degree, Cardinality, Views, Database Relations, Properties of Relations, Attributes, Keys, Attributes of Relation, Domain Constraints, Integrity Constraints. Relational Algebra: Concepts and Operations: Select, Project, Division, Intersection, Union, Division, Rename, Join etc.

UNIT-III: SQL: Purpose of SQL, Data Definition Language (DDL) Statements, Data Manipulation Language (DML) Statements Update Statements & Views in SQL, Data Control Language (DCL), Triggers. Relational Database Design: Purpose of Normalization, Data Redundancy and Update Anomalies, Functional Dependency, Process of Normalization, Various Normal Forms: 1NF, 2NF, 3NF, BCNF, Decomposition, Desirable Properties of Decomposition: Dependency Preservation, Lossless Join, Problems with Null Valued & Dangling Tuple, Multivalued Dependencies.

UNIT-IV: Transaction Management: Transaction Concept, Transaction State, Concurrent Executions, Serializability: Conflict and View Serializability, Concurrency Control: Lock-Based Protocol, Recovery: Log-Based Recovery.

UNIT-V (DYNAMIC CONTENTS): Big Data and Distributed Databases: CAP theorem, eventual consistency, sharding. NoSQL and NewSQL Databases: Document (MongoDB), Key-Value (Redis), Column-family (Cassandra), Graph (Neo4j), Google Spanner, CockroachDB. Real-Time Data Processing and Streaming Databases: Apache Kafka, Apache Flink.

RECOMMENDED BOOKS

1. Database System Concepts, Abraham Silberschatz Henry F. Korth S. Sudarshan, McGraw-Hill 6th Edition.
2. Database Management System, Raghu Ramakrishnan Johannes Gehrke, McGraw Hill 3rd Edition.
3. Fundamentals of Database System, Elmasri & Navathe, Addison-Wesley Publishing, 5th Edition.
4. An Introduction to Database Systems, Date C. J, Addison-Wesley Publishing, 8th Edition.

Centre for Artificial Intelligence

COURSE OUTCOMES

After completion of this course, the students would be able to:

CO1	Differentiate between traditional file systems and the database approach.
CO2	Construct relational schemas and demonstrate the use of relational algebra operations to query and manipulate relational data.
CO3	Develop SQL queries for data definition, manipulation, and control, and analyze relational schemas for normalization.
CO4	Examine transaction processing concepts and analyze concurrency control and recovery mechanisms in database systems.
CO5	Compare traditional RDBMS with modern distributed and NoSQL databases, and evaluate their suitability for real-time data processing scenarios.

CO-PO Mapping Matrix

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	2	1	1							2	2	2
CO2	3	3	2	2	2							2	2	2
CO3	3	3	2	2	2				2	2	2	2	2	3
CO4	3	3	3	2	2				2	2	2	2	2	3
CO5	2	2	3	2	3	2	2	1	2	2	2	3	3	3

Centre for Artificial Intelligence
THEORY OF COMPUTATION
(31242202/ 27242202/ 28242202)

COURSE OBJECTIVES

- To understand computability, decidability, and complexity through problem solving.
- To analyze and design abstract model of computation & formal languages
- To understand and conduct mathematical proofs for computation and algorithms.

UNIT-I: Introduction of Automata Theory: Examples of automata machines, Finite Automata as a language acceptor and translator, Moore machines and mealy machines, composite machine, Conversion from Mealy to Moore and vice versa.

UNIT-II: Types of Finite Automata: Non Deterministic Finite Automata (NDFA), Deterministic finite automata machines, conversion of NDFA to DFA, minimization of automata machines, regular expression, Arden's theorem. Meaning of union, intersection, concatenation and closure, 2 way DFA.

UNIT-III: Grammars: Types of grammar, context sensitive grammar, and context free grammar, regular grammar. Derivation trees, ambiguity in grammar, simplification of context free grammar, conversion of grammar to automata machine and vice versa, Chomsky hierarchy of grammar, killing null and unit productions. Chomsky normal form and Greibach normal form.

UNIT-IV: Push down Automata: example of PDA, deterministic and non-deterministic PDA, conversion of PDA into context free grammar and vice versa, CFG equivalent to PDA, Petri Net model. Turing Machine: Techniques for construction. N-P complete problems.

UNIT-V (DYNAMIC CONTENTS): Decidability and Recursively Enumerable Languages, decidability, decidable languages, undecidable languages. Verifying neural network behaviour (undecidability challenges). Halting problem analogues in AI agent planning. Context-free grammars for large code-model training datasets.

RECOMMENDED BOOKS

1. Introduction to Automata Theory Language & Computation, Hopcroft & Ullman, Narosa Publication.
2. Element of the Theory Computation, Lewis & Christors, Pearson.
3. Theory of Computation, Chandrasekhar & Mishra, PHI.
4. Theory of Computation, Wood, Harper & Row.
5. Introduction to Computing Theory, Daniel I-A Cohen, Wiley.

COURSE OUTCOMES

After completion of this course, the students would be able to:

CO1	explain the basic concepts of switching and finite automata theory & languages.
CO2	relate practical problems to languages, automata, computability and complexity.
CO3	construct abstract models of computing and check their power to recognize the languages.
CO4	analyze the grammar, its types, simplification and normal form.
CO5	apply formal mathematical methods to prove properties of languages, grammars and automata.

Centre for Artificial Intelligence

CO-PO Mapping Matrix													
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
CO1	3	2	3	3	2							3	1	1
CO2	2	3	3	2	3			2				3	2	2
CO3	3	3	3	3	3				2	2	2	2	2	2
CO4	3	2	3	3	2	3	3		3	3	2	3	3	2
CO5	3	2	2	3	3	3	2	2		2	3	3	3	3

Centre for Artificial Intelligence
SOFTWARE ENGINEERING
(31242203/ 24242204/ 27242203/ 28242203)

COURSE OBJECTIVES

- To understand the process of software development and software life cycle models.
- To understand project management and risk management associated with various types of projects.
- To know the familiarity with the concept of software testing, quality assurance and configuration management process.

UNIT-I: Introduction to Software Engineering, Software Development Life Cycle (SDLC): Waterfall, Incremental, Spiral Model, Prototyping, RAD, V-Model. Agile Software Development: Scrum, Kanban, XP, Roles, Ceremonies, Artifacts. Software Engineering Ethics and Professional Practice

UNIT-II: Requirements Engineering: Functional vs Non-functional requirements. Elicitation techniques (interviews, observation, surveys, JAD, prototyping). Requirements specification & documentation (SRS). Requirements modeling (DFDs, Use Case diagrams). Requirements validation & management. System Design: Architectural design styles - Layered, Client–Server, Microservices. Design principles. Coupling, Cohesion, Modularity. UML Diagrams: Class, Sequence, Activity, State, Component

UNIT-III: Software Construction: Coding standards, guidelines, and best practices. Code refactoring & code reviews. Software reuse & component-based development. Software Testing: Testing levels - Unit, Integration, System, Acceptance. Testing techniques - White-box testing (basis path, control-flow testing); Black-box testing (equivalence partitioning, boundary value analysis). Test case design, test automation fundamentals. Debugging strategies. Maintenance: Types of maintenance - corrective, preventive, perfective, adaptive. Reverse engineering, reengineering. Software evolution & legacy system migration.

UNIT-IV: Project Management: Project planning and scheduling (Gantt charts, PERT/CPM). Estimation techniques: LOC, Function Point, Use Case Points, COCOMO. Risk management and mitigation. Quality Assurance & Standards: Software quality metrics. Reviews, audits, inspections. ISO 9001, CMMI, Six Sigma in software. Software reliability models.

UNIT-V (DYNAMIC CONTENTS): Modern Development Paradigms: Microservices Architecture & Serverless Computing, Cloud-Native Software Engineering, Configuration management systems (Git, CI/CD basics). DevOps fundamentals (Continuous Integration, Delivery, Deployment), Containerization & Orchestration (Docker, Kubernetes).

RECOMMENDED BOOKS

1. Software Engineering, Sommerville, Pearson.
2. Software Engineering: A Practitioner's Approach, Roger S. Pressman, McGraw Hill.
3. Software Engineering, K.K. Agrawal & Yogesh Singh, New Age Publication.
4. Software Engineering, Rajib Mall, PHI.

Centre for Artificial Intelligence

COURSE OUTCOMES

After completion of this course, the students would be able to:

CO1	Select appropriate development models for specific software project scenarios.
CO2	Evaluate requirement specifications, propose improvements and design models.
CO3	Analyze program behavior to identify defects and implement debugging strategies.
CO4	Design project plans using estimation and scheduling techniques.
CO5	Analyse modern software development paradigms.

CO-PO Mapping Matrix														
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
CO1	3	2	2	1	1							2	2	2
CO2	3	3	2	2	2						1	2	2	2
CO3	3	3	2	2	2					1	2	2	2	3
CO4	3	3	3	2	2	1	1	1	2	2	2	2	2	3
CO5	2	2	3	2	3	2	2	1	2	2	2	3	3	3

Centre for Artificial Intelligence

NETWORK AND WEB SECURITY

(31242204/ 24242202/ 27242204/ 28242204)

COURSE OBJECTIVES

- To introduce foundational principles of network and web security.
- To equip students with the ability to analyze, design, and implement security solutions.
- To develop competency in emerging and modern security technologies

UNIT-I: Concepts of security: CIA triad, threats, vulnerabilities, attacks. Types of attacks: passive vs active, DoS/DDoS, spoofing, MITM, replay. Security services & mechanisms. Steganography. Cryptographic foundations: Symmetric & asymmetric encryption, DES and AES.

UNIT-II: Public Key Infrastructure (PKI) & Key Management, RSA and Diffie-Hellman key exchange. Hashing, digital signatures, certificates, Kerberos. Secure communication basics (TLS/SSL).

UNIT-III: Firewalls: packet filters, stateful inspection, proxy firewalls. Intrusion Detection & Prevention Systems (IDS/IPS). VPN: IPSec, SSL VPN. Secure routing & switching. Wireless network security: WEP, WPA, WPA2/WPA3. Email security: S/MIME, PGP. Network access control, RADIUS.

UNIT-IV: Web application architecture & threat landscape. SQL injection, XSS, CSRF, Broken authentication, Insecure deserialization. Secure coding principles for web apps. Web communication security (HTTPS, HSTS, CSP). Web server security and misconfigurations. Cookie security, session management attacks.

UNIT-V (DYNAMIC CONTENTS): Zero Trust Architecture, Smart contract security, Blockchain-based identity management, Penetration testing, Quantum-safe cryptography.

RECOMMENDED BOOKS

1. Network Security Essentials: Applications & Standards, William Stallings, Pearson.
2. Network Security - Private Communication in a Public World, G. Kaufman, Prentice-Hall.
3. Cryptography & Network Security – Principles & Practice, William Stallings, Prentice Hall.

COURSE OUTCOMES

After completion of this course, the students would be able to:

CO1	Analyze security concepts and basic cryptographic techniques like DES and AES.
CO2	Evaluate PKI and key management methods and secure authentication and communication (RSA, DH, TLS).
CO3	Compare and configure network security tools such as firewalls, IDS/IPS, VPNs, and wireless security protocols.
CO4	Apply secure coding and web security controls to mitigate major web vulnerabilities.
CO5	Explain emerging security concepts (Zero Trust, blockchain, smart contracts) and assess quantum-safe techniques.

Centre for Artificial Intelligence

CO-PO Mapping Matrix														
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
CO1	3	2	3	3	2					2	2	3	1	1
CO2	2	3	3	2	3			2		3	3	3	2	2
CO3	3	3	3	3	3				2	2	2	2	2	2
CO4	3	2	3	3	2	3	3		3	3	2	3	3	2
CO5	3	2	2	3	3	3	2	2		2	3	3	3	3

Centre for Artificial Intelligence

CALCULUS AND OPTIMIZATION TECHNIQUES

(31242205/ 24242205/ 27242205/ 28242205)

COURSE OBJECTIVES

- To introduce the techniques of differential and integral calculus in engineering problems
- To illustrate the concept of ordinary differentiation equation
- To explore linear programming problem and numerical optimization

UNIT-I: Maclaurin's and Taylor's theorem, Partial differentiation, Euler's theorem, Jacobian, Maxima and Minima of one and two variables.

UNIT-II: Definite integral as limit of a sum, application in summation of series, Beta and Gamma function and its properties, transformation of Beta function, Gamma functions, transformation of Gamma function, relation between Beta and Gamma function, Legendre's duplication formula, double & triple integral, Change of order of integration, Length of the curves, Volumes and surfaces.

UNIT-III: Ordinary differential equations of first and higher order, Linear higher order differential equation with constant coefficients, Homogeneous linear differential equation and Simultaneous differential equations.

UNIT-IV: Concept of optimization, constrained and unconstrained optimization, LPP formulation, Graphical method, Simplex method, Duality of LPP, Transportation and Assignment problems.

UNIT-V: Concept of numerical methods, methods for solving matrix problems and linear systems by LU decomposition: Crout & do little method, Gauss elimination, Gauss-Seidel, and Gauss Jacobi, Interpolation: finite differences, difference operators, Newton's interpolation formula, Newton's divided difference formula, Lagrange's interpolation formula, singular value decomposition.

RECOMMENDED BOOKS

1. E. Kreyszig: Advance Engineering Mathematics, John Wiley & Sons, 10 th Edition (2011).
2. H. A. Taha: Operations Research an Introduction, Pearson, 9 th Edition (2014).
3. R. K. Jain, S. R. K. Iyengar: Advance Engineering Mathematics, Narosa Publishing House Pvt.Ltd, 5 th Edition (2016).
4. F. B .Hildebrand: Advanced Calculus for application, Englewood Cliffs, N. J. Prentice- Hall, 2nd Edition (1980).
5. J. Nocedal and S. Wright: Numerical Optimization, Springer Series in Operations Research and Financial Engineering, 2006.
6. B.V. Ramanna: Higher Engineering Mathematics, McGraw Hill Education, 1 st Edition (2017).
7. Introduction to Linear Optimization by Bertsimas, Tsitsiklis. MIT Press (1997)

COURSE OUTCOMES

After completion of this course, the students would be able to:

CO1	apply differential calculus in basic engineering problems
CO2	use integration techniques to determine the solution of various complex problems
CO3	solve the differential equations by various methods
CO4	find the optimal solution using various methods of linear programming problems.
CO5	evaluate the numerical techniques

Centre for Artificial Intelligence

CO-PO Mapping Matrix														
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
CO1	3	3	3	3	1	2		1		1	1	3	2	2
CO2	3	3	3	3	2	3	1	2	1	1	1	3	2	2
CO3	3	3	3	3	3	1	1		1	1	1	3	2	1
CO4	3	3	3	3	3				2	2	3	3	2	2
CO5	3	3	3	3	3				2	2		3	2	1

Centre for Artificial Intelligence

ROBOT KINEMATICS

(24242201)

COURSE OBJECTIVES

- To introduce the functional elements of Robotics
- To impart knowledge on the kinematics of mechanism
- To introduce the dynamics and control of manipulators

UNIT-I: Introduction to Robotics: History, Law of Robotics: Terminologies, Classifications Overview: Links & Joints, Coordinate Systems, Work Volume, Precision, Repeatability & Accuracy Position and Orientation of Objects: Roll, Pitch and Yaw Angles, Joint Configuration of Five Types of Serial Manipulators, Wrist Configuration, Overview of end effector.

UNIT-II: Degrees of Freedoms: of various mechanisms and its application, Kinematics: Mobility Analysis, Displacement Analysis: constrained mechanisms and robots, Velocity Analysis: constrained mechanisms and robots, singularity.

UNIT-III: Translation Matrix - Rotation matrix, Euler Angles, Quaternion Fundamental, Dot and Cross Products, Frames and Joint Coordinates, Homogeneous Transformation, D-H Convention and Procedures: Forward kinematics Solution using D-H Convention: 3R Planar mechanism, 3 DOF RRP, Cartesian, Articulated 3 DOF robots.

UNIT-IV: Joint space technique, Trajectory planning and control, use of p-degree polynomial, Cubic Polynomial-Cartesian space Technique, Parametric descriptions, Straight line and circular paths, Position and orientation planning.

UNIT-V (DYNAMIC CONTENTS): SCARA Manipulator Kinematic and Dynamic analysis, Jacobian-Prismatic and rotary joints, Lagrange Euler formulation, Dynamic model: Manipulator control problem, Linear control schemes, Adaptive controller. Selection and Application of Serial Manipulators: Industrial applications of robots, Medical, Household, Entertainment, Space, Underwater, Defence, Disaster management. Applications of Micro and Nano robots.

Centre for Artificial Intelligence

RECOMMENDED BOOKS

1. R.K. Mittal and I.J.Nagrath, Robotics and Control, Tata McGraw Hill, New Delhi,4th Reprint, 2005.
2. John J. Craig ,Introduction to Robotics Mechanics and Control, Third edition, Pearson Education, 2009.
3. M. P. Groover, M. Weiss, R.N. Nageland N. G.Odrej, Industrial Robotics, McGraw-Hill Singapore, 1996.
4. Ashitava Ghoshal, Robotics-Fundamental Concepts and Analysis', Oxford University Press, Sixth impression, 2010.
5. K. K.Appu Kuttan, Robotics, I K International, 2007.
6. Edwin Wise, Applied Robotics, Cengage Learning, 2003.
7. R.D.Klafter,T.A.Chimielewski and M.Negin, Robotic Engineering–An Integrated Approach, Prentice Hall of India, New Delhi, 1994.
8. B.K.Ghosh, Control in Robotics and Automation: Sensor Based Integration, Allied Publishers,Chennai, 1998.
9. S.Ghoshal, " Embedded Systems & Robotics" – Projects using the 8051 Microcontroller", Cengage Learning, 2009.
10. S. B. Nikku, Introduction to Robotics – Analysis, Control, Applications, 3rd edition, John Wiley & Sons Ltd., 2020.
11. K. Saha, Introduction to Robotics 2e, TATA McGraw Hills Education 2014.

COURSE OUTCOMES

After completion of this course, the students would be able to:

CO1	illustrate the significance, social impact and future prospects of robotics and automation in various engineering applications.
CO2	describe the components and anatomy of robotic systems and basics of robotics.
CO3	explain different motions of a robotic system through kinematic modeling.
CO4	employ a suitable path planning of end-effectors for a given robotics application.
CO5	develop the kinematic and dynamic model for a robot manipulator.

CO-PO Mapping Matrix

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	1		1		1	3	2	2	1		1	1	3	2
CO2	3	1	1		1									
CO3	2	3	2	3	3	1			1			1	3	1
CO4	2	3	3	2	3	2								
CO5	3	3	3	3	3	1	1	1	2	2	1	3	3	2

Centre for Artificial Intelligence

DATA SCIENCE

(27242201)

COURSE OBJECTIVES

- To provide foundational understanding of data science concepts, workflow, and applications.
- To equip students with essential skills in data collection, cleaning, preprocessing, statistical analysis, core machine learning algorithms and exploratory data analysis.

UNIT-I: Data Science overview, roles & responsibilities of a data scientist, CRISP-DM methodology, Types of data, data collection methods, data quality issues. Data acquisition: APIs, web scraping, databases. Data cleaning: handling missing values, outliers, duplicates.

UNIT-II: Data transformation: normalization, standardization, encoding categorical variables, feature scaling. Feature engineering: extraction, selection, dimensionality reduction (PCA basics). Visualization using Matplotlib, Seaborn, Plotly. Advanced visualizations: heatmaps, pair plots, box plots, violin plots.

UNIT-III: Descriptive statistics: mean, median, variance, skewness. Probability distribution functions: Normal, Binomial, Poisson, Exponential. Hypothesis testing: z-test, t-test, chi-square test. Correlation and covariance. Regression basics: simple and multiple linear regression; evaluation metrics.

UNIT-IV: Types of machine learning: supervised, unsupervised, reinforcement learning. Supervised learning: Linear regression, Logistic regression, KNN, Naive Bayes, Decision Trees, Random Forest, SVM. Unsupervised learning: K-Means clustering, Hierarchical clustering, DBSCAN, PCA. Model evaluation metrics (Accuracy, Precision, Recall, F1, ROC-AUC, Confusion matrix, RMSE, MAE). Overfitting, underfitting, bias-variance tradeoff, cross-validation, regularization.

UNIT-V (DYNAMIC CONTENTS): Big Data technologies overview (Hadoop, Spark basics). Deep Learning fundamentals and CNN/RNN concepts, frameworks (TensorFlow/PyTorch).

RECOMMENDED BOOKS

1. Python Data Science Handbook, Jake VanderPlas, O'Reilly Media, 2016.
2. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, Aurélien Géron, 3rd Edition, O'Reilly Media, 2022.
3. An Introduction to Statistical Learning with Applications in R (or Python), Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani, Springer, 2nd Edition, 2021.
4. Python for Data Analysis, Wes McKinney, 3rd Edition, O'Reilly, 2022.

COURSE OUTCOMES

After completion of this course, the students would be able to:

CO1	Explain the fundamental concepts, workflow, and tools of data science
CO2	Apply appropriate data preprocessing and visualization techniques.
CO3	Apply statistical reasoning and hypothesis testing for data-driven decisions.
CO4	Implement various supervised and unsupervised machine learning algorithms.
CO5	Explain emerging trends and advanced technologies in Data Science.

Centre for Artificial Intelligence

CO-PO Mapping Matrix													
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
CO1	3	2	1	1	3				1	1		2	2	2
CO2	3	3	3	3	3				2	2		3	3	3
CO3	3	3	1	3	3				2	1	1	3	3	3
CO4	3	3	3	3	3	1	1		3	2	2	3	3	3
CO5	2	2	2	2	3	2	2	1	1	2	1	3	3	3

Centre for Artificial Intelligence

MACHINE LEARNING

(28242201)

COURSE OBJECTIVES

- To introduce fundamental concepts, mathematical foundations, and scope of machine learning.
- To develop skills for model evaluation, optimization, and practical application of ML algorithms.

UNIT-I: Introduction to ML: definition, applications, types of learning. ML pipeline: data loading, preprocessing, modeling, evaluation, deployment. Mathematical foundations: linear algebra basics, gradient descent, cost functions. Parametric vs. non-parametric models. Bias–variance trade-off. Python ML ecosystem: NumPy, Pandas, Scikit-learn.

UNIT-II: Linear models: Linear & Polynomial Regression, Ridge, Lasso, Logistic Regression. Tree-based methods: Decision Trees, Random Forest, Gradient Boosting. Instance-based: K-Nearest Neighbors. Probabilistic models: Naive Bayes, Gaussian Discriminant Analysis. Support Vector Machines (hard & soft margin, kernel trick).

UNIT-III: Clustering: K-means, hierarchical clustering, DBSCAN. Association rule mining: Apriori, FP-growth. Anomaly detection. Dimensionality reduction: PCA, SVD. Visualization of high-dimensional data (t-SNE basics).

UNIT-IV: Performance metrics: precision, recall, F1-score, confusion matrix, ROC-AUC. Cross-validation techniques. Hyperparameter tuning: Grid Search, Random Search, Bayesian Optimization. Ensemble methods: bagging, boosting (AdaBoost, Gradient Boosting, XGBoost basics). Overfitting/Underfitting prevention, regularization. Handling imbalanced datasets (SMOTE, resampling).

UNIT-V (DYNAMIC CONTENTS): Deep Learning overview: Neural networks, backpropagation, activation functions. Modern architectures: Transformers, Vision Transformers, CNN advancements.

RECOMMENDED BOOKS

1. Machine Learning: A Probabilistic Perspective, Kevin Murphy
2. Pattern Recognition and Machine Learning, Christopher Bishop
3. Introduction to Machine Learning with Python, Andreas C. Müller & Sarah Guido.

COURSE OUTCOMES

After completion of this course, the students would be able to:

CO1	Explain the fundamental concepts, workflow, and mathematical foundations of machine learning.
CO2	Implement major supervised learning algorithms for regression and classification tasks.
CO3	Apply unsupervised learning techniques to discover patterns in data.
CO4	Evaluate, optimize, and improve machine learning models using advanced techniques.
CO5	Explain modern machine learning advancements.

Centre for Artificial Intelligence

CO-PO Mapping Matrix													
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
CO1	3	2	1	1	3				1	1		2	2	2
CO2	3	3	3	3	3				2	2		3	3	3
CO3	3	3	1	3	3				2	1	1	3	3	3
CO4	3	3	3	3	3	1	1		3	2	2	3	3	3
CO5	2	2	2	2	3	2	2	1	1	2	1	3	3	3

Centre for Artificial Intelligence

PROJECT MANAGEMENT, ECONOMICS & FINANCING

(31242211/ 24242211/ 27242211/ 28242211)

COURSE OBJECTIVES

- To provide knowledge about project attributes and planning essentials, develop project networks, make rational decisions for project completion, utilize resources effectively, and understand the basics of project finances and management.

UNIT-I: Project Planning: Introduction to Project Management, Difference between Project and Production, Attributes of a Project: Time, Cost, Quality and Safety. Stakeholders of a Project, Project life cycle. Project Planning: Types of Project Plans and feasibility.

UNIT-II: Project Network logic: Project Networking and work flows, Activity duration and methods of estimating activity duration – One time estimate three time estimates, Duration estimation procedure. Use of Bar Charts, Mile stone charts and networks, Network representation schemes: Activity on Arrow and Activity on Node Networks (A-o-A & A-o-N), Logic behind developing project network and simple network calculations, Critical paths and floats.

UNIT-III: Decision making through networks: CPM, PERT & PDM: Use of network in Decision Making: Importance of critical path, Monitoring the progress and updating the project plan. Use of floats in Resource smoothening, Introduction to Precedence Diagramming Method (PDM), Different lag and lead relations in terms of SS(Start to Start), SF(Start to Finish), Finish to Start(FS), and Finish to Finish(FF) and composite relations.

UNIT-IV: Project Cost Control: Breakeven analysis in planning stage, Direct and indirect cost, slope of direct cost curve, Total project cost and optimum duration, contracting the network for cost optimization. Escalation & Variation in prices.

UNIT-V: Projects Financing: Introduction to project financing; Role of governments in financing projects, Funder and Concessionaire: Economic multiplier effects of Projects; Means of financing-public finance and private finance, Granting authority: World Bank Group, IMF,ADB, Micro and Small Enterprises Funding Scheme (MSME), Elementary understanding of Procurement of infrastructure projects through Public Private Partnership (PPP) route, Build Operate Transfer (BOT), Build Operate Own & Transfer (BOOT); Stakeholders' perspectives, Lifecycle of PPP projects, Micro & Macro economics concepts and its application in Project Financing.

RECOMMENDED BOOKS

1. Project Management Scheduling PERT and CPM by Dr. B.C. Punmia, K.K. Khandelawal
2. PERT & CPM Principles and Applications by L.S. Srinath, Affiliated EWP Pvt. Ltd.
3. Project Planning and Control by Albert Lester, Fourth Edition Elsevier Butterworth-Heinemann.
4. A Management Guide to PERT/CPM With GERT/PDM/DCPM and Other networks by Jerome D. Wiest, Ferdinand K. Levy, Prentice Hall.
5. Project Management with CPM and PERT by Joseph J . Moder, Cecil R . Phillips, Van Nostrand Reinhold Company

Centre for Artificial Intelligence

COURSE OUTCOMES

After completion of this course, the students would be able to:

CO1	Explain the attributes of the project and its different phases.
CO2	Develop the project network based on work breakdown structure and estimation of activity durations.
CO3	Analyze the project network and make decisions on the various alternates.
CO4	Evaluate the optimum cost of the project for assigned deadlines.
CO5	Apply different methods for managing the finances to complete a project within stipulated time.

CO-PO Mapping Matrix

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	2	1	1			1				1	3	1		
CO2	3	3	3	2	2				1		3	1	1	1
CO3	2	3	2	3	2						3	1		
CO4	2	3	2	3	2						3	1	1	1
CO5	2	2	2	2	1	1		1	1		3	1	1	1

ANNEXURE - VI

Experiment list/ Lab manual and project list (Macro Project-II) of all the Laboratory Courses
B.Tech.
[Artificial Intelligence (AI)/ Information Technology (Artificial Intelligence and Robotics)/ Artificial Intelligence (AI) and Data Science/ Artificial Intelligence (AI) and Machine Learning]
IV Semester
for batch admitted in 2024-25

Centre for Artificial Intelligence

DATABASE MANAGEMENT SYSTEM LAB

(31242206/ 24242207)

1. SQL Queries: SELECT queries using WHERE, BETWEEN, LIKE, ORDER BY, GROUP BY, HAVING.
2. DDL Commands: CREATE, ALTER, DROP tables. Defining constraints (PK, FK, UNIQUE, CHECK, NOT NULL).
3. DML Commands: INSERT, UPDATE, DELETE operations using sample datasets.
4. SQL Joins: INNER JOIN, LEFT JOIN, RIGHT JOIN, FULL JOIN queries.
5. Set Operations: UNION, INTERSECT, MINUS/EXCEPT operations.
6. Aggregate Functions: COUNT, SUM, AVG, MIN, MAX with grouping.
7. Subqueries & Nested Queries: Single-row, multi-row, and correlated subqueries.
8. Views: Creating, updating, and deleting views.
9. Indexes & Sequences: Creating and using indexes for optimization. Creating sequences (if supported).
10. PL/SQL Basics: Anonymous blocks demonstrating variables, loops, and conditional statements.
11. Stored Functions: Writing user-defined functions (UDFs).
12. Stored Procedures: Creating and invoking stored procedures.
13. Cursors: Explicit cursor to iterate over table rows.
14. Exceptions in PL/SQL: Predefined and user-defined exceptions.
15. Database Triggers: Row-level & statement-level triggers (BEFORE/AFTER INSERT/UPDATE/DELETE).
16. Transactions & Concurrency: Demonstration of COMMIT, ROLLBACK, SAVEPOINT. Concurrent update simulation (using two sessions).

COURSE OUTCOMES

After completion of this course, the students would be able to:

CO1	Apply basic SQL commands (DDL, DML, DCL, TCL) to create and manage relational database objects.
CO2	Execute complex SQL queries involving joins, subqueries, set operations, indexing, and views.
CO3	Develop PL/SQL blocks including functions, procedures, cursors, exceptions, and triggers to implement business logic.
CO4	Demonstrate transaction management, concurrency control, and integrity constraints in relational databases.

CO-PO Mapping Matrix

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	2	1	1	1					1	1	1	1
CO2	3	2	2	2	1	1					1	1	2	2
CO3	3	2	3	2	2	2		1	1	2	1	2	2	2
CO4	2	2	2	3	2	2	1	1	1	2	1	2	2	2

Centre for Artificial Intelligence
JAVA PROGRAMMING LAB
(31242207/ 27242207/ 28242207)

1. Introduction to Java Programming Environment: Overview of Java platform: JVM, JDK, JRE. Installing and configuring Java. Writing and executing basic Java programs. Command-line arguments, data types, operators, expressions.
2. Control Structures and Arrays: Conditional statements: if, if–else, switch. Looping constructs: for, while, do-while. Arrays: single and multi-dimensional. String operations: String, StringBuffer, StringBuilder.
3. Object-Oriented Programming Concepts: Classes, objects, constructors. Method overloading & overriding. Inheritance (types, uses, super keyword). Abstraction: abstract classes and interfaces. Encapsulation and access modifiers. Packages and Java API usage.
4. Exception Handling, I/O, and File Management: Exception types, try-catch-finally, throw & throws. User-defined exceptions. Java I/O streams: byte streams, character streams. File reading and writing.
5. Multithreading and Collections Framework: Thread creation: extending Thread, implementing Runnable. Thread lifecycle and synchronization. Introduction to Collections. Using ArrayList, LinkedList, HashMap, HashSet, Iterator.
6. GUI Programming (AWT/Swing): AWT/Swing components: Frame, Button, Label, TextField. Layout managers. Event handling (ActionListener, MouseListener, etc.). Simple GUI-based applications.

COURSE OUTCOMES

After completion of this course, the students would be able to:

CO1	Apply fundamental Java programming constructs
CO2	Implement object-oriented programming concepts
CO3	Develop robust applications using exception handling, Collections Framework and file handling, and multithreading.
CO4	Design interactive GUI applications and demonstrate an integrated Java solution.

CO-PO Mapping Matrix

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	2	1	1							1	1	1
CO2	3	2	3	2	1							1	2	2
CO3	3	2	3	2	2		1		1	1	1	2	2	2
CO4	2	2	2	2	2	1	1	1	2	1	1	2	2	2

Centre for Artificial Intelligence
COMPETITIVE PROGRAMMING LAB
(31242208/ 24242208/ 27242208/ 28242208)

OBJECTIVE:

- To build strong foundational skills in programming logic, time–space complexity, and problem-solving techniques used in competitive programming.
- To enable students to apply data structures, algorithms, graphs, dynamic programming, and number theory concepts for solving medium to advanced-level coding problems.
- To prepare students for national and global coding contests by developing speed, accuracy, debugging skills, and consistent practice on online judge platforms.

UNIT 1: Basics of C++/Python, input–output, loops, conditionals, functions. Introduction to time–space complexity and constraints. Arrays, strings, frequency maps, simple logic problems. Basics of STL (vector, set, map, pair). Practice: 40–50 beginner problems on LeetCode/CodeChef.

UNIT 2: Core Data Structures & Techniques: Stacks, queues, linked lists, heaps, hash maps. Two-pointer, sliding window, prefix sums, hashing. Binary search and sorting-based logic patterns. Practice on GFG DS Track & LeetCode/CodeChef Data Structures. Target: 30–40 DS-based problems.

UNIT 3: Graphs & Dynamic Programming, Graph basics: BFS, DFS, components, cycles. Shortest paths: Dijkstra, Bellman-Ford, MST algorithms. Recursion, memoization, DP introduction. DP topics: knapsack, LIS, LCS, grid DP.Practice: 30-40 graph & DP problems on LeetCode/CodeChef.

UNIT 4: Advanced Algorithms & Number Theory, Sieve, modular arithmetic, combinatorics, modular inverse. Fast exponentiation, prime factorization. Greedy strategies and optimization techniques. Bit manipulation & bitmask DP. Practice: 30–40 advanced-level problems across platforms.

UNIT 5: Contest Practice & Deployment Registration on LeetCode, CodeChef, GFG, HackerRank. Weekly contests, mock tests, debugging strategies. Editorial writing and clean coding practices. Mini-project: CP notebook/GitHub repository. Target: 30–40 problems + 6 mock contests.

Reference Books

1. "Competitive Programming 4" by Steven Halim & Felix Halim (2018).
2. "Guide to Competitive Programming" by Antti Laaksonen (Springer, 2017).
3. "Programming Challenges: The Competitive Programmer's Handbook" by Laaksonen (Revised Edition 2016).
4. Elements of Programming Interviews in C++" by Adnan Aziz et al. (2016).

COURSE OUTCOMES

After completion of this course, the students would be able to:

CO1	Solve beginner-level CP problems using programming fundamentals.
CO2	Apply appropriate data structures and algorithmic techniques for intermediate problems.
CO3	Use graph algorithms and dynamic programming to solve complex problems.
CO4	Implement advanced number theory, greedy methods, and optimization in contest environments.
CO5	Participate effectively in coding contests and maintain consistent problem-solving practice.

Centre for Artificial Intelligence

CO-PO Mapping Matrix													
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
CO1	3	2	2	1	1							1	1	1
CO2	3	2	3	2	1							1	2	2
CO3	3	2	3	2	2		1		1	1	1	2	2	2
CO4	2	2	2	2	2	1	1	1	2	1	1	2	2	2
CO5	2	2	2	2	2	1	1	1	1	1	1	2	1	1

Centre for Artificial Intelligence

ROBOT KINEMATICS LAB

(24242206)

List of Programs/Experiments

1. Study of kinematic links, pairs and chains
2. To find the degree of freedom of a given mechanism.
3. To Study Straight Line Mechanism.
4. Study of Open and Closed kinematic chain mechanism:
    a. Oldham Coupling Mechanism,
    b. A quick return mechanism and
    c. CAM follower mechanism.
5. Experimentation on RoboAnalyzer software.
6. Validation of the forward kinematics of manipulators through RoboAnalyzer.
    a. Demonstration of 2D, 3D Transformation, Scaling
    b. Demonstration Rotation,
    c. Demonstration Translation,
    d. Demonstration Multiple transformation, and Homogeneous Transformations
7. Kinematics analysis of 2R Manipulators by RoboAnalyser.
8. Demonstration of D-H convention for kinematic analysis.
9. Kinematics of PUMA 560: Robot teaching
10. Study of adaptive controllers.

COURSE OUTCOMES

After completion of this course, the students would be able to:

CO1	Apply the fundamental principles of kinematic links, pairs, mechanisms, transformations, and D–H parameters to analyze robotic manipulators.
CO2	Implement forward kinematics and kinematic analysis of serial manipulators using RoboAnalyzer and validate results experimentally.
CO3	Evaluate basic robotic control concepts such as robot teaching and PID control through laboratory experiments.

CO-PO Mapping Matrix														
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
CO1	3	2	2	1	2							1	1	1
CO2	2	2	1	3	3							2	1	1
CO3	1	2	2	2	2	1	1	1	1	1	1	2	2	2

Centre for Artificial Intelligence

DATA SCIENCE LAB

(27242206)

List of Programs/Experiments

1. Basic Python syntax, variables, data types, control structures, functions.
2. Working with NumPy: Array creation, indexing, slicing, reshaping. Vectorized operations and broadcasting.
3. Data Manipulation using Pandas: Creating DataFrames and Series. Handling missing values, filtering, sorting, merging, grouping.
4. Data Import and Export: Loading CSV, Excel, JSON, and SQL data. Writing datasets to files.
5. Exploratory Data Analysis (EDA) – Descriptive Statistics: Summary statistics, distributions, cross-tabulation. Detecting outliers.
6. Data Visualization using Matplotlib: Line, bar, histogram, scatter, box plots. Subplots and styling.
7. Data Visualization using Seaborn: Pairplot, heatmap, countplot, distplot. Correlation matrix visualization.
8. Data Cleaning and Preprocessing: Handling missing data, encoding, scaling, normalization. Date-time processing.
9. Feature Engineering: Creating new features. Binning, one-hot encoding, label encoding. Dimensionality reduction basics (PCA).
10. Linear Regression: Implement simple and multiple linear regression. Visualize regression lines. Evaluate using RMSE, MAE, R^2.
11. Logistic Regression: Binary classification using logistic regression. Confusion matrix, accuracy, precision, recall, F1-score.
12. Decision Trees & Random Forests: Building decision tree classifiers/regressors. Applying ensemble learning (Random Forest). Feature importance analysis.
13. K-Means Clustering: Implement K-Means. Elbow method for optimal K. Cluster visualization.
14. Naïve Bayes Classifier: Text classification (spam detection or sentiment analysis).
15. Model Evaluation & Validation: Train-test split, k-fold cross-validation. Hyperparameter tuning with GridSearchCV.
16. Time Series Analysis (Basic): Plotting time series. Moving averages, decomposition.

COURSE OUTCOMES

After completion of this course, the students would be able to:

CO1	Apply Python programming and essential data manipulation techniques
CO2	Implement exploratory data analysis (EDA) and visualize datasets
CO3	Build and evaluate machine learning models

CO-PO Mapping Matrix														
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
CO1	3	2	2	2	2							1	1	2
CO2	3	3	3	3	3	1						2	3	3
CO3	3	3	3	3	3	2	1	1	1	1	1	3	3	3

Centre for Artificial Intelligence

MACHINE LEARNING LAB

(28242206)

1. Introduction to Python for ML: Basics of Python, NumPy, Pandas, Matplotlib, Seaborn.
2. Data Preprocessing: Handling missing values, outlier detection. Encoding (label, one-hot), feature scaling (standardization/normalization).
3. Exploratory Data Analysis (EDA): Descriptive statistics, distribution plots, correlation heatmaps.
4. Train-Test Split and Cross-Validation: Dataset partitioning and performing k-fold cross-validation.
5. Linear Regression: Simple & Multiple linear regression. Model evaluation: RMSE, MAE, R^2 score.
6. Logistic Regression: Binary classification using logistic regression. Confusion matrix, ROC curve, AUC.
7. K-Nearest Neighbors (KNN): Classification using distance metrics. Optimal K selection using cross-validation.
8. Naïve Bayes Classifier: Gaussian/Multinomial NB on text or tabular data. Spam detection / sentiment classification.
9. Support Vector Machines (SVM): Linear and RBF kernel. Hyperparameter tuning using GridSearchCV.
10. Decision Trees: Classification & regression trees. Visualizing the tree. Impact of max_depth, min_samples_split.
11. Random Forests & Ensemble Methods: Bagging, Random Forest, and feature importance. Comparing single tree vs ensemble performance.
12. K-Means Clustering: Clustering workflow. Elbow method and silhouette score.
13. Hierarchical Clustering: Agglomerative clustering. Dendrogram visualization.
14. Dimensionality Reduction (PCA): Eigenvalues/eigenvectors, explained variance ratio. Visualization in 2D space.
15. Artificial Neural Networks (ANN) – Basic: Building a simple feedforward network using TensorFlow/Keras or Scikit-learn. Training, validation, accuracy evaluation.
16. Model Evaluation & Hyperparameter Tuning: GridSearchCV, RandomizedSearchCV. Bias-variance tradeoff.

COURSE OUTCOMES

After completion of this course, the students would be able to:

CO1	Apply data preprocessing, cleaning, and preparation techniques essential for machine learning tasks.
CO2	Implement supervised & unsupervised learning algorithms and evaluate them using appropriate performance metrics.
CO3	Develop an end-to-end machine learning workflow

CO-PO Mapping Matrix														
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
CO1	3	2	2	2	2							1	1	2
CO2	3	3	3	3	3	1						2	3	3
CO3	3	3	3	3	3	2	1	1	1	1	1	3	3	3

Centre for Artificial Intelligence

MACRO PROJECT-II

(31242210/ 24242210/ 27242210/ 28242210)

1. Intelligent Student Performance Prediction System
2. Smart Healthcare Diagnosis Assistant
3. Online Book Recommendation System
4. Intelligent Traffic Management System
5. E-Commerce Customer Behavior Analytics System
6. Crop Disease Detection & Yield Prediction System
7. Banking Fraud Detection System
8. Job Recommendation and Skill Assessment Portal
9. Smart Energy Consumption Monitor
10. Intelligent Chatbot for Academic Queries
11. Library Management System with Book Recommendation
12. Sports Analytics and Player Performance Predictor
13. Transport Fare Estimation and Route Optimization System
14. Smart Attendance System with Face Recognition
15. Loan Eligibility & Default Prediction System
16. Intelligent Inventory Management & Demand Forecasting
17. Accident Detection & Emergency Response System
18. Student Mentorship and Course Recommendation Engine
19. Social Media Sentiment Analysis Tool
20. Real Estate Price Prediction & Property Finder
21. Hospital Resource Optimization System
22. Personalized Fitness & Nutrition Recommendation System
23. Vehicle Number Plate Recognition System
24. Smart Banking Chatbot with Predictive Insights
25. Intelligent Complaint Management System
26. Indoor Navigation System for Large Buildings
27. Blood Donation & Matching System
28. Cyberbullying Detection System
29. Smart Tourism Recommendation System
30. Intelligent Resume Screening and Ranking System