TECHNO INDIA UNIVERSITY

WEST BENGAL

Syllabus

for

4-Years B.Tech.

in

Computer Science and Engineering
(Specialization in Artificial Intelligence)

Academic Year: 2024-2025

Semester 6

Operations Research & Optimization Techniques (TIU-UMA-T302)

Program: B. Tech. in CSE-AI	**Year, Semester:** 3rd., 6th
Course Title: Operations Research & Optimization Techniques	**Subject Code:** TIU-UMA-T302
Contact Hours/Week: 3–0–0 (L–T–P)	**Credit:** 3

COURSE OBJECTIVE:

Enable the student to:

1. understand the importance and value of Operations Research in real life, and finding solutions to various real-life problems
2. formulate linear programming problem from verbal description, and finding solutions
3. learn the basics in the field of game theory and solution techniques of various problems
4. choose the appropriate queuing model for a given practical application and finding solutions
5. draw a network diagram and determine related time, path, etc.

COURSE OUTCOME:

On completion of the course, the student will be able to:

CO-1:	analyze any real-life system with limited constraints, present it in a linear programming form and hence find its solution.	K4
CO-2:	solve both balanced and unbalanced transportation problems using various methods.	K3
CO-3:	determine solutions of a variety of problems mathematically such as assignment, travelling salesman etc. and associate how real-life problems are depicted.	K4
CO-4:	formulate game models and solve them by utilizing different methods.	K4
CO-5:	choose the appropriate queuing situations and deduce the optimal solutions using models for different situations.	K3
CO-6:	construct network diagrams for service and manufacturing systems, and find related time, path, etc.	K4

COURSE CONTENT:

MODULE 1:	**Linear Programming**	**14 Hours**
Formulation of Linear Programming Problem, Linear dependence and independence, Basic solutions, Convex Sets, Graphical Method, Simplex Method		

MODULE 2:	**Transportation and Assignment Problem**	**12 Hours**
Formulation of Transportation Problem, Initial Feasible Solution Methods, Optimality Test, Degeneracy in TP; Assignment Problem, Hungarian Method, Travelling Salesman Problem		
MODULE 3:	**Game Theory**	**8 Hours**
Two Person Zero Sum Game, Pure and Mixed Strategies, Algebraic Solution Procedure, Graphical Solution		
MODULE 4:	**Introduction to Queuing Models**	**5 Hours**
Elements of Queuing Model, Pure Birth Death Model.		
MODULE 5:	**Network Analysis**	**6 Hours**
CPM review, Crashing of an activity, Crash-cost slope, Time-cost trade		
TOTAL LECTURES		**45 Hours**

Books:

1. Ghosh, M. K., & Chakraborty, S. (2010). Linear programming and game theory. Prentice-Hall of India.
2. Taha, H. A. (2017). Operations research: An introduction (10th ed.). Pearson.
3. Sharma, J. K. (2017). Operations research: Theory and applications (6th ed.). Macmillan Publishers India.
4. Sharma, S. D. (2009). Operations research. Kedar Nath Ram Nath.
5. Swarup, K., Gupta, P. K., & Man Mohan. (2014). Operations research. Sultan Chand & Sons.

Computer Network (TIU-UCS-T304)

Program: B. Tech. in CSE-AI	**Year, Semester:** 3rd., 6th
Course Title: Computer Network	**Subject Code:** TIU-UCS-T304
Contact Hours/Week: 3–0–0 (L–T–P)	**Credit:** 3

COURSE OBJECTIVE :

Enable the student to:

1. understand network fundamentals including network types and topologies.
2. Analyze Network Protocols including TCP/IP, UDP, HTTP, FTP, and DNS, and understand their roles in data communication.
3. Explore OSI and TCP/IP Models and how data flows through different network layers.
4. Implement Routing and Switching Techniques including static and dynamic routing protocols

COURSE OUTCOME :

The students will be able to:

CO1:	Describe the general principles of data communication, the concept of the layered approach	K2

CO2:	Describe how computer networks are organized with the concept of layered approach	K2
CO3:	Design logical sub-address blocks with a given address block and network topology	K3
CO4:	Understanding of simple LAN with hubs, bridges, and switches	K2
CO5:	Describe how routing protocols work	K3
CO6:	Understand network security threats and basic security mechanisms to protect data and communication.	K2

COURSE CONTENT :

MODULE 1:		**10 Hours**
Network hardware, Network software, OSI, TCP/IP Reference models, Example Networks: ARPANET, Internet. Physical Layer: Guided Transmission media: twisted pairs, coaxial cable, fiber optics, Wireless transmission.		
MODULE 2:		**12 Hours**
Data link layer: Design issues, framing, Error detection and correction. Elementary data link protocols: simplex protocol, A simplex stop and wait protocol for an error-free channel, A simplex stop and wait protocol for noisy channels. Sliding Window protocols: A one-bit sliding window protocol, A protocol using Go-Back-N, A protocol using Selective Repeat, Example data link protocols. Medium Access sublayer: The channel allocation problem, Multiple access protocols: ALOHA, Carrier sense multiple access protocols, collision free protocols. Wireless LANs, Data link layer switching.		
MODULE 3:		**10 Hours**
Network Layer: Design issues, Routing algorithms: shortest path routing, Flooding, Hierarchical routing, Broadcast, Multicast, distance vector routing, Congestion Control Algorithms, Quality of Service, Internetworking, The Network layer in the internet.		
MODULE 4:		**8 Hours**
Transport Layer: Transport Services, Elements of Transport protocols, Connection management, TCP and UDP protocols.		
MODULE 5:		**5 Hours**
Application Layer –Domain name system, SNMP, Electronic Mail; the World WEB, HTTP, Streaming audio and video.		
	TOTAL LECTURES	**45 Hours**

Books:

1. A. S. Tanenbaum and D. J. Wetherall, “Computer Networks”, Pearson, 5th Edition, 2010, ISBN-10: 0132126958, ISBN-13: 978-0132126953.
2. B. A. Forouzan, “Data Communications and Networking”, McGraw-Hill Education, 5th Edition, 2012, ISBN-10: 0073376221, ISBN-13: 978-0073376226.
3. J. F. Kurose and K. W. Ross, “Computer Networking: A Top-Down Approach”, Pearson, 8th Edition, 2021, ISBN-10: 0136681553, ISBN-13: 978-0136681557.

4. W. Stallings, "Data and Computer Communications", Pearson, 10th Edition, 2013, ISBN-10: 0133506487, ISBN-13: 978-0133506488.
5. D. E. Comer, "Computer Networks and Internets", Pearson, 6th Edition, 2014, ISBN-10: 0133587932, ISBN-13: 978-0133587937.
6. M. A. Gallo and W. M. Hancock, "Computer Communications and Networking Technologies", Cengage Learning, 1st Edition, 2001, ISBN-10: 053437130X, ISBN-13: 978-0534371305.

Compiler Design (TIU-UCS-T320)

Program: B. Tech. in CSE-AI	**Year, Semester:** 3rd, 6th
Course Title: Compiler Design	**Subject Code:** TIU-UCS-T320
Contact Hours/Week: 3–0–0 (L–T–P)	**Credit:** 3

COURSE OBJECTIVE:

1. To make the student aware about the basic concepts, i.e. different phases such as lexical analysis, syntax analysis, semantic analysis and code generation of compiler.
2. The students should know the different functionalities of compiler.
3. To make the students aware about the possible errors that can occur at different phases and how they can be addressed.
4. Make the students aware about the tools LEX and YACC.

COURSE OUTCOME:

The students will be able to:

CO1:	Understand fundamentals of language parser and identify the relationships among different phases of compiler	K2
CO2:	Illustrate the use of different types of parsers and their constructions, production rules and language semantics.	K3
CO3:	Inherited and synthesized attributes with their evaluations, run time storage allocation.	K3
CO4:	Describe techniques for intermediate code generation and code optimization.	K2
CO5:	Analyze error detection and recovery techniques in different compiler phases.	K3
CO6:	Implement and evaluate code generation techniques for efficiency.	K4

COURSE CONTENT:

MODULE 1:	**COMPILER STRUCTURE**	**3 Hours**
Analysis-synthesis model of compilation, various phases of a compiler, tool-based approach to compiler construction.		
MODULE 2:	**LEXICAL ANALYSIS**	**6 Hours**
Interface with input, parser and symbol table, token, lexeme and patterns, difficulties in lexical analysis, error reporting, and implementation. Regular definition, Transition diagrams, LEX		
MODULE 3:	**SYNTAX ANALYSIS**	**18 Hours**

Context free grammar, ambiguity, associativity, precedence, top-down parsing, recursive descent parsing, transformation on the grammars, predictive parsing, Bottom-up parsing, operator precedence grammars, LR parsers (SLR, LALR, LR), YACC.		
MODULE 4:	**SYNTAX DIRECTED DEFINITIONS**	**3 Hours**
Inherited and synthesized attributes, dependency graph, evaluation order, bottom up and top-down evaluation of attributes, L- and S-attributed definitions		
MODULE 5:	**TYPE CHECKING**	**2 Hours**
Type system, type expressions, structural and name equivalence of types, type conversion, overloaded functions and operators, polymorphic functions.		
MODULE 6:	**RUN TIME SYSTEM**	**2 Hours**
Storage organization, activation tree, activation record, parameter passing, Symbol table, dynamic storage allocation. Intermediate code generation: Intermediate representations, translation of declarations, assignments Intermediate Code generation for control flow, Boolean expressions and procedure calls, implementation issues.		
MODULE 7:	**CODE GENERATION AND INSTRUCTION SELECTION**	**6 Hours**
Issues, basic blocks and flow graphs, register allocation, code generation DAG representation of programs, code generation from DAGs, peep-hole optimization, code generator generators, specifications of machine		
MODULE 8:	**CODE OPTIMIZATION**	**5 Hours**
Source of optimizations, and optimization of basic blocks, loops, global dataflow analysis, and solution to iterative data flow equations. Code improving transformations, dealing with aliases, data flow analysis of structured flow graphs.		
TOTAL LECTURES		**45 Hours**

Books:

1. Aho, Ullman,Sethi and Lam, Principles of Compiler Design, Pearson Education
2. Holub, Compiler Design in C, PHI
3. Andrew L. Appel, Modern Compiler Implementation in C, Foundation Books, Delhi
4. Dick Gruneet. Al., Modern Compiler Design, Wiley Dreamtech
5. S. Chattopadhyay, Compiler Design, PHIS. Pal: Systems Programming, Oxford University Press

Software Engineering (TIU-UCS-T314)

Program: B. Tech. in CSE-AI	**Year, Semester**: 3rd, 6th
Course Title: Software Engineering	**Subject Code**: TIU-UCS-T314
Contact Hours/Week: 3–0–0 (L–T–P)	**Credit**: 3

COURSE OBJECTIVE

1. To develop basic Knowledge in Software Engineering and its applications.
2. To understand software Engineering layered architecture and the process framework.
3. To analyze software process models such as the waterfall, spiral, evolutionary models and agile method for software development.
4. To design software requirements and specifications of documents.

COURSE OUTCOME

The students will be able to:

CO1:	Identify and describe fundamental software engineering concepts, principles and models.	K2
CO2:	Analyze and document software requirements using appropriate elicitation techniques and requirement engineering processes.	K4
CO3:	Design software solutions using modeling techniques such as UML, architectural styles, and design patterns.	K3
CO4:	Implement software applications by applying programming principles, coding standards, and development methodologies.	K3
CO5:	Evaluate software quality through testing strategies, verification, validation, and project management techniques.	K4
CO6:	Demonstrate teamwork, ethical considerations, and professional responsibility in software development projects.	K4

COURSE CONTENT

MODULE 1:	**FOUNDATIONS OF SOFTWARE ENGINEERING**	**12 Hours**
Introduction to software engineering: Software and software engineering, phases in software development, software development process models, role of management in software development, role of metrics and measurement.		
MODULE 2:	**SOFTWARE REQUIREMENTS AND PROJECT PLANNING**	**12 Hours**
Software requirement specifications: Role of SRS, problem analysis, requirement specification, validation, metrics, monitoring and control. Planning a software project: Cost estimation, project scheduling, staffing, personal planning, team structures, SCM, quality assurance plans, project-monitoring plans, risk management, Knowledge driven approach and development.		
MODULE 3:	**SYSTEM AND DETAILED DESIGN**	**7 Hours**
System design: Design objectives, design principles, module level concepts, design methodology, structured design, design specifications, verification metrics, monitoring and control. Detailed design: Module specification, detailed design and process design language, verification.		
MODULE 4:	**CODING AND TESTING**	**7 Hours**

Coding: Programming practice, verification, and metrics. Testing: Testing fundamentals, functional testing, structural testing, testing process, comparison of different V & V techniques.		
MODULE 5:	**SOFTWARE QUALITY AND RELIABILITY**	**7 Hours**
Software quality; Garvin's quality dimensions, McCall's quality factor, ISO 9126 quality factor; Software Quality Dilemma; Introduction to Capability Maturity Models (CMM and CMMI); Introduction to software reliability, reliability models and estimation.		
	TOTAL LECTURES	**45 Hours**

Books:

1. Roger S Pressman, Software Engineering-A Practitioners Approach, McGraw Hill Publications.
2. Pankaj Jalote, An Integrated Approach to Software Engineering, BPB Publications
3. Rajib Mall, Fundamentals of Software Engineering, PHI Learning Private Limited
4. Software Engineering, Ian Sommerville

Artificial Intelligence (TIU-UCS-T350)

Program: B. Tech. in CSE-AI	**Year, Semester:** 3rd, 6th
Course Title: Artificial Intelligence	**Subject Code:** TIU-UCS-T350
Contact Hours/Week: 3–0–0 (L–T–P)	**Credit:** 3

COURSE OBJECTIVES:

1. Understand the core concepts, history and evolutions of Artificial Intelligence
2. Explore search and optimization techniques like heuristic and uninformed search, evolutionary algorithms
3. Develop Logical and Probabilistic Reasoning like Bayesian network, knowledge representations
4. Apply Machine Learning Concepts like supervised, unsupervised, reinforcement learning

COURSE OUTCOMES:

The students will be able to

CO1:	Understand the fundamental concepts Artificial Intelligence such as knowledge representation, problem solving and expert systems	K2
CO2:	Understand the use of AI to solve communication problems using Natural Language Processing	K2
CO3:	Develop knowledge of decision making and learning methods.	K3
CO4:	develop new facts from existing knowledge base using resolution and unification.	K4
CO5:	Demonstrate the way of writing Facts and Rules in order to solve some	K4

	problems based on rules and to develop systems for question-answer.	
CO6:	Apply AI techniques like heuristic search, genetic algorithms, and neural networks to solve real-world problems.	K3

Course Content

MODULE 1:	BASICS OF AI	8 Hours
Introduction: Overview of Artificial intelligence- Problems of AI, AI technique, Tic - Tac - Toe problem., Intelligent Agents: Agents & environment, nature of environment, structure of agents, goal-based agents, utility-based agents, learning agents., Learning: Forms of learning, inductive learning, learning decision trees, explanation-based learning, learning using relevant information, neural net learning & genetic learning.		
MODULE 2:	**DIFFERENT TYPES OF SEARCHING ALGORITHMS, PROBLEM SOLVING**	**13 Hours**
Problems, Problem Space & search: Defining the problem as state space search, production system, constraint satisfaction problems, issues in the design of search programs, Search techniques: Solving problems by searching: Problem solving agents, searching for solutions; uniform search strategies: breadth first search, depth first search, depth limited search, bidirectional search, comparing uniform search strategies, Heuristic search strategies: Greedy best-first search, A* search, memory bounded heuristic search: local search algorithms & optimization problems: Hill climbing search, simulated annealing search, local beam search, genetic algorithms; constraint satisfaction problems, local search for constraint satisfaction problems, MCTS-Monte Carlo Tree Search, Adversarial search: Games, optimal decisions & strategies in games, the minimax search procedure, alpha-beta pruning, additional refinements, iterative deepening.		
MODULE 3:	**KNOWLEDGE & REASONING, KNOWLEDGE & REASONING**	**12 Hours**
Knowledge representation issues, representation & mapping, approaches to knowledge representation, issues in knowledge representation, Using predicate logic: Representing simple fact in logic, representing instant & ISA relationship, computable functions & predicates, resolution, natural deduction, Representing knowledge using rules: Procedural vs declarative knowledge, logic programming, forward vs backward reasoning, matching, control knowledge, Probabilistic reasoning: Representing knowledge in an uncertain domain, the semantics of Bayesian networks, Fuzzy sets, and fuzzy logics, belief propagation. Markov processes, and Hidden Markov models.		
MODULE 4:	**DIFFERENT FIELDS OF AI, NATURAL LANGUAGE PROCESSING**	**12 Hours**
Introduction, Syntactic processing, semantic analysis, discourse, and pragmatic processing., Expert Systems: Representing and using domain knowledge, expert system shells, and		

knowledge acquisition. Basic knowledge of programming languages like Prolog.	
TOTAL LECTURES	**45 Hours**

Books:

1. Artificial Intelligent e: Elaine Rich, Kevin Knight, Mc-Graw Hill.
2. Introduction to AI & Expert System: Dan W. Patterson, PHI.
3. Artificial Intelligent by Luger (Pearson Education)
4. Russel & Norvig, Artificial Intelligent e: A Modern Approach, Pearson Education

Computer Networks Lab (TIU-UCS-L394)

Program: B.Tech. in CSE-AI	**Year, Semester:** 3rd, 6th.
Course Title: Computer Networks Lab	**Subject Code:**TIU-UCS-L394
Contact Hours/Week: 0–0–3	**Credit:** 1.5

COURSE OBJECTIVE :

Enable the student to:

1. Enumerate various network topologies and identify situations when different network topologies would be useful.
2. Explain and apply error control mechanisms to ensure reliable data transmission in computer networks.

COURSE OUTCOME :

On completion of the course, the student will be able:

CO-1	Explain the error control mechanisms in computer network infrastructure.	K3
CO-2	Identify and describe the network layers, structure/format, and the role of each network layer.	K3
CO-3	Design and implement various network applications such as data transmission.	K4
CO-4	Illustrate the connectivity and data transmission between client and server in real-time multimedia transmission.	K3
CO-5	Distinguish and explain various routing protocols, algorithms, and internetworking mechanisms.	K4
CO-6	Evaluate and troubleshoot network performance, addressing issues related to bandwidth, latency, and network reliability in both local and wide-area networks.	K3

COURSE CONTENT :

MODULE 1:	**INTRODUCTION TO NETWORK**	**9 Hours**
Network hardware, Network software, OSI, TCP/IP Reference models, Example Networks: ARPANET, Internet. Physical Layer: Guided Transmission media: twisted pairs, coaxial cable, fiber optics, Wireless transmission.		
MODULE 2:	**DATA LINK LAYER**	**10 Hours**

Data link layer: Design issues, framing, Error detection and correction. Elementary data link protocols: simplex protocol, A simplex stop and wait protocol for an error-free channel, A simplex stop and wait protocol for noisy channels. Sliding Window protocols: A one-bit sliding window protocol, A protocol using Go-Back-N, A protocol using Selective Repeat, Example data link protocols. Medium Access sublayer: The channel allocation problem, Multiple access protocols: ALOHA, Carrier sense multiple access protocols, collision free protocols. Wireless LANs, Data link layer switching.		
MODULE 3:	**NETWORK LAYER**	**9 Hours**
Design issues, Routing algorithms: shortest path routing, Flooding, Hierarchical routing, Broadcast, Multicast, distance vector routing, Congestion Control Algorithms, Quality of Service, Internetworking, The Network layer in the internet.		
MODULE 4:	**TRANSPORT LAYER**	**9 Hours**
Transport Services, Elements of Transport protocols, Connection management, TCP and UDP protocols.		
MODULE 5:	**APPLICATION LAYER**	**8 Hours**
Domain name system, SNMP, Electronic Mail; the World WEB, HTTP, Streaming audio and video.		
TOTAL LAB HOURS		**45 Hours**

Books:

1. Tanenbaum, A. S., & Wetherall, D. J. (2010). Computer Networks (5th ed.). Pearson.
2. Forouzan, B. A. (2017). Data Communications and Networking (5th ed.). McGraw-Hill Education.
3. Stallings, W. (2020). Data and Computer Communications (11th ed.). Pearson.
4. Comer, D. E. (2018). Computer Networks and Internets (6th ed.). Pearson.
5. Kurose, J. F., & Ross, K. W. (2021). Computer Networking: A Top-Down Approach (8th ed.). Pearson.
6. Peterson, L. L., & Davie, B. S. (2021). Computer Networks: A Systems Approach (6th ed.). Morgan Kaufmann.

Software Engineering Lab (TIU-UCS-L352)

Program: B. Tech. in CSE-AI	**Year, Semester:** 3rd, 6th
Course Title: Software Engineering Lab	**Subject Code:** TIU-UCS-L352
Contact Hours/Week: 0–0–3 (L–T–P)	**Credit:** 1.5

COURSE OBJECTIVE

Enable the student to:

1. Analyze software process models such as the waterfall, spiral, evolutionary models and agile method for software development.
2. Design software requirements and specifications of documents, project planning, scheduling, cost estimation, risk management.

3. Describe data models, object models, context models, behavioral models, coding style and testing issues. Also to know about the quality checking mechanism for software processes and products.

COURSE OUTCOME

On completion of the course, the student will be able:

CO1	Identify and examine requirements from problem statements to construct software solutions.	K3
CO2	Develop and design software solutions using UML modeling techniques.	K4
CO3	Illustrate and apply software engineering principles to organize and manage software projects effectively.	K4
CO4	Test and validate software systems using appropriate testing strategies and tactics.	K3
CO5	Implement and assess software metrics to improve the quality and maintainability of software products.	K4
CO6	Demonstrate ethical, social, and legal responsibilities in software development and ensure compliance with industry standards.	K3

COURSE CONTENT:

MODULE 1:	**INTRODUCTION TO SOFTWARE ENGINEERING**	**9 Hours**
Introduction to software engineering: Software and software engineering, phases in software development, software development process models, role of management in software development, role of metrics and measurement.		
MODULE 2:	**REQUIREMENT ANALYSIS AND SPECIFICATION, PROJECT MANAGEMENT**	**12 Hours**
Software requirement specifications: Role of SRS, problem analysis, requirement specification, validation, metrics, monitoring and control. Planning a software project: Cost estimation, project scheduling, staffing, personal planning, team structures, SCM, quality assurance plans, project-monitoring plans, risk management, Knowledge driven approach and development.		
MODULE 3:	**SOFTWARE DESIGN**	**9 Hours**
System design: Design objectives, design principles, module level concepts, design methodology, structured design, design specifications, verification metrics, monitoring and control. Detailed design: Module specification, detailed design and process design language, verification.		
MODULE 4:	**CODING AND TESTING**	**9 Hours**
Coding: Programming practice, verification, and metrics. Testing: Testing fundamentals, functional testing, structural testing, testing process, comparison of different V & V techniques.		
MODULE 5:	**SOFTWARE QUALITY**	**6 Hours**
Software quality; Garvin's quality dimensions, McCall's quality factor, ISO 9126 quality factor; Software Quality Dilemma; Introduction to Capability Maturity Models (CMM and CMMI);		

Introduction to software reliability, reliability models and estimation.	
TOTAL LAB HOURS	**45 Hours**

Books:

1. Software Engineering, Ian Sommerville
2. R. Mall, "Fundamentals of Software Engineering", Prentice Hall of India
3. R. S. Pressman, "Software Engineering: A Practitioner's Approach", Tata McGraw Hill
4. D. Bell, "Software Engineering for Students", Pearson

Artificial Intelligence Lab (TIU-UCS-L350)

Program: B.Tech. in CSE-AI	**Year, Semester:** 3rd, 6th.
Course Title: Artificial Intelligence Lab	**Subject Code:**TIU-UCS-L350
Contact Hours/Week: 0–0–3	**Credit:** 1.5

COURSE OBJECTIVE:

Enable the student to:

1. Understand and implement fundamental Artificial Intelligence techniques, including search algorithms, optimization, and decision-making.
2. Develop AI-based problem-solving strategies using heuristic and adversarial search techniques.

COURSE OUTCOME:

On completion of the course, the student will be able:

CO-1	Implement and analyze graph traversal algorithms using Python.	K3
CO-2	Apply heuristic search algorithms to solve AI problems efficiently.	K3
CO-3	Develop and implement adversarial search algorithms for game playing.	K4
CO-4	Utilize optimization techniques for solving real-world AI problems.	K4
CO-5	Implement intelligent agent models and logical reasoning techniques.	K3
CO-6	Apply fuzzy logic systems for decision-making applications.	K4

COURSE CONTENT:

MODULE 1:	**GRAPH SEARCH ALGORITHMS**	**9 Hours**
Introduction to graph traversal techniques, Breadth-First Search (BFS) and Depth-First Search (DFS), Implementation of BFS and DFS in Python, Application of BFS in solving the Water Jug Problem.		
MODULE 2:	**INFORMED SEARCH ALGORITHMS**	**6 Hours**
Introduction to heuristic search, A* Algorithm: concept, working, and implementation, Application of A* Algorithm in solving the 8-Puzzle Problem.		
MODULE 3:	**GAME PLAYING AND DECISION MAKING**	**9 Hours**

Introduction to adversarial search, Min-Max Algorithm and its application in game playing, Implementation of Tic-Tac-Toe using Min-Max Algorithm, Heuristic optimization with Alpha-Beta Pruning.		
MODULE 4:	**OPTIMIZATION TECHNIQUES**	**6 Hours**
Introduction to optimization algorithms, Genetic Algorithms (GA): concept and working principle, Implementation of the Traveling Salesman Problem using a Simple Genetic Algorithm.		
MODULE 5:	**INTELLIGENT AGENTS AND LOGICAL REASONING**	**9 Hours**
Types of intelligent agents, Rational Agents: Simple Reflex and Model-Based Reflex, Implementation of First-Order Predicate Logic (FOPL) in AI systems.		
MODULE 6:	**FUZZY LOGIC AND DECISION MAKING**	**6 Hours**
Introduction to fuzzy logic systems, Fuzzification and Defuzzification techniques, Implementation of Type-1 Fuzzy Logic in decision-making problems.		
TOTAL LAB HOURS		**45 Hours**

Books:

1. Artificial Intelligent e: Elaine Rich, Kevin Knight, Mc-Graw Hill.
2. Introduction to AI & Expert System: Dan W. Patterson, PHI.
3. Artificial Intelligent by Luger (Pearson Education)
4. Russel & Norvig, Artificial Intelligent e: A Modern Approach, Pearson Education

Mini Project (TIU-UCS-P302)

Program: B.Tech. in CSE-AI	**Year, Semester:** 3rd, 6th.
Course Title: Mini Project	**Subject Code:**TIU-UCS-P302
Contact Hours/Week: 0–4–8	**Credit:** 2

COURSE OBJECTIVE :

Enable the student to:

1. Introduce students to research methodologies and techniques for identifying and formulating research problems in computer science.
2. Equip students with the ability to conduct a structured literature review and critically analyze existing research.
3. Develop students' skills in identifying research gaps and formulating clear, well-defined research objectives aligned with industry and academic needs..

COURSE OUTCOME :

On completion of the course, the student will be able:

CO-1	Design and implement a small-scale software or hardware-based solution.	K6
CO-2	Apply appropriate tools, algorithms, or frameworks to develop a functional prototype.	K3

CO-3	Test and evaluate the performance of the developed system.	K5
CO-4	Analyze challenges encountered during project development and propose solutions.	K4
CO-5	Effectively document the project and present findings through reports and presentations.	K4
CO-6	Collaborate effectively within a team to manage project timelines, resources, and deliverables.	K6

COURSE CONTENT :

Module-1	**PROJECT PLANNING & INITIAL DEVELOPMENT**
Identifying project topics based on real-world problems or emerging technologies. Gathering requirements and defining objectives. Selecting suitable development tools, languages, and frameworks, Evaluating model/system performance using metrics like accuracy, latency, security level, etc.	
Module-2	**PROTOTYPING & IMPLEMENTATION**
Developing the initial version of the software/hardware system. Integrating different components and debugging errors. Conducting basic performance evaluations and making iterative improvements.	
Module-3	**CONCLUSION & FUTURE SCOPE**
Validating the mini-project against expected outcomes, Summarizing research findings and project outcomes. Discussing industry impact and real-world applications.	

Natural Language Processing (NLP) and its Applications (TIU-CASD-UCS-S398A)

Program: B. Tech. in CSE-AI	**Year, Semester:** 3rd., 6th
Course Title: Natural Language Processing	**Subject Code:** TIU-CASD-UCS-S398A
Contact Hours/Week: 3–0–0 (L–T–P)	**Credit:** 3

COURSE OBJECTIVE:

1. Introducing cutting-edge systems and trends in natural language processing to the students.
2. Make sure they comprehend the language's morphology, syntax, semantics, and pragmatic notions and are able to provide the necessary examples to support the aforementioned ideas.
3. Teach them the importance of pragmatics in interpreting natural language.
4. Give students the tools they need to explain a natural language processing application and to demonstrate syntactic, semantic, and pragmatic processing.

COURSE OUTCOME:

The students will be able to:

CO1:	Explain the fundamental concepts of Natural Language Processing (NLP), including syntax, semantics, and pragmatics.	K2
CO2:	Apply various text preprocessing techniques such as tokenization, stemming, and lemmatization to prepare data for NLP tasks.	K 3
CO3:	Analyze different NLP models like N-grams, Hidden Markov Models (HMM), and neural networks to solve language-based problems.	K3
CO4:	Evaluate the performance of NLP algorithms using appropriate metrics (e.g., accuracy, precision, recall, and F1 score).	K4
CO5:	Design and implement NLP applications such as sentiment analysis, machine translation, and chatbots using modern frameworks (e.g., NLTK, Spacy, or TensorFlow).	K3
CO6:	Critically assess the ethical considerations and biases in NLP models and their real-world impact.	K3

COURSE CONTENT:

MODULE 1:	**Introduction to NLP**	**10 Hours**
Natural language processing issues and strategies. Tools of NLP, Linguistic organization of NLP, NLP as an Application domain. Word Classes: Regular Expressions: Chomsky hierarchy, CFG and different parsing techniques, Morphology: Inflectional, derivational, parsing and parsing with FST, Combinational Rules, Joint and conditional probability. Probabilistic Language modeling and its Applications.		
MODULE 2:	**Language Modeling and Naïve Bayes**	**14 Hours**
Markov models, N- grams. Estimating the probability of a word and smoothing. Counting words in Corpora, simple N-grams, smoothing (Add One, Written-Bell, Good-Turing). Part of Speech Tagging and Hidden Markov Models: Part of Speech tagging, Indian Language on focus Morphology Analysis, Accuracy Measure and Probability, HMM, Viterbi algorithm for finding most likely HMM Path. HMM tagging, transformation based tagging. Probabilistic Context Free Grammars: Weighted context free grammars.		
MODULE 3:	**Semantics**	**12 Hours**
Representing Meaning: Unambiguous representation, canonical form, expressiveness, meaning structure of language Semantic Analysis: NLP and IR, How NLP has used IR Towards Latent Semantic. Lexical Semantics: Lexemes(synonymy, hyponymy etc), WordNet, metonymy and their computational approaches Supervised and Unsupervised methods Word Sense Disambiguation: Selectional restriction based, machine learning based and dictionary based approaches.		
MODULE 4:	**Pragmatics**	**9 Hours**

Information Theory: Entropy, Cross-entropy, information gain. Reference resolution and phenomena, syntactic and semantic constraints. Pronoun resolution algorithm, text coherence, and discourse structure Natural Language Generation: Introduction to language generation, architecture, discourse planning (text schemata, rhetorical relations). Resource Constrained WSD, Parsing Algorithms, Parsing Ambiguous Sentences, Probabilistic Parsing Algorithms.	
TOTAL LECTURES	**45 Hours**

Books:

1. D. Jurafsky & J. H. Martin – "Speech and Language Processing – An introduction to Language processing, Computational Linguistics, and Speech Recognition", Pearson Education
2. Allen, James. 1995. – "Natural Language Understanding". Benjamin/Cummings, 2ed. Bharathi, A., Vineet Chaitanya and Rajeev Sangal. 1995.
3. Natural Language Processing- "A Pananian Perspective". Prentice Hall India, Eastern Economy Edition. 3. Eugene Cherniak: "Statistical Language Learning", MIT Press, 1993.
4. Manning, Christopher and Heinrich Schutze. 1999. "Foundations of Statistical Natural Language Processing". MIT Press.
5. Cognitively Inspired Natural Language Processing Abhijit Mishra, Pushpak Bhattacharyya Springer.

Modern Web Technologies (TIU-CASD-UCS-S398B)

Program: B. Tech. in CSE-AI	**Year, Semester:** 3rd , 6th
Course Title: Modern Web Technologies	**Subject Code**: TIU-CASD-UCS-S398B
Contact Hours/Week: 3–0–0 (L–T–P)	**Credit:** 3

COURSE OBJECTIVE:

Enable the student to:

1. Learn HTML5, CSS3, JavaScript, and responsive design.
2. Build full-stack apps with React, Node.js, and databases.
3. Optimize security, performance, and cloud deployment.

COURSE OUTCOME:

The student will be able to:

CO-1:	Understand modern web development fundamentals, including HTML5, CSS3, and JavaScript frameworks.	K2
CO-2:	Develop responsive web applications using front-end technologies like React.js, Vue.js, or Angular.	K3
CO-3:	Implement server-side programming with Node.js and Express.js for scalable applications.	K4
CO-4:	Manage dynamic data using databases like MongoDB and Firebase.	K3

CO-5:	Ensure web security and performance with authentication, authorization, and optimization techniques.	K3
CO-6:	Deploy and manage web applications using cloud platforms like AWS, Firebase, and Docker.	K4

COURSE CONTENT :

MODULE 1:	**Introduction to Web Technologies**	**7 Hours**
Basics of Internet, WWW, HTTP, HTTPS, HTML5, Semantic Elements, Forms, Multimedia, Canvas & SVG, CSS3, Grid, Flexbox, Animations, Media Queries, Introduction to JavaScript, ES6+ Features, DOM Manipulation		
MODULE 2:	**Frontend Development**	**7 Hours**
Responsive Web Design using Bootstrap/Tailwind CSS, JavaScript Frameworks, React.js, Vue.js, or Angular, Components, Props, State Management (Redux, Vuex), Routing and Navigation in SPA (Single Page Applications)		
MODULE 3:	**Backend Development**	**7 Hours**
Introduction to Node.js & Express.js, RESTful APIs, Building and Consuming APIs, Middleware, Authentication (JWT, OAuth), WebSockets for Real-time Communication.		
MODULE 4:	**Database Technologies**	**8 Hours**
SQL vs NoSQL Databases, MongoDB, Schema Design, CRUD Operations, Mongoose, Firebase Realtime Database & Firestore Integrating Frontend with Databases.		
MODULE 5:	**Web Security & Performance Optimization**	**8 Hours**
Security Concepts: CORS, CSRF, XSS, SQL Injection, Authentication and Authorization Techniques, Web Performance Optimization (Lazy Loading, Caching, Minification), Progressive Web Apps (PWAs) and Service Workers.		
MODULE 6:	**Deployment & DevOps**	**8 Hours**
Cloud Deployment (AWS, Netlify, Firebase Hosting), CI/CD Pipelines for Web Development, Docker and Containerization for Web Apps, SEO Optimization and Analytics.		
TOTAL LECTURES		45 **Hours**

Books:

1. J. Duckett, “HTML & CSS: Design and Build Websites”, Wiley, 2011, ISBN-10: 1118008189, ISBN-13: 978-1118008188.
2. E. Freeman and E. Robson, “Head First HTML and CSS”, O’Reilly Media, 2012, ISBN-10: 0596159900, ISBN-13: 978-0596159900.
3. M. Haverbeke, “Eloquent JavaScript: A Modern Introduction to Programming”, No Starch Press, 2018, ISBN-10: 1593279507, ISBN-13: 978-1593279509.
4. A. Rauschmayer, “Speaking JavaScript: An In-Depth Guide for Programmers”, O’Reilly Media, 2014, ISBN-10: 1449365035, ISBN-13: 978-1449365035.
5. B. W. Griffith, “Full-Stack React Projects”, Packt Publishing, 2020, ISBN-10: 1839215410, ISBN-13: 978-1839215414.

6. K. S. Syed, “Node.js Design Patterns”, Packt Publishing, 2020, ISBN-10: 1839214112, ISBN-13: 978-1839214110.
7. M. Nebeling, “Modern Web Development: Understanding Domains, Technologies, and User Experience”, Springer, 2019, ISBN-10: 303017593X, ISBN-13: 978-3030175932.

Career Advancement & Skill Development-VI Aptitude and Soft Skill (TIU-CASD-UTR-S398A)

Program: B. Tech. in CSE-AI	**Year, Semester: 3rd**, 6th
Course Title: Career Advancement & Skill Development-VI Aptitude and Soft Skill	**Subject Code:** TIU-CASD-UTR-S398A
Contact Hours/Week: 2–0–0 (L–T–P)	**Credit:** 2

COURSE OBJECTIVE:

Enable the student to:

5. develop foundational skills in quantitative aptitude, covering algebra, number systems, and data interpretation.
6. enhance problem-solving abilities using mathematical techniques like probability, permutations, and coordinate geometry.
7. strengthen logical and analytical reasoning through puzzles, syllogisms, and non-verbal reasoning.

COURSE OUTCOME:

The student will be able to:

CO-1	Recall fundamental concepts of algebra, number systems, statistics, and arithmetic calculations.	K1
CO-2	Apply quantitative techniques to solve problems related to percentages, profit & loss, time & work, and speed & distance.	K2
CO-3	Analyze complex mathematical problems involving permutations & combinations, probability, logarithms, and coordinate geometry.	K3
CO-4	Evaluate reasoning-based problems such as syllogism, puzzles, blood relations, and coding-decoding for logical conclusions.	K3
CO-5	Solve non-verbal reasoning problems involving patterns, sequences, and spatial reasoning techniques.	K4
CO-6	Interpret data from charts, graphs, and tables to draw meaningful conclusions for decision-making.	K4

COURSE CONTENT:

Module 1:	**Fundamentals of Quantitative Aptitude**	**5 Hours**
Algebra – Basics, Number Systems & Divisibility Rules, HCF & LCM, Numbers and Decimal Fractions, Arithmetic Mean, Geometric Mean, Median, Mode, Standard Deviation & Variance.		
Module 2:	**Arithmetic and Data Interpretation**	**5 Hours**

Percentage, Profit & Loss, Time and Work, Speed, Time & Distance (Including Relative Speed), Ratio and Proportion, Averages, Data Interpretation (Tables, Pie Charts, Graphs).		
Module 3:	**Advanced Quantitative Aptitude**	**5 Hours**
Simple Interest & Compound Interest, Mixtures & Alligation, Pipes & Cisterns, Factor Theorem & Remainder Theorem, Permutation & Combination, Probability, Indices & Surds, Logarithm & Quadratic Equations.		
Module 4:	**Logical Reasoning – Basics**	**4 Hours**
Syllogism, Statement & Conclusions, Statement & Assumptions, Series Completion & Analogy, Alphabet Test, Seating Arrangements, Venn Diagrams.		
Module 5:	**Advanced Logical Reasoning**	**4 Hours**
Direction Sense Test, Blood Relations, Puzzle Test, Classification, Coding-Decoding, Number, Ranking, and Time Sequence Test, Eligibility Test.		
Module 6:	**Non-Verbal & Analytical Reasoning**	**3 Hours**
Alphanumeric Sequence Puzzle, Mathematical Operations, Logical Sequence of Words, Arithmetical Reasoning, Inserting the Missing Character, Non-Verbal Reasoning (Series, Analogy, Classification, Analytical Reasoning, Completion of Patterns), Paper Cutting & Paper Folding, Cubes and Dice, Rule Detection, Grouping of Identical Figures, Dot Situation, Construction of Squares and Triangles.		
Module 7:	**Special Topics & Exam Preparation**	**4 Hours**
Clocks & Calendar, 2D Coordinate Geometry (Straight lines, Locus, Circles, Ellipse, Parabola, Hyperbola), Mensuration (2D & 3D), Geometry & Trigonometry, Graphical Concepts of Equations and Inequations, Function, Variation, and Series (A.P, G.P, H.P), Binomial Theorem Applications (Fermat's, Wilson's, Totient Method), Data Sufficiency Problems (Covering all modules), Mock Tests & Problem-Solving Sessions.		
	TOAL LECTURE	**30 Hours**

Books:

1. Aggarwal, R. S. (2018). Quantitative aptitude for competitive examinations (Revised ed.). S. Chand Publishing.
2. Verma, R. (2018). Fast track objective arithmetic (2nd ed.). Arihant Publications.
3. Aggarwal, R. S. (2019). A modern approach to verbal & non-verbal reasoning (Revised ed.). S. Chand Publishing.
4. Sinha, N. K. (2020). Logical reasoning and data interpretation for the CAT (5th ed.). Pearson Education.
5. Sharma, A. (2021). How to prepare for quantitative aptitude for the CAT (10th ed.). McGraw Hill.