Syllabus: CSAI3100, Soft Computing

School: SSCSE		**Batch : 2025-2029**		
Program: B.Tech		**Current Academic Year: 2025-26**		
Branch: CSE (AIML)		**Semester: V**		
1	**Course Code**	**CSAI3100**	**Course Name: Soft Computing**	
2	**Course Title**	**Soft Computing**		
3	Credits	3		
4	Contact Hours (L-T-P)	3	0	0
	Course Status	Core		
5	Course Objective	The primary objective of this course is to provide an introduction to the basic principles, techniques, and applications of soft computing. ● Upon successful completion of the course, students will have an understanding of the basic areas of Soft Computing including Artificial Neural Networks, Fuzzy Logic and Genetic Algorithms. ● Provide the mathematical background for carrying out the optimization associated with neural network learning. ● Aim of this course is to develop some familiarity with current research problems and research methods in Soft Computing by working on a research or design project.		
6	Course Outcomes	Upon successful completion of the course, students will be able to: **CO1:** Define the basic concepts of soft computing. **CO2:** Explain applications & operations of Fuzzy Logic in real life problems. **CO3:** Apply different FIS models to solve optimization problems. **CO4:** Analyse and examine Evolutionary and swarm algorithms in solving real world Multi-Objective optimization problems **CO5:** Choose of different optimization algorithms to solve real-life multi objective problems. **CO6:** Discuss applications of Soft Computing and solve Problems in Varieties of Application Domains.		
7	Course Description	This course will cover fundamental concepts used in Soft computing. The concepts of Fuzzy logic (FL) will be covered first, followed by Artificial Neural Networks (ANNs) and optimization techniques using Genetic Algorithm (GA). Applications of Soft Computing techniques to solve a number of real life problems will be covered to have hands on practices.		
8				CO Mapping
	Unit 1	**Introduction to Soft Computing**		
	A	Concept of computing systems. What is Soft Computing?		CO1
	B	"Soft" Computing versus "Hard" computing		CO1
	C	Characteristics of Soft computing, Some applications of Soft computing techniques		CO1, CO6
	Unit 2	**FUZZY LOGIC**		
	A	Introduction to Fuzzy logic, Fuzzy sets and membership functions		CO2
	B	Operations on Fuzzy sets. Fuzzy relations, rules, propositions, implications and inferences.		CO2
	C	Defuzzification techniques, Fuzzy logic controller design, Some real life societal applications of Fuzzy logic.		CO2

	Unit 3	**Fuzzy inference System**			
	A	Fuzzy Inference Systems, Different Fuzzy Models: Mamdani Fuzzy Models, Sugeno Fuzzy Models			CO3
	B	Tsukamoto Fuzzy Models, Input Space Partitioning and Fuzzy Modeling.			CO3
	C	Neuro Fuzzy Modelling: Adaptive Neuro-Fuzzy Inference Systems, Architecture, Hybrid Learning Algorithm, Learning Method that Cross- fertilize ANFIS and RBFN			CO3
	Unit 4	**Swarm and Evolutionary Algorithms**			
	A	Concept of "Genetics" and "Evolution" and its application to probabilistic search techniques			CO4
	B	Basic GA framework and different GA architectures, GA operators: Encoding, Crossover, Selection, Mutation, Solving single-objective optimization problems			CO4
	C	Swarm Optimization: Introduction to Ant Colony Optimization, Particle Swarm Optimization etc.			CO4
	Unit 5	**Multi-objective Optimization Problem Solving**			
	A	Concept of multi-objective optimization problems (MOOPs) and issues of solving them.			CO5,CO6
	B	Multi-Objective Evolutionary Algorithm (MOEA) Non-Pareto approaches to solve MOOPs, Pareto-based approaches to solve MOOPs , Some applications with MOEAs			CO5,CO6
	C				CO5,CO6
	Mode of examination	Theory			
	Weightage Distribution	CA	MSE	ESE	
		25%	25%	50%	
	Text book/s*	1. George J. Klir and Bo Yuan, "Fuzzy sets and Fuzzy Logic", Prentice Hall, USA. 2. Goldberg D.E., Genetic Algorithms in Search, Optimization, and Machine Learning Addison Wesley. 3. Timothy J. Ross, "Fuzzy Logic with Engineering Applications", McGraw Hill			
	Other References	1. Jang J.S.R., Sun C.T. and Mizutani E, "Neuro-Fuzzy and Soft computing", Prentice Hall. 2. An Introduction to Genetic Algorithms, Melanie Mitchell, MIT Press, 2000. 3. Genetic Algorithms In Search, Optimization And Machine Learning, David E. Goldberg, Pearson Education, 2002. 4. Practical Genetic Algorithms, Randy L. Haupt and sue Ellen Haupt, John Willey & Sons, 2002			

PO and PSO mapping with level of strength for Course Name SOFT COMPUTING **(Course Code** CSAI-3100**)**

	PO 1	PO 2	PO 3	PO4	PO 5	PO 6	PO 7	PO8	PO 9	PO10	PO1 1	PSO 1	PSO 2	PSO 3
	Engineering Knowledge	Problem Analysis	Design/ Development of Solutions	Conduct Investigations of Complex Problems	Engineering Tool Usage	The Engineer and The World	Ethics	Individual and Collaborative Teamwork	Communication	Project Management and Finance	Life-Long Learning	Familiarity and practical proficiency	Understand, analyze and develop	Apply standard Software
CO 1	3	2	1	1	1	1	-	-	1	1	2	1	3	1
CO 2	3	2	2	3	2	3	-	-	2	3	2	3	3	3
CO 3	3	3	3	3	3	3	-	-	2	3	2	3	3	3
CO 4	3	3	3	3	3	3	-	-	2	3	2	3	3	3
CO 5	3	3	3	3	3	3	2	2	2	3	2	3	3	3
CO 6	3	3	3	3	3	3	2	2	2	3	2	3	3	3

Average of non-zeros entry in following table (should be auto calculated).

Course Code/ Name	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO1 0	PO1 1	PSO 1	PSO 2	PSO 3
	Engineering Knowledge	Problem Analysis	Design/ Development of Solutions	Conduct Investigations of Complex Problems	Engineering Tool Usage	The Engineer and The World	Ethics	Individual and Collaborative Teamwork	Communication	Project Management and Finance	Life-Long Learning	Familiarity and practical proficiency	Understand, analyze and develop	Apply standard Software
CSAI3100 / Soft Computing	3.00	2.67	2.50	2.67	2.50	2.67	2.00	2.00	1.83	2.67	2.00	2.67	3.00	2.67

1. Addressed to **Slight (Low=1) extent** **2.** Addressed to **Moderate (Medium=2) extent**
3. Addressed to **Substantial (High=3) extent**

Syllabus: CSE356, Design and Analysis of Algorithms

School: SET			
Program:B.Tech			
Branch:CSE		**Semester:V**	
1	Course Code	CSE356 | Course Name: Design and Analysis of Algorithms	
2	Course Title	Design and Analysis of Algorithms	
3	Credits	3	
4	Contact Hours (L-T-P)	3-0-0	
	Course Status	UG	
5	Course Objective	Objective of this course is to 1. Reinforce basic design concepts (e.g., pseudocode, specifications, top-down design) 2. Knowledge of algorithm design strategies 3. Familiarity with an assortment of important algorithms. 4. Enable students to analyze time and space complexity	
6	Course Outcomes	Students will be able to: **CO1: Analyze** the asymptotic performance of algorithms **CO2**: **Describe** the dynamic-programming and Greedy paradigm and explain when an algorithmic design situation calls for it. **CO3: Demonstrate** a familiarity with major algorithms and data structures **CO4: Apply** important algorithmic design paradigms and methods of analysis **CO5: Discuss** NP-complete problems and develop algorithms to solve the problems. **CO6: Choose** appropriate algorithm design techniques for solving problems.	
7	Course Description	This course introduces concepts related to the design and analysis of algorithms. Specifically, it discusses recurrence relations, and illustrates their role in asymptotic and probabilistic analysis of algorithms. It covers in detail greedy strategies divide and conquer techniques, dynamic programming and max flow - min cut theory for designing algorithms, and illustrates them using a number of well-known problems and applications.	
8	Outline syllabus		CO Mapping
	Unit 1	**Introduction**	
	A	Introduction : Algorithms, Analyzing algorithms, Complexity of algorithms, Growth of functions, Performance measurements	CO1, CO3
	B	Asymptotic Notations and their properties – Mathematical analysis for Recursive and Non-recursive algorithms, Recurrences relations, Master Method	CO1, CO2, CO3

	C	Divide-and-conquer: Analysis and Structure of divide-and-conquer algorithms, Divide-and-conquer examples-Quick sort, Merge sort, Sorting in Linear Time counting sort, Heap Sort			CO1, CO2, CO4
	Unit 2	**Greedy Method**			
	A	Overview of the Greedy paradigm, Analysis and example: task scheduling,			CO1,CO2,CO3
	B	Fractional Knapsack problem, Single source shortest paths problem: Dijkastra's Algorithm, Bellman-ford Algorithm,			CO1,CO2,CO3
	C	Overview and analysis of Backtracking & Branch and Bound: N-Queens problem and Sum of subsets			CO1,CO2,CO3
	Unit 3	**Dynamic Programming**			
	A	Overview, Difference between dynamic programming and divide and conquer, All pair shortest path problems: Floyd-Warshall Algorithm			CO1,CO2,CO3, CO4
	B	Applications and analysis: Matrix Chain Multiplication,			CO1, CO2, CO3, CO4
	C	Applications and analysis: Longest Common sub-sequence, 0/1 Knapsack Problem			CO1,CO2,CO3
	Unit 4	**Advanced Data Structures**			
	A	Red-Black Trees - Definition, Applications, Insertion and deletion of elements in RB-Tree			CO1, CO2, CO3, CO4
	B	B-Trees - Definitions, Applications, Insertion and Deletion in B-Trees			CO1, CO2, CO4
	C	Data Structure for Disjoint Sets – Definition, Binomial Heaps, Fibonacci Heaps.			CO1, CO2, CO3, CO4
	Unit 5	**Selected Topics**			
	A	Introduction to NP Complete and NP Hard Problems, Examples, Amortized Analysis			CO1,CO2,CO3,
	B	Approximation Algorithms – Travelling Sales Person Problem and Vertex Cover Problem, Randomized Algorithms, Randomized Quick Sort Algorithm			CO1,CO2,CO3
	C	String Matching Algorithms – Naive String Matching Algorithm, Rabin Karp Algorithm.			CO1,CO2,CO3, CO4
	Mode of examination	Theory			
	Weightage Distribution	CA	MTE	ETE	
		25%	25%	50%	
	Text book/s*	1. Cormen et al., "Introduction of Computer Algorithms", Prentice Hall India			

	Other References	1. Sahni et al., "Fundamentals of Computer Algorithms", Galgotia Publications. 2. Hopcroft A, The Design And Analysis Computer Algorithms, Addison Wesley	

CO and PO Mapping

S. No.	Course Outcome	Program Outcomes (PO) & Program Specific Outcomes (PSO)
1.	**CO1: Analyze** the asymptotic performance of algorithms	PO1, PO2, PO3, PO4, PO9, PSO1, PSO2, PSO3
2.	**CO2**: **Describe** the dynamic-programming and Greedy paradigm and explain when an algorithmic design situation calls for it.	PO1, PO2, PO3, PO4, PO9, PSO1, PSO2, PSO3
3.	**CO3: Demonstrate** a familiarity with major algorithms and data structures	PO1, PO2, PO3, PO9, PSO1, PSO2
4.	**CO4: Apply** important algorithmic design paradigms and methods of analysis	PO1, PO2, PO3, PO4, PO9, PSO1, PSO2, PSO3
5.	**CO5: Discuss** NP-complete problems and develop algorithms to solve the problems.	PO1, PO2, PO3, PO4, PO9, PSO1, PSO2, PSO3
6.	**CO6: Choose** appropriate algorithm design techniques for solving problems.	PO1, PO2, PO3, PO4, PO5, PO9, PSO1, PSO2

PO and PSO mapping with level of strength for Course Name Design and Analysis of Algorithms Course Code CSE 354)

Cos	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO1 0	PO1 1	PO1 2	PSO 1	PSO 2	PSO 3
CO 1	2	3	1	2	-	--	--	-	2	-	-	-	3	2	2
CO 2	2	2	2	2	-	--	--	-	3	-	-	-	2	3	2
CO 3	2	1	2	-	-	--	--	-	1	-	-	-	3	2	-
CO 4	1	2	2	3	-	--	--	-	2	-	-	-	2	2	2
CO 5	3	3	1	3	-	-	-	-	3	-	-	-	2	1	3
CO 6	2	2	3	2	2	-	-	--	2	-	-	-	3	2	-

Average of non-zeros entry in following table (should be auto calculated).

Course Code	Course Name	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO1	PSO2	PSO3
CSP 354	**Design and Analysis of Algorithms Lab**	2	2.17	1.83	2.4	2	-	-	-	2.2	-	-	-	2.5	2	2.3

Strength of Correlation

1. Addressed to ***Slight (Low=1) extent*** ***2.*** Addressed to ***Moderate (Medium=2) extent***
3. Addressed to ***Substantial (High=3) extent***

Syllabus: Design and Analysis of Algorithm lab

School:		**Batch:**	
Program:		**Current Academic Year:**	
Branch:		**Semester:**	
1	Course Code	CSP 350	
2	Course Title	**Design and Analysis of Algorithm lab**	
3	Credits	1	
4	Contact Hours (L-T-P)	0-0-2	
	Course Status	Compulsory/Elective	
5	Course Objective	Objective of this course is to 1. Reinforce basic design concepts (e.g., pseudocode, specifications, top-down design) 2. Knowledge of algorithm design strategies 3. Familiarity with an assortment of important algorithms. ● Enable students to analyze time and space complexity	
6	Course Outcomes (same as theory course)	Students will be able to: **CO1: calculate** time complexity of searching algorithm **CO2**: **Write** program based on dynamic programming. **CO3: apply** greedy algorithm to any problem **CO4: develop** program based on advanced data structure CO5: **design** a program based on different string matching algorithm CO6**: implement** real world problem based on greedy and dynamic algorithm	
7	Course Description	This course introduces concepts related to the design and analysis of algorithms. Specifically, it discusses recurrence relations, and illustrates their role in asymptotic and probabilistic analysis of algorithms. It covers in detail greedy strategies divide and conquer techniques, dynamic programming and max flow - min cut theory for designing algorithms, and illustrates them using a number of well-known problems and applications.	
8	Outline syllabus		CO Mapping
	Unit 1	**Practical based on Searching and sorting**	
		1. WAP to demonstrate the concept of Linear and Binary Search 2. WAP to implement Merge sort 3. WAP to implement Quick Sort	CO1
	Unit 2	**Practical based on Dynamic Programming**	
		1. WAP to implement Matrix Chain Multiplication problem 2. WAP to demonstrate the concept of Longest Common Subsequence(LCS) 3. WAP to demonstrate concept of 0 – 1 Knapsack Problem	CO2, CO6

	Unit 3	**Practical based on Greedy Programming**			
		1. WAP to demonstrate concept of Minimum Spanning Tree(Prim's Algorithm) 2. WAP to demonstrate concept of Fractional Knapsack Problem 3. WAP to implement single source shortest problem using Dijkstra's Algorithm			CO3, CO6
	Unit 4	**Practical based on Advance concepts**			
		WAP to demonstrate concept of Red Black Tree insertion and Deletion			CO4
	Unit 5	**Practical based on String Matching**			
		1. WAP to demonstrate the concept of Naïve String matching algorithm. 2. WAP to demonstrate the concept of Robin Karp Algorithm.			CO5
	Mode of examination	Jury/Practical/Viva			
	Weightage Distribution	CA	CE(Viva)	ETE	
		25%	25%	50%	
	Text book/s*	-			
	Other References				

PO and PSO mapping with level of strength for Course Name Design and Analysis of Algorithms Lab. Course Code CSP 350)

Cos	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2	PSO 3
CO 1	3	3	2	3	1	--	--	-	2	-	-	-	2	3	3
CO 2	2	3	3	2	2	--	--	-	2	-	-	-	3	2	2
CO 3	3	2	2	-	3	--	--	-	1	-	-	-	2	1	-
CO 4	2	3	3	3	1	--	--	-	3	-	-	-	3	3	1
CO 5	3	2	2	3	2	-	-	-	2	-	-	-	2	3	2
CO 6	2	3	3	1	3	-	-	--	1	-	-	-	3	2	3

Average of non-zeros entry in following table (should be auto calculated).

Course Code	Course Name	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO1	PSO2	PSO3
CSP 350	**Design and Analysis of Algorithms Lab**	2.5	2.7	2.5	2.4	2	-	-	-	1.8	-	-	-	2.5	2.3	2.2

Strength of Correlation

1. Addressed to ***Slight (Low=1) extent*** ***2.*** Addressed to ***Moderate (Medium=2) extent***
3. Addressed to ***Substantial (High=3) extent***

School:		**School of Engineering and technology**	
Department		**Department of Computer Science and Engineering**	
Program:		**B.Tech**	
Branch:		Computer Science and Engineering	
1	Course Code	CSE357	Course Name: Software Engineering and Testing Methodologies
2	Course Title	Software Engineering and Testing Methodologies	
3	Credits	3	
4	Contact Hours (L-T-P)	2-0-1	
	Course Status	Core /Elective/Open Elective: Core	
5	Course Objective	The objective of this course is to provide 1. Fundamental knowledge of software engineering 2. To make student aware of best software engineering practices 3. Inculcate ability in students to work as an effective member or leader of software engineering teams 4. To help students to develop skills that will enable them to construct software of high quality	
6	Course Outcomes	CO1: Illustrate and compare an effective software engineering process, based on knowledge of widely used development lifecycle model CO2: Apply effective requirement elicitation techniques to develop SRS for a project. CO3: Construct design documents with the help of designing tools CO4: Analyze testing strategies for a software system CO5: Develop and deliver quality software as an individual or as part of a multidisciplinary team. CO6: Adapt techniques and tools necessary for software engineering practices.	
7	Course Description	This course provides knowledge of software engineering. It introduces concepts such as software processes and agile methods and essential software development activities, from initial specification to system maintenance. Formalisms and tools to assist in software development are also presented, including common design patterns and UML notation. Course focuses on all levels of testing.	
8	Outline syllabus		CO Mapping
	Unit 1	**Introduction to Software Engineering and Process Models**	
	A	Significance challenges and Software Myths in software engineering, Software Characteristics	CO1
	B	Software Development Methodologies: Waterfall model, prototyping model, Incremental model, Spiral model, V model	CO1
	C	Agility, Agile Process models: Extreme Programming (XP), Adaptive Software Development (ASD), Scrum	CO1
	Unit 2	**Software Requirement Engineering**	
	A	Requirement Engineering process, Elicitation techniques	CO2
	B	Types of Requirements , Feasibility study	CO2
	C	Requirement Documentation: Characteristics of SRS,	CO2

		Document SRS according to IEEE standards			
	Unit 3	**Software Design**			
	A	Design Concepts, Design Strategies: Function Oriented Design, Object Oriented Design, Top-Down and Bottom-Up Design			CO3, CO6
	B	Effective modular design: Cohesion, Coupling			CO3, CO6
	C	UML Diagrams and Tools: Introduction to UML Diagrams, Use Case, Object and Class, Interaction diagram: Sequence			CO3, CO6
	Unit 4	**Software Implementation and Testing**			
	A	Fundamental of testing: Objectives, principles, myths and facts, limitations of testing			CO4
	B	Levels of testing, Acceptance Testing and its types, Integration techniques			CO4, CO6
	C	White Box Testing, Black Box Testing, Verification and Validation, Test case designing, Debugging			CO4, CO6
	Unit 5	**Maintenance & Quality Management**			
	A	Introduction to Maintenance, Need for Maintenance, Categories of Maintenance, Cost of Maintenance			CO5, CO6
	B	Quality Concepts: Quality, Quality Control, Cost of Quality, Software Quality Assurance, SQA Plan			CO5, CO6
	C	Statistical Software Quality Assurance: Six Sigma, The ISO 9000 Quality Standards, Capability Maturity Model			CO5, CO6
	Mode of examination	Theory/Jury/Practical/Viva			
	Weightage Distribution	CA	MTE	ETE	
		25%	25%	50%	
	Text book/s*	1. Pressman R S, Software Engineering: A Practitioners Approach, McGraw Hill.			
	Other References	1.Datta S, Software Engineering: Concepts and Applications, Oxford University Press, 2010. 2. K.K. Aggrawal and Yogesh Singh, "Software Engineering", New Age International Publication 3 .Sommerville, Ian. "Software Engineering", Pearson(Latest Ed).			

CO and PO Mapping

S. No.	Course Outcome	Program Outcomes (PO) & Program Specific Outcomes (PSO)
1.	CO1: Illustrate and compare an effective software engineering process, based on knowledge of widely used development lifecycle model	PO1,PO3,PO8,PO9,PO10,PO12,PSO1,PSO3
2.	CO2: Apply effective requirement elicitation techniques to develop SRS for a project	PO1,PO2,PO3,PO4,PO5,PO8,PO9,PO10, PO11,PO12,PSO1,PSO3
3.	CO3: Construct design documents	PO1,PO2,PO3,PO4,PO5, PO8,PO9,PO10,

	with the help of designing tools	PO11,PO12,PSO1,PSO3
4.	CO4:Analyze testing strategies for a software system	PO1,PO2,PO4,PO5,PO6,PO7,PO8,PO9,PO10, PO11,PO12,PSO1,PSO3
5.	CO5: Develop and deliver quality software as an individual or as part of a multidisciplinary team.	PO1,PO2,PO3,PO4,PO5, PO6,PO7, PO8,PO9,PO10,PO11,PO12,PSO1,PSO3
6.	CO6: Adapt techniques and tools necessary for software engineering practices .	PO1,PO4,PO5,PO8,PO9,PO10,PO11,PSO3

PO and PSO mapping with level of strength for Course Name Software Engineering and Testing Methodologies (Course Code CSE357)

Course Code_ Course Name	CO's	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2	PS O3
CSE357_ Software Engineering and Testing Methodologies	CO1	3	-	2	-	-	-	-	1	2	3	-	3	1	-	2
	CO2	3	3	2	3	3	-	-	1	2	3	2	3	2	-	3
	CO3	3	2	3	3	3	-	-	1	2	3	1	2	2	-	3
	CO4	3	1	-	1	3	2	2	2	3	3	2	3	1	-	3
	CO5	3	1	3	3	3	3	3	2	3	3	1	3	1	-	3
	CO6	2	-	-	1	3	-	-	1	2	2	2	-	-	-	3

Average of non-zeros entry in following table (should be auto calculated).

Course Code	Course Name	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 11	PSO 2	PSO 3
CSE357	Software Engineering and Testing Methodologies	2.8	1.75	2.5	2.2	3	2.5	2.5	1.3	2.3	2.8	1.6	2.8	1.4	0	2.8

Strength of Correlation

1. Addressed to ***Slight (Low=1) extent*** ***2.*** Addressed to ***Moderate (Medium=2) extent***
3. Addressed to ***Substantial (High=3) extent***

SHARDA SCHOOL OF ENGINEERING & TECHNOLOGY
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Session	2024-2025
Term	Term I
Course	B.TECH
Program	CSE
Name of Lab Course	SOFTWARE ENGINEERING &TESTING METHDODOLOGIES
Course Code	CSP357

L	T	P
0	0	2

LIST OF EXPERIMENTS

S. No.	Name or Objective of Experiment	CO#
1	Design requirement analysis and develop Software Requirement Specification Sheet (SRS) for suggested system in IEEE standard.	CO1
2	Draw the user's view analysis for the suggested system: Use case diagram, Activity Diagram and State chart Diagram.	CO1
3	Draw the structural view diagram for the system: Class diagram, object diagram and Sequence diagram.	CO2
4	Perform Unit testing of calculator program using Junit testing framework.	CO2
5	Perform Integration testing of calculator program using Junit testing framework.	CO3
6	Design Test Case for Inventory Management System based on System Specification.	CO3
7	Design Test cases of login module of FlipKart (as an example).	CO3
8	Implement QA validation for Amazon online purchase.	CO3
9	Implement QA validation for Gmail Login feature.	CO3
10	Implement Test Cases of website using open-source test automation platform (TestSigma).	CO4

Added List of Experiments

S. No.	Name or Objective of Experiment	CO#
A	Study on Project tracking tool **JIRA** for Quality assurance	CO5
B	Study on widely used software test automation tool **Selenium**.	CO5

Prof(Dr.) Shajee Mohan		Ms. Priya Sharma

Syllabus: CSCR3203, Intro to LLMs and Gen AI

School: SSCSE		**Batch: 2025-2029**	
Program: B. Tech		**Current Academic Year: 2025-26**	
Branch: CSE		**Semester: V**	
1	Course Code	CSCR3203	Course Name: Intro to LLMs and Gen AI
2	Course Title	Intro to LLMs and Gen AI	
3	Credits	2	
4	Contact Hours (L-T-P)	2-0-2	
	Course Status	Core	
5	Course Objective	1. Understand the evolution, architecture, and working principles of LLMs and generative models. 2. Explore transformer-based architectures and their role in sequence modeling and natural language understanding. 3. Apply LLMs to generate text, code, and images using industry-standard tools. 4. Analyze ethical and societal implications of deploying generative models. 5. Develop and evaluate domain-specific applications of generative AI.	
6	Course Outcomes	After the completion of this course students will be able to: CO1: Describe the architecture, evolution, and key components of LLMs and Generative AI. CO2: Illustrate the working of transformer-based models like BERT, GPT, and T5. CO3: Implement basic applications using pretrained LLMs and prompt engineering. CO4: Analyze the capabilities, limitations, and biases in LLM-generated outputs. CO5: Evaluate ethical concerns and safety guidelines in generative model deployment. CO6: Design and present a mini-project using LLMs for a real-world application.	

7	Course Description	This course introduces the foundational concepts and recent advances in Large Language Models (LLMs) and Generative AI, emphasizing transformer architectures, prompt engineering, ethical implications, and real-world applications in creative and cognitive domains. The course blends theoretical understanding with hands-on experiments using modern frameworks such as OpenAI, Hugging Face, and LangChain.	
8	Outline syllabus		CO Mapping
	Unit 1	**Foundations of Generative AI and Language Mode**	
	A	Introduction to Generative AI: Definition, key concepts, comparison with traditional AI; Use-cases (text, image, music, code).	CO1
	B	Language Modeling Fundamentals: Statistical vs Neural Language Models; N-grams; Perplexity; Training objectives (causal vs masked).	CO1
	C	Emergence of LLMs: History and evolution: From Word2Vec to GPT; Transfer learning in NLP; Pretraining vs fine-tuning.	CO1
	Unit 2	**Transformer Architectures and LLMs**	
	A	Attention Mechanism: Scaled dot-product attention, self-attention, positional encodings.	CO2
	B	Transformer Architecture: Encoder-decoder framework, comparison of models: BERT, GPT, T5, LLaMA.	CO2
	C	Parameter Scaling and Capabilities: Model scaling laws, in-context learning, zero-shot vs few-shot capabilities.	CO2
	Unit 3	**Prompt Engineering and Toolkits**	
	A	Prompt Engineering Basics: Prompt types: zero-shot, one-shot, few-shot; Prompt formatting; Instruction tuning.	CO3
	B	Using APIs and Libraries: Hands-on with OpenAI API, Hugging Face Transformers, Google PaLM; Input-output control.	CO3
	C	LangChain and Workflow Integration: Agents, Chains, Memory in LangChain; Connecting LLMs to tools and data.	CO3
	Unit 4	**Evaluation, Bias, and Ethics**	
	A	Output Evaluation Metrics: BLEU, ROUGE, perplexity, truthfulness, helpfulness, human vs automated evaluation.	CO4, CO5

	B	Bias and Hallucination: Common failure cases: hallucination, toxicity, social bias; Red-teaming models.			CO4, CO5
	C	Ethical & Regulatory Frameworks: Responsible AI principles; Guidelines by OpenAI, EU AI Act, AI ethics frameworks.			CO4, CO5
	Unit 5	**Capstone Project and Applications**			
	A	Domain-Specific Applications: Chatbots, educational tutors, creative writing, legal assistants, code completion.			CO6
	B	End-to-End LLM Project Design: Problem formulation, data pipelines, prompt & API design, interface integration.			CO6
	C	Project Presentation and Review: Final presentations, peer reviews, ethical assessment, future scope discussion.			CO6
	Mode of examination	Monitoring and Continuous Evaluation			
	Weightage Distribution	CA	CE (Viva)	ESE	
		25%	25%	50%	
	Textbook/s*	1. 'You Look Like a Thing and I Love You: How AI Works and Why It's Making the World a Weirder Place', Janelle Shane, Publisher: Voracious (Little, Brown), 2019			
	Other References	2. Deep Learning with Python, François Chollet, Manning Publications, Edition: 2nd Edition (2021) 3. Transformers for Natural Language Processing: Build and Train State-of-the-Art Natural Language Processing Models Using the Transformer Architecture, Denis Rothman, Packt Publishing, 2nd Edition (2022).			

PO and PSO mapping with level of strength for Course: CSCR3203, Intro to LLMs and Gen AI

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
	Engineering Knowledge	Problem Analysis	Design/Development of Solutions	Conduct Investigations of Complex Problems	Engineering Tool Usage	The Engineer and The World	Ethics	Individual and Collaborative Team work	Communication	Project Management and Finance	Life-Long Learning	Familiarity and practical proficiency	Understand, analyze and develop	Apply standard Software
CO1	3	2	-	-	2	-	1	1	-	1	-	2	-	-
CO2	3	2	-	-	2	-	-	-	-	1	-	2	-	-
CO3	2	2	3	2	3	-	-	-	2	1	2	2	3	-
CO4	2	3	-	2	2	2	2	2	-	2	-	3	-	2
CO5	2	-	-	2	1	3	3	3	2	2	-	2	-	3
CO6	2	2	3	2	3	-	-	1	3	3	3	2	3	3

Average of non-zeros entry in following table (should be auto calculated).

Course Code/ Name	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
	Engineering Knowledge	Problem Analysis	Design/Development of Solutions	Conduct Investigations of Complex Problems	Engineering Tool Usage	The Engineer and The World	Ethics	Individual and Collaborative Team work	Communication	Project Management and Finance	Life-Long Learning	Familiarity and practical proficiency	Understand, analyze and develop	Apply standard Software
CSCR3203, Intro to LLMs and Gen AI	2.33	2.20	3.00	2.00	2.17	2.50	2.00	1.75	2.33	1.67	2.50	2.17	3.00	2.67

Strength of Correlation: 32.28

1. Addressed to ***Slight (Low=1)*** ***extent***

2. Addressed to ***Moderate (Medium=2)*** ***extent***

3. Addressed to ***Substantial (High=3)*** ***extent***

SHARDA UNIVERSITY

Greater Noida, Uttar Pradesh, India

SCHOOL OF ENGINEERING & TECHNOLOGY

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

School: SET		**Batch : 2021-2025**	
Program: B.Tech		**Current Academic Year: 2025-2026**	
Branch: CSE		**Semester: V**	
1	**Course Code**	**CSA-022**	**Course Name: Introduction to Cloud Computing with ML**
2	**Course Title**	**Introduction to Cloud Computing with ML**	
3	**Credits**	**3**	
4	**Contact Hours (L-T-P)**	**3-0-0**	
	Course Status	Specialization Elective	
5	**Course Objective**	This introductory course on Cloud computing will teach both the fundamental concepts of how and why Cloud systems works, as well as Cloud technologies that manifest these concepts.	
6	**Course Outcomes (CO's)**	At the end of the course, students will have achieved the following learning objectives. CO1. Define the basics of cloud and recall the computer Science concepts which are helpful in understanding on demand service architecture. CO2. Classify and describe the architecture and taxonomy of parallel and distributed computing, including shared and distributed memory, and data and task parallel computing. CO3. Apply the PAAS and SAAS to manage the workflow and use of cloud in scientific application. CO4. Categorize and Characterize between Infrastructure services, deployment models, and governance in cloud computing. Examine the design of task and data parallel distributed algorithms for Clouds and use them to construct Cloud applications. CO5. Evaluate the importance of cloud using monitoring and management of services for performance improvement of HPC and to follow the	

		Governance and Compliances. CO6. Elaborate the design concept and formulate to build the solution using cloud service providers as AWS as EC2, LAMBDA, S3 and Machine Learning Service as AWS SageMaker.	
7	**Course Description**	This course is an introductory course for cloud computing concepts and helps in understanding the core functionalities, algorithms, models and workflows in cloud environment. In this course Students will get demonstrations of real-time cloud services for better exposure and research understanding.	
8	**Syllabus Outline**		**CO Mapping**
	Unit 1	**FOUNDATIONS**	
	A	**Introduction to compute** Types of Computing, Grid computing, distributed computing, Client-server computing, Three Tier Architecture, use of Sockets and Remote Procedure Call, working of RMI and CORBA, Web services, Web Sockets, Message Queues and Message Brokers.	CO1
	B	**Introduction to Cloud Computing** Cloud Computing definition, Roots of Cloud Computing, Layers and Types of Clouds, Desired Features of a Cloud, Cloud Infrastructure Management, Infrastructure as a Service Providers, Platform as a Service Providers, Challenges and Risks	CO1
	C	**Migrating and Integrating into Cloud** Broad Approaches to Migrating into the Cloud, The Seven-Step Model of Migration into a Cloud, Enriching the 'Integration as a Service' Paradigm for the Cloud Era, Evolution and Challenges of SaaS Paradigm, Integration Scenarios, The Integration Methodologies	CO1

	Unit 2	**ENTERPRISE CLOUD COMPUTING AND IAAS**	
	A	**The Enterprise Cloud Computing Paradigm** Issues for Enterprise Applications on the Cloud, Transition Challenges, Enterprise Cloud Technology and Market Evolution, Business Drivers Toward a Marketplace for Enterprise Cloud Computing, The Cloud Supply Chain	CO1,CO2
	B	**Virtual Machines Provisioning and Migration Services** Introduction to Virtual Machines,The Anatomy of Cloud Infrastructures, VM Provisioning and Manageability, Virtual Machine Migration Services, Management of Virtual Machines for Cloud Infrastructures,, Distributed Management of Virtual Infrastructures, Scheduling Techniques	CO1,CO2
	C	**Enhancing Cloud Computing Environments Using a Cluster as a Service** Introduction and Related Work, RVWS Design, Cluster as a Service: The Logical Design, Secure Distributed Data Storage in Cloud Computing, Cloud Storage, Technologies for Data Security in Cloud Computing	CO1,CO2
	Unit 3	**PLATFORM AND SOFTWARE AS A SERVICE**	
	A	**Aneka and CometCloud** Aneka—Integration of Private and Public Clouds, Technologies and Tools for Cloud Computing, Aneka Cloud Platform, CometCloud: An Autonomic Cloud Engine, Introduction of CometCloud (Architecture, Autonomic Behavior, Applications overview)	CO1,CO3
	B	**Business Solutions and WorkFlow** Cloud-Based Solutions for Business Applications (Introduction of Enterprises Demand and Cloud Computing, Dynamic ICT Services), Workflow Engine for Clouds, Workflow Management Systems, Architecture of Workflow Management Systems	CO1,CO3,CO6

	C	**Scientific Applications and MapReduce Model** Scientific Application for Cloud Environments, Classification of Scientific Applications and Services in the Cloud, SAGA-based Scientific Applications, MapReduce Programming Model, MapReduce Impacts and Research Directions	CO1,CO3,CO6
	Unit 4	**MONITORING, MANAGEMENT & GOVERNANCE**	
	A	**SLA Management in Cloud Computing** Introduction of typical Use Cases, Model for Federated Cloud Computing, Security Considerations, SLA Management in Cloud Computing: A Service Provider's Perspective, Types of SLA, Life Cycle of SLA, Automated Policy-based Management	CO1,CO4
	B	**Performance Predictions for HPC on Clouds** Introduction and Background of Grid and Cloud, HPC in the Cloud: Performance-related Issues, Game Hosting on Cloud Resources, Building Content Delivery Networks Using Clouds, Resource Cloud Mashups	CO1,CO4
	C	**Security and Governance** Basic Concept of Organizational Readiness, Drivers for Changes: Common Change Management Models, Security and Risk in the Cloud, Cloud Computing and Identity, Content Level Security—Pros and Cons, Legal Issues in Cloud Computing(PCI DSS), Data Privacy and Security Issues	CO1,CO4
	Unit 5	**AWS with Machine Learning**	

	A	AWS Services:EC2, IAM, S3, Lambda, Introduction to Amazon SageMaker, Machine Learning with Amazon SageMaker, Explore, Analyze, and Process Data, Train a Model with Amazon SageMaker, Deploy a Model in Amazon SageMaker, Set Up Amazon SageMaker, Amazon SageMaker Notebook Instance			CO1,CO5,CO6
	B	Amazon SageMaker Studio, Perform Common Tasks in Amazon SageMaker Studio, Amazon SageMaker API reference, Actions and Data Types, Use Autopilot to automate model development and Problem types, Create and Manage Workforces , Use Ground Truth for Labeling			CO1,CO5,Co6
	C	Process Data and Evaluate Models, Build Models and Choose an Algorithm, Train Models, Debugger, Perform Automatic Model Tuning, Tune Multiple Algorithms, Use Reinforcement Learning, Incremental Training, Deploy Models, Multi-Model Endpoints,			CO1,CO5,CO6
	Mode of examination	Theory			
	Weightage Distribution	**CA**	**MTE**	**ETE**	
		25%	25%	50%	
	Text Books	1. CLOUD COMPUTING Principles and Paradigms, Edited by Rajkumar Buyya, Jam 2. Cloud Computing: A Practical Approach, Anthony T. Velte, Toby J. Velte, Robert Elsenpeter			
	Reference Books	Amazon SageMaker, Developer Guide, https://docs.aws.amazon.com/sagemaker/latest/dg/sagemaker-dg.pdf#gs			
	Online Materials	https://aws.amazon.com/getting-started/hands-on/build-train-deploy-machine-learning-model-sagemaker/ https://aws.amazon.com/machine-learning/			