Academic Regulations, Curriculum and Syllabi 2023

B. Tech. CSE Specialization
Artificial Intelligence & Machine Learning

(Duration of Study: 4 years)

GMR Institute of Technology
Rajam, Andhra Pradesh
(An Autonomous Institute Affiliated to JNTU-GV Vizianagaram, AP)
NBA Accredited and NAAC Accredited

Academic Rules and Regulations

2023

Undergraduate Programs

GMR Institute of Technology
Rajam 532 127, AndhraPradesh
(An Autonomous Institute, Affiliated to JNTU-Gv, Vizianagaram, AP)
Accredited by NAAC & NBA

The Vision of GMRIT

- To be among the most preferred institutions for engineering and technological education in the country
- An institution that will bring out the best from its students, faculty and staff – to learn, to achieve, to compete and to grow – among the very best
- An institution where ethics, excellence and excitement will be the work religion, while research, innovation and impact, the work culture

The Mission of GMRIT

- To turnout disciplined and competent engineers with sound work and life ethics
- To implement outcome based education in an IT-enabled environment
- To encourage all-round rigor and instill a spirit of enquiry and critical thinking among students, faculty and staff
- To develop teaching, research and consulting environment in collaboration with industry and other institutions

Department Vision

To be a preferred department of learning for students and teachers alike, with a commitment towards Academics & Research, serving the students in an atmosphere of innovation, critical thinking and making them Industry ready.

Department Mission

M1. To provide adaptable education in a collaborative and innovative environment in skilling the graduates to solve real world problems in the field of Computer Science and Engineering.

M2. To prepare the students as critical thinking professionals with multidisciplinary research orientation and Innovation.

M3. To instil ethical values and nurture the graduates who will be able to contribute to society.

Program Educational Objectives

PEO1: Apply logical and analytical skills in solving complex real-world engineering problems in the areas of computer Science and allied fields

PEO2: Adaptable to emerging technologies with enhanced professional skills and ability towards continuous learning, facilitating higher studies and research

PEO3: Demonstrate professional ethics, leadership qualities and promote inclusive and collaborative growth with human values towards societal interest

Program Outcomes

Engineering graduate will be able to

PO 1: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems. (Engineering knowledge)

PO 2: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences. (Problem analysis)

PO 3: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations. (Design/development ofsolutions)

PO 4: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions. (Conduct investigations of complex problems)

PO 5: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations. (Modern tool usage)

PO 6: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice. (The engineer and society)

PO 7: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development. (Environment and sustainability)

PO 8: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice. (Ethics)

PO 9: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings. (Individual and team work)

PO 10: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions. (Communication)

PO 11: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments. (Project management and finance)

PO 12: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change. (Life-long learning)

PSO1: Implement principles of computer science focusing on the concepts of artificial intelligence and machine learning in developing solutions to the real world problems. (Program Specific)

PSO2: Facilitate data driven decision making using machine learning algorithms for process analyzing and visualization in various domains. (Program Specific)

Department of CSE-AI&ML

Minimum Credits to be earned: 160 (for Regular Students)
120 for Lateral entry students

S.No	Course Code	Course Name	POs	L	T	P	C
First Semester							
1	23PYX01 23CYX01	Engineering Physics/Chemistry	3/3	3	0	0	3/3
2	23MAX01 23MAX02	Linear Algebra& Calculus/Differential Equations and Vector calculus	3/3	3	0	0	3/3
3	23BEX01/ 23BEX02	Basic Electrical and Electronics Engineering/ Basic Civil & Mechanical Engineering	3/3	3	0	0	3/3
4	23BEX03	Introduction to Programming	3	3	0	0	3
5	23BEX04/ 23HSX01	Engineering Graphics/Communicative English	2/2	2	0	2/0	3/2
6	23PYX02/ 23CYX03/	Engineering Physics Lab/Chemistry Lab	4	2	0	2/2	1/1
7	23BEX05/ 23BEX06	Electrical & Electronics Engineering workshop/Engineering Workshop	1,9,10/1,5,10	2	0	3/3	1.5/1.5
8	23BEX07	Computer Programming Lab	4	2	0	3	1.5
9	23HSX11	-/ECA (Yoga / Sports)	-	-	0	-/1	-/0.5
10	23HSX12	-/CCA (NSS/NCC/Community Service)	-	-	0	-/1	-/0.5
11	23BEX08	IT Workshop/-		0	0	2/-	1/-
12	23HSX02	- /Communicative English Lab		0	0	-/2	-/1
			Total	14/14	00	12/12	20/20
Second Semester							
1	23HSX01/ 23BEX04	Communicative English/ Engineering Graphics	10,12	2/2	0	0/2	2/3
2	23MAX02/ 23MAX01	Differential Equations and Vector calculus/Linear Algebra& Calculus	1	3/3	0	0	3/3
3	23CYX01/ 23PYX01	Chemistry /Engineering Physics	1/1	3/3	0	0	3/3
4	23BEX02/ 23BEX01	Basic Civil & Mechanical Engineering/ Basic Electrical and Electronics Engineering	1,12/1,12	3/3	0	0	3/3
5	23CS201	Data Structures(CSE,CSE-AI&DS,CSE-AI&ML, IT)	1,12	3/3	0	0	3/3
6	23CYX03/ 23PYX02	Chemistry Lab / Engineering Physics Lab	4	0	0	2/2	1/1
7	23BEX06/ 23BEX05	Engineering Workshop/Electrical & Electronics Engineering workshop	1,9,10/1,5,10	0	0	3/3	1.5/1.5
8	23BEX08	IT Workshop/-	4/4	0	0	2/-	1/-
9	23HSX02	Communicative English Lab/-	-/10,12	0	0	2/-	1/-
10	23CS202	Data Structures Lab (CSE, CSE-AI&DS, CSE-AI&ML, IT)	2,3,4,5	0	0	3/3	1.5/1.5
11	23HSX11	-/ECA (Yoga/ Sports)		-	-	-/1	-/0.5
12	23HSX12	-/CCA (NSS/NCC/Community Service)		-	-	-/1	-/0.5
			Total	14/14	0	12/12	20/20
Third Semester							
1	23CS301	Problem Solving using Python	1,2,3,5,12	3	-	2	4
2	23ML302	Artificial Intelligence	1,2,3,PSO1,PSO2	3	-	-	3
3	23CS303	Design and Analysis of Algorithms	1,2,12, PSO1	3	-	-	3
4	23CS304	Digital Logic Design	1, 2,3,4,5,PSO1	3	-	2	4
5	23DS305	Mathematical Foundation for Data Science	1,2,3,4,12	3	-	-	3
6	23CS306	Object Oriented Programming with JAVA	1,2,3,12,PSO2	3	-	-	3
7	23CS307	Design and Analysis of Algorithms Lab	1,2,3,4,10,12,PSO1	-	-	3	1.5
8	23CS308	JAVA Lab	1,2,3,4,10,12,PSO2	-	-	3	1.5
9	23BEA01	Environmental Studies	1,7	-	-	-	-
10	23ESX01	Employability Skills I	1,2,3,4,5,9,10,11,12	0	-	2	-
			Total	**18**	**-**	**12**	**23**

Fourth Semester							
1	23IT304	Database Management Systems	1,2,3,4,12,PSO2	3	-	-	3
2	23IT403	Operating Systems	1,2,3,4,7,12	3	-	-	3
3	23CS403	Computer Organization and Architecture	1,2,3,12,PSO1	3	-	-	3
4	23MA404	Probability and Statistics using python	1,2,4,10,12	3	-	2	4
5	23ML405	Foundations of Machine learning	1,2, 4, PSO1,PSO2	3	-	-	3
6	23IT308	Database Management Systems Lab	1,2,3,4,5,9,PSO2	-	-	3	1.5
7	23ML407	Foundations of Machine learning Lab	1,2,3,4,5,PSO1,PSO2	-	-	3	1.5
8	23ESX01	Employability Skills I	1,2,3,4,5,9,10,11,12	0	-	2	2
			Total	**15**	**-**	**10**	**21**

Fifth Semester							
1	23IT405	Web Technologies	1,2,3,5,PSO1,PSO2	3	-	2	4
2	23ML502	Neural Networks	1,2,4,5,8,PSO1,PSO2	3	-	-	3
3	23DS503	Data Analytics &Visualization Techniques	1,2,3,4,5,12,PSO1,PSO2	3	-	2	4
4	23ML504	Computer Networks	1, 2,4,5,7,12,PSO1	3	-	-	3
5		Elective I (Professional Elective)		3	-	-	3
6		Elective II (Open Elective I)		3	-	-	3
7	23ML507	Neural Networks Lab	1,2,4,5,8,PSO1,PSO2	-	-	3	1.5
8	23TPX01	Term Paper	1,2,4,5,6,8,9,10,12,PSO1, PSO2	-	-	3	1.5
9	23ESX02	Employability Skills II	1,2,3,4,5,9,10,11,12	0	-	2	-
10	23SIX01	Summer Internship I	All PO's & PSO's	-	-	-	1
			Total	**18**	-	**12**	**24**
Sixth Semester							
1	23ML601	Deep Learning Techniques	1,2,3,4,5,PSO1,PSO2	3	-	-	3
2	23ML602	Automata Theory and Language Processors	1,2,3,5,12	3	-	-	3
3	23CS603	Software Engineering	1,2,3,5,8,11, PSO1	3	-	-	3
4		Elective III (Professional Elective)		3	-	2	4
5		Elective IV (Open Elective II)		3	-	-	3
6	23ML606	Deep Learning Techniques Lab	1,2,3,4,5,8,PSO1,PSO2	-	-	3	1.5
7	23MPX01	Mini Project	All POs & PSOs	-	-	3	1.5
8	23ESX02	Employability Skills II	1,2,3,5,6,8,10,12	0	-	2	2
9	23ATX01	Environmental Studies	1,7	-	-	-	-
10	23ATX02	Professional Ethics and Human Values	**----**	-	-	-	-
11	23ATX---	Audit Course	6,7,8,9,10,12	-	-	-	-
			Total	**15**	-	**10**	**21**
Seventh Semester– AI & ML							
1		Elective V (Professional Elective)		3	-	-	3
2		Elective VI (Professional Elective)		3	-	-	3
3		Elective VII (Open Elective III)		3	-	-	3
4	23SIX02	Summer Internship II	All PO's & PSO's	-	-	-	1
5	23PWX01	Project	All PO's & PSO's	-	-	16	8
			Total	**9**	**-**	**16**	**18**
Eighth Semester-AI & ML							
1		Elective VIII (Professional Elective)		-	-	-	3
2		Elective IX (Open Elective IV)		-	-	-	2
3	23FIX01	Full Semester Internship (FSI)	All PO's & PSO's	-	-	-	8
			Total	**6**	**-**	**-**	**13**

List of Electives

Language Electives							
No.	**Course Code**	**Course**	**POs**	**Contact Hours**			
				L	**T**	**P**	**C**
1	23HSX03	Advanced Communicative English	10,12	2	-	-	2
2	23HSX04	Communicative German		2	-	-	2
3	23HSX05	Communicative French		2	-	-	2
4	23HSX06	Communicative Japanese		2	-	-	2
5	23HSX07	Communicative Spanish		2	-	-	2
6	23HSX08	Communicative Korean		2	-	-	2
7	23HSX09	Communicative Hindi		2	-	-	2

Elective I							
Career Path I							
1	23MLC11	Computer Vision & Pattern Recognition(ML Ops)	1,2,3,12,PSO1	3	-	-	3
2	23CSC21	Backend Programming Languages (Full Stack Developer)	1,2,3,5,PSO1, PSO2	3	-	-	3
3	23MLC31	Fundamentals of Cloud Computing (Cloud Computing)	1,2,3,4,5,PSO1, PSO2	3	-	-	3
4	23CS004	Principles of Programming Languages	1, 2, 3, 4,PSO1	3	-	-	3
5	23CS005	Mobile Computing	1,3,8,10,12,PSO2	3	-	-	3
6	23CS006	Distributed Operating Systems	1,2,3,12,PSO1, PSO2	3	-	-	3
Elective III							
Career Path II							
1	23MLC12	Machine Learning for Business Intelligence(ML Ops)	1,2,3,4,5,PSO1, PSO2	3	-	2	4
2	23CSC22	Web Application Frameworks(Full Stack Developer)	1,2,3,5,PSO1,PSO2	3	-	2	4
3	23MLC32	Cloud Services using AWS(Cloud Computing)	1,2,3,4,5,PSO1	3	-	2	4
4	23CS007	Cloud Computing Essentials	1,3,6,7,8,9	3	-	2	4
5	23CS008	Cryptography and Network Security	1,2,3,6,8,12	3	-	2	4
Elective V							
Career Path III							
1	23MLC13	Conversational AI(ML Ops)	1,2,4,12,PSO1	3	-	-	3
2	23CSC23	Web Application Databases (Full Stack Developer)	1,3,5,PSP1	3	-	-	3
3	23MLC33	Cloud Security Essentials(Cloud Computing)	1,2,3,4,5,PSO1	3	-	-	3
4	23IT017	Social Network Analysis	2, 4, 5,12	3	-	-	3
5	23ML001	Human Computer Interaction	1,2, 3,12,PSO1	3	-	-	3
6	23CS012	Wireless Adhoc Networks	1,2,3,6,7,10	3	-	-	3
Elective VI							
1	23ML002	Large Language Models	1,2,4,12,PSO1	3	-	-	3
2	23CS015	Software Project Management	1,2,3,11,12,PSO2	3	-	-	3
3	23ML003	Reinforcement Learning	1,2,3,PSO1,PSO2	3	-	-	3

Elective VIII: Professional Elective							
1	23CS018	Information Retrieval Systems	1,2,3,4,10,PSO2	-	-	-	3
2	23CS019	Fundamentals of Devops	1,2,3,11,PSO2	-	-	-	3
3	23DS003	Cyber Security	1,3,4,5,PSO1				3

Audit Course							
1	23AT001	Communication Etiquette in Workplaces	-	-	-	-	-
2	23AT002	Contemporary India: Economy, Policy and Society	-	-	-	-	-
3	23AT003	Design The Thinking	-	-	-	-	-
4	23AT004	Ethics and Integrity	-	-	-	-	-
5	23AT005	Indian Heritage and Culture	-	-	-	-	-
6	23AT007	Intellectual Property Rights and Patents	-	-	-	-	-
7	23AT008	Introduction to Journalism	-	-	-	-	-
8	23AT009	Mass Media Communication	-	-	-	-	-

9	23AT010	Science, Technology and Development	-	-	-	-	-
10	23AT011	Social Responsibility	-	-	-	-	-
11	23AT012	The Art of Photography and Film Making	-	-	-	-	-
12	23AT013	Gender Equality for Sustainability	-	-	-	-	-
13	23AT014	Women in Leadership	-	-	-	-	-
14	23AT015	Introduction to Research Methodology	-	-	-	-	-
15	23AT016	Climate Change and Circular Economy	-	-	-	-	-

B. Tech. (Honors)							
Domain I (Data Engineering)							
1	23CSH11	Advanced Data Structures	2,3,4	4	-	-	4
2	23CSH12	Advanced Databases	2,3,4	4	-	-	4
3	23CSH13	Programming, Data Structures and Algorithms Using Python	2,3,4,5	4	-	-	4
4	23CSH14	Bioinformatics	2,3	4	-	-	4
Domain II (Modern Software Engineering)							
1	23CSH21	DevOps	1,3,5,8,10	4	-	-	4
2	23CSH22	Design Patterns	2,3	4	-	-	4
3	23CSH23	Advanced Software Engineering	1,3,4, PSO1	4	-	-	4
4	23CSH24	Robotic Process Automation	3,5, 8, PSO2	4	-	-	4
Domain III(Security)							
1	23CSH31	Fundamentals of Systems Security	1,2	4	-	-	4
2	23CSH32	Python Programming for Security	2,3,4	4	-	-	4
3	23CSH33	Management of Information Security	3,6,7	4	-	-	4
4	23CSH34	Computer Forensics	2,3	4	-	-	4
Domain IV (User Interface Design)							
1	23CSH41	Computer Graphics	1,2,3,4	4	-	-	4
2	23CSH42	Multimedia Systems	3,4	4	-	-	4
3	23CSH43	Human Computer Interaction	2,3	4	-	-	4
4	23CSH44	Mobile Programming	3,4	4	-	-	4

3rd Semester

23CS301 Problem Solving using Python

3 0 2 4

Course Outcomes

At the end of the course, students will be able to

1. Explain the fundamentals of Python with syntax and semantics.
2. Apply conditional and control flow statements, and analyze strings and regular expressions for pattern matching and text processing.
3. Apply the built-in data structures such as dictionaries, sets, lists, and tuples for data manipulation.
4. Apply the concepts of functions and functional programming techniques.
5. Apply the file Handling and exception handling techniques.
6. Describe the principles of object oriented programing in Python and demonstrate the use of Modules and Packages.

CO – PO Mapping

CO's	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	2	-	3	-	-	-	-	-	-	1	-	-
CO2	2	3	3	-	3	-	-	-	-	-	-	2	-	-
CO3	2	3	3	-	3	-	-	-	-	-	-	2	-	-
CO4	2	2	3	-	3	-	-	-	-	-	-	2	-	-
CO5	2	3	2	-	3	-	-	-	-	-	-	2	-	-
CO6	3	2	3	-	3	-	-	-	-	-	-	2	-	-

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I **12+8 Hours**

Introduction: History and need for python programming, Identifiers, Statements, Variables, Keywords, Input-Output statements, Indentation, command line arguments.

Data Types - Integers, Strings, Booleans. **Operators** - Arithmetic, Comparison/Relational, Assignment, Logical, Bitwise, Membership, Identity. **Expressions and order of evaluations -** Precedence and Associativity. **Built-in Functions** – Importing from Packages. **Conditional Statements**: If, Elif, Else statements, iteration statements - While, For - Break, Continue and pass statements.

Applications of Python, REPL (Read, Evaluation, Print, Loop)

Practical Components

1. Basic Operations: Validating and Converting Data Formats and Numeric Systems in Python
2. Control Statements: Implementing Fibonacci Series and Advanced List Operations
3. Array Manipulations : Matrix Operations: Transpose, Symmetry Check, and Multiplication

Unit II **12+6 Hours**

Collections: Lists: Create, Access, Slicing, Negative indices, List methods, List comprehensions – Tuples: Create, Indexing and slicing, Operations on tuples – Dictionary: Create, add, and replace values, Operations on dictionaries – Sets: Creation and operations.

Strings: Strings: Comparison, Formatting, Slicing, Splitting, Stripping . **Regular Expressions**: Matching, Search and replace, Patterns.

Collections.deque, grouping with Dictionaries

Practical Components

1. Array Manipulations: Intersection, Kth Largest Element, and Maximum Subarray in Lists
2. Working with Dictionaries: Word Frequency Counting, Dictionary Merging, and Filtering
3. String Manipulation: Acronyms, Valid Parentheses, Anagram Check, and Common Prefix
4. Regular Expressions for Email Validation, Phone Number Extraction, and Password Strength Checking

Unit III **12 +8 Hours**

Functions- Defining Functions, Calling Functions, Types of Arguments, Recursion, Anonymous Functions: Lambda Expressions, Global and Local Variables. Functional Programming: map, filter and reduce, decorators. **Modules and Packages-** Built-in modules, User-Defined modules, itertools module. **Numpy and Pandas:** Introduction to Numpy and Pandas packages of python, Basic operations on Numpy and Pandas.

Higher order function, keyworded variable length arguments.

Practical Components

1. Recursive Functions: Prime Factorization, Subsets Generation, and Fibonacci Sequence
2. Exploring Default and Keyword Arguments, Variable Scopes, and Lambda Functions
3. Functional Programming: Map, Filter, and Reduce Operations
4. Itertools Applications: Counting, Cycling, Permutations, Combinations, and Sliding Windows
5. Working with Numpy Arrays and Pandas DataFrames

Unit IV **12+10 Hours**

Classes and Objects: Defining and instantiating Class, Class Attributes and Methods, Constructor, self-keyword - inheritance – overriding methods from parent class – Instances as Attributes, inner class. **File Handling-**Creating files, operations on files - Open, Close, Read and Write - Working with File Paths - File and Directory Operations. **Exception Handling-** Exception, try except block, Raising Exceptions, User Defined Exception.
Polymorphism, Abstract class and abstract methods.

Practical Components

1. Object-Oriented Programming: Creating and Using Classes and Objects
2. Inheritance and Method Overloading/Overriding in Python
3. File Handling and Simple File Explorer Implementation
4. Exception Handling: Custom Exceptions

Total: 48+32 Hours

Textbook (s)

1. Richard L. Halterman, "Fundamentals of Python Programming", 3rd Edition, Southern Adventist University, 2019.
2. Matthes, E. ,"Python Crash Course: A Hands-On, Project-Based Introduction to Programming", 2nd Edition, No Starch Press, 2019.
3. Willaim Mckenny, "Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython" 2nd Edition, O'Reily 2017.

Reference(s)

1. Wesley J Chun, "Core Python Applications Programming", 3rd Edition, Pearson Education India, 2015.
2. Mertz, D., "Functional Programming in Python", 1st Edition, O'Reilly Media,2015.
3. Kenneth A. Lambert. "Fundamentals of Python: First Programs", 2nd Edition, Publisher: Cengage Learning 2018
4. Python Programming: A Modern Approach, VamsiKurama, Pearson 2017 Learning Python, Mark Lutz, Orielly, 5th Edition.

Internal Assessment Pattern

Cognitive Level	Int. Test 1 (%)	Int. Test 2 (%)	Assignment Test/OBE (%)
Remember	20	10	--
Understand	50	40	--
Apply	30	30	--
Analyze	--	20	--
Evaluate	--	--	--
Create	--	--	--
Total (%)	100	100	--

Remember

1. What are the benefits of Python?
2. What is an Interpreted language?
3. List any four string handling functions

Understand

1. Explain lists and tuples?
2. Differentiate between the two?
3. What are Dict and List comprehensions?

Apply

1. Explain user defined exception
2. Justify multiple inheritance is supported in python?
3. Design a simple GUI page using Tkinter

23ML302 Artificial Intelligence

3 0 0 3

Course Outcomes

1. Summarize the fundamental concepts of Artificial Intelligence.
2. Understand the control strategies and production systems.
3. Illustrate the Concepts of Heuristic Search Techniques.
4. Demonstrate working knowledge of reasoning in the presence of incomplete and/or uncertain information
5. Make use of concept of Game Playing Algorithms.
6. Outline the concept of Planning System

CO–PO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
1	2	1	2	-	-	-	-	-	-	-	-	-	2	2
2	2	2	3	-	-	-	-	-	-	-	-	-	3	3
3	3	1	1	-	-	-	-	-	-	-	-	-	3	3
4	2	2	2	-	-	-	-	-	-	-	-	-	3	3
5	1	3	2	-	-	-	-	-	-	-	-	-	3	3
6	1	2	2	-	-	-	-	-	-	-	-	-	3	3

3. Strongly linked | 2-Moderately linked | 1-Weaklylinked

Unit I — 12 Hours

Introduction

Definition, AI problems, AI techniques, Defining problem as a state space search, Production systems-Control Strategies, Problem characteristics, Production system characteristics. Problems – tic-tac-toe, 8-puzzle problem, Chess problem, Water Jug Problem, Missionaries and cannibals problem, Monkey and banana problem.

Application of AI-Tower of Hanoi problem- *Travelling Salesman Problem*

Unit II — 12 Hours

Heuristic Search Techniques

Generate-and-test, Hill climbing, Best-first-search – OR Graphs – A* Algorithm, Problem reduction – AND-ORGraphs – AO* Algorithm, constraint satisfaction - cryptarithmetic problem.

Depth-first search – Breadth-first search- Data abstraction

Unit III — 12 Hours

Knowledge representation

Knowledge Representation Issues, Representation and mapping, Approaches to Knowledge Representation, Frame Problem, Propositional logic, Predicate logic, Resolution, weak slot and filler structure-semantic nets.

Filler structures-conceptual dependency-scripts

Unit IV — 12 Hours

Game Playing and Planning

Mini-max search, Alpha-beta cutoffs, planning system, Block world problem, goal stack planning, hierarchical planning, Natural language processing, syntactic processing. Decision trees, Perception, Vision, Speech recognition.

Learning - Navigation – Manipulation - Robot Architecture

Textbook (s)

1. E. Rich K .Knight, and B. Nair, Artificial Intelligence, 3rdEdition, TMH, 1 July 2017
2. Russel Norvig, Artificial Intelligence A modern Approach, 3rd Edition, Pearson Education, 2010

Reference (s)

1. Patrick henry Winston, Artificial Intelligence, third edition, Pearson Education Asia, 2005
2. Dan W. Patterson, Introduction to Artificial intelligence and Expert Systems, 2nd Edition, PHI, 2009

Internal Assessment Pattern

Cognitive Level	Int. Test 1 (%)	Int. Test 2 (%)	Open Book Exam (%)
Remember	40	40	--
Understand	30	40	--
Apply	30	20	70
Analyze	--	--	30

Evaluate	--	--	--
Create	--	--	--
Total (%)	100	100	100

SAMPLE QUESTION (S)

Remember

1. State the tasks, which are associated with A.I.
2. Give an example of crypt-arithmetic problem.
3. Define Speech Recogonition.
4. Recall Intelligence

Understand

1. Explain Turing Test.
2. Explain the characteristics of production system.
3. Write A* algorithm in detail and explain its functionality to solve 8-puzzle problem.
4. Explain the semantic analysis phase done through case grammars in Natural Language understanding

Apply

1. Distinguish between weak and strong slot filler structures
2. Discuss the tic-tac-toe problem in detail and explain how it can be solved using AI technique
3. Construct semantic net representation for the following:
   (i) Pompeian (Marcus), Blacksmith (Marcus)
   (ii) Mary gave the green flowered vase to her favorite cousin.

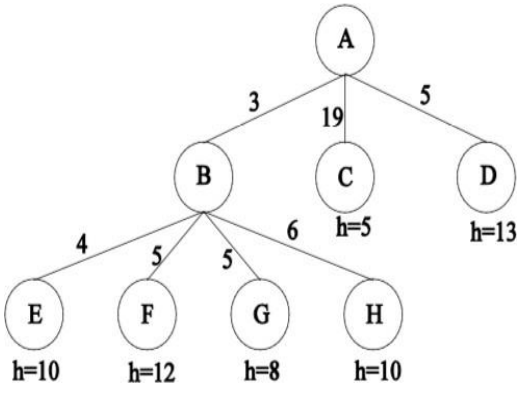
Analyze

1. Show that the tower of Hanoi problem can be classified under the area of AI. Give a state space representationof the problem.
2. Give an example of a problem for which breadth-first search would work better than depth-firstsearch.Justify your answer.
3. Consider the following sentences:
   John likes all kinds of food.
   Apples are food.
   Chicken is food
   Anything anyone eats and isn't killed by food.
   Bill eats peanuts and is still alive.
   Sue eats everything Bill eats.
   (a) Translate these sentences into formula in predicate logic.
   (b) Convert the formulas of part a into clause form.
   Prove that John likes peanuts using resolution

Open Book Exam Questions

Question 1:

Consider the following search tree produced after expanding node A, where each arc is labeled and the leaves are labeled with the value of a heuristic function, h.



(i) Identify the node, which is expanded next by each of the following search methods?

(a) Breadth First Search

(b) Depth-First Search

(c) Best-First Search

(d) A* Search

(e) AO* Search

(ii) Construct the Min Max Tree for the above figure (ignoring the cost values)

(iii) Identify whether the tree is eligible for pruning or not, if it is yes trace the alpha beta prunes

23CS303 Design and Analysis of Algorithms

3 0 0 3

Course Outcomes

At the end of the course, students will be able to

1. Analyse the performance of algorithms using space and time complexity, and asymptotic notations.
2. Implement algorithms using brute force and divide and conquer approaches.
3. Apply advanced data structures and graph algorithms to solve complex computational problems effectively.
4. Demonstrate the ability to solve problems efficiently using greedy approach.
5. Develop problem-solving skills by tackling complex computational problems using dynamic programming and backtracking.
6. Apply branch and bound techniques to solve complex problems, and understand the concepts of NP-completeness to identify NP-complete problems.

COs–POs Mapping

CO's	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	2	-	-	-	-	-	-	-	-	1	3	-
CO2	2	2	3	-	-	-	-	-	-	-	-	1	3	-
CO3	3	3	3	-	-	-	-	-	-	-	-	2	3	-
CO4	3	2	3	-	-	-	-	-	-	-	-	2	3	-
CO5	3	2	2	-	-	-	-	-	-	-	-	2	3	-
CO6	3	3	2	-	-	-	-	-	-	-	-	2	3	-

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I **12 hours**

Algorithm Analysis and Divide and Conquer

Fundamentals of algorithmic problem solving - Analysis framework - Performance Analysis: Space and Time complexity - Growth of Functions: Asymptotic Notations - Mathematical Analysis of Non-recursive and Recursive algorithms - Master's theorem - Brute Force String Matching Algorithm- Divide and Conquer: Merge sort, Quick sort, Fibonacci search.

Interpolation search and Radix sort

Unit II **12 hours**

Advanced Data Structures and Graph Algorithms:

Balanced Search Trees: AVL trees, B-trees - Heaps: Heapsort - Disjoint set: Find and Union Operations - Graphs: Introduction, Representation - Graph Traversals: BFS, DFS, Topological sort.

Tries and Strongly Connected Components

Unit III **12 hours**

Greedy and Dynamic Programming

Greedy Method: General method, Job sequencing with deadlines, Optimal Merge Patterns - Minimum cost spanning trees: Prim's and Kruskal's Algorithms - Single source shortest path problem: Dijkstra's Algorithm - Dynamic Programming: General method, Matrix chain multiplication, 0/1 knapsack problem, All pairs shortest path problem: Floyd Warshall Algorithm.

Graph coloring problem, Huffman Trees and Codes

Unit IV **12 hours**

Backtracking and Brach and Bound

Backtracking: General method, n-queen problem, sum of subsets problem, Hamiltonian Circuit problem. Branch and Bound: General method, Knapsack Problem, Traveling sales person problem, Assignment Problem. Complexity Classes : P, NP and NP-Complete Problems.

Exhaustive Search, Space and Time Trade-Offs

Total: 48hours

Textbook (s)

1. Introduction to The Design and Analysis of Algorithms, 3rd Edition, AnanyLevitin, Pearson Education, 2017.
2. Introduction to Algorithms, second edition, T.H.Cormen, C.E.Leiserson, R.L. Rivest, and C.Stein, PHI Pvt. Ltd./ PearsonEducation
3. Fundamentals of Computer Algorithms, Ellis Horowitz, SatrajSahni and Rajasekaran, University press.

Reference (s)

1. Design and Analysis of algorithms,Aho,Ullman and Hopcroft,Pearsoneducation.
2. Algorithms–Richard Johnson BaughandM arcus Schaefer,Pearson Education.
3. www.geeksforgeeks.org
4. www.hackerearth.com
5. www.tutorialspoint.com

Internal Assessment Pattern

Cognitive Level	Int. Test 1 (%)	Int. Test 2 (%)	Open Book Exam (%)
Remember	20	20	--
Understand	40	40	--
Apply	20	20	50
Analyze	20	20	50
Evaluate	--	--	--
Create	--	--	--
Total (%)	100	100	100

Remember :

1. Define time complexity and space complexity in the context of algorithms.
2. What is the purpose of asymptotic notations like Big O, Big Omega, and Big Theta?
3. List the steps involved in the Merge Sort algorithm.
4. State the Master's theorem for divide-and-conquer recurrences.
5. What are the key differences between AVL trees and B-trees?

Understand :

1. Explain how the Brute Force String Matching Algorithm works and its time complexity.
2. Describe the basic idea of the Divide and Conquer approach with an example.
3. How does the Floyd-Warshall algorithm find the shortest paths between all pairs of vertices in a graph?
4. Explain the difference between a min-heap and a max-heap.
5. What is the significance of the depth-first search (DFS) in finding strongly connected components in a graph?

Apply :

1. Apply the Quick Sort algorithm to the following list of numbers: [34, 7, 23, 32, 5, 62].
2. Using Dijkstra's algorithm, find the shortest path from a given source vertex to all other vertices in a weighted graph.
3. Given a set of jobs with deadlines and profits, use the Greedy method to find the maximum profit sequence.
4. Implement the BFS algorithm on a graph represented by an adjacency list.
5. Use the dynamic programming approach to solve the 0/1 Knapsack problem with given weights and values.

Analyze :

1. Analyze the time complexity of the Radix Sort algorithm and compare it with other sorting algorithms like Quick Sort and Merge Sort.
2. Compare and contrast the space efficiency of AVL trees and B-trees.
3. Analyze the advantages and disadvantages of using the Backtracking approach for solving the n-queen problem.
4. Discuss the time complexity of the Floyd-Warshall algorithm and its feasibility for large graphs.
5. Examine the differences between P, NP, and NP-Complete problems, and provide examples of each.

23CS304 Digital Logic Design

3 0 2 4

Course Outcomes

1. Demonstrate different number systems, its conversions, and binary arithmetic.
2. Classify logic circuits using basic Logic gates and simplify logic expressions using theorems, K-map.
3. Design and implement logical devices using combinational circuits.
4. Demonstrate and compare the construction of programmable logic devices and different types of ROM.
5. Analyze Sequential circuits like latches and flip-flops.
6. Analyze and Design sequential circuits like Registers and Counters.

CO-PO Mapping

CO's	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	-	-	-	-	-	-	-	-	-	-	2	-
CO2	3	3	1	1	1	-	-	-	-	-	-	-	2	-
CO3	3	1	3	2	2	-	-	-	-	-	-	-	3	-
CO4	2	3	3	2	2	-	-	-	-	-	-	-	3	-
CO5	3	2	2	-	2	-	-	-	-	-	-	-	3	-
CO6	3	3	3	2	2	-	-	-	-	-	-	-	3	-

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I **12+6 Hours**

Number systems and Boolean algebra

Review of Number Systems, Conversion of Numbers from One Radix to Another Radix, Complements, Representation of Negative Numbers, Binary Arithmetic, Binary Codes, Error detecting & correcting codes, Basic Theorems and Properties of Boolean Algebra, Digital Logic Gates, Universal Gates.

IC specifications and pin diagram of gates

Practical Components

Realization of Logic gates and verification of Truth tables
Realization of basic gates using Universal gates (NOR)
Realization of basic gates using Universal gates (NAND)

Unit II **12+12Hours**

Boolean Function Minimization

Minimization of Switching Functions using K-Map up to 4-variables, Prime implicants, don't care combinations, Minimal SOP and POS forms

Combinational Arithmetic Logic Circuits-1

Adders, Subtractors, Binary Multiplier-*parity bit Generator*, Ripple carry adder, Multiplexer, De-Multiplexer, Encoder, Priority encoder, Decoder, MUX Realization of switching functions.

Code Converters, Magnitude Comparator

Practical Components

1. Implementation of Half Adder
2. Implementation of Full Adder
3. Implementation and verification of Encoder
4. Implementation and verification of Decoder
5. Implementation and verification of MUX
6. Implementation and verification of DE-MUX

Unit III **12+6Hours**

Programmable Logic Devices and Sequential Logic Circuits-1

Basic PLD's-ROM-PROM-PLA-PAL - Realization of Switching functions using PLD's Classification of Sequential Circuits (Synchronous and Asynchronous): Latches and Basic Flip-Flops-Truth Tables and Excitation Tables

Conversion of flip-flops

Practical Components

1. Realization of Flip-Flops using ICs (SR Flip Flop)
2. Realization of Flip-Flops using ICs (JK Flip Flop)
3. Realization of Flip-Flops using ICs (D and T Flip Flop)

Unit IV **12+8 Hours**

Sequential Logic Circuits - II

Design of Registers - Buffer Register - Control Buffer Registers - Bidirectional Shift Registers - Universal Shift Register - Design of Synchronous Counters – Ripple counter, Up-down Counters, Design of Asynchronous Counters-Variable Modulus Counters (Mod-2,4,6,10 & 16), Ring Counter, Johnson Counter, Sequence generator.

Sequence detector

Practical Components

1. Verification of SISO and SIPO Shift Registers
2. Verification of PIPO and PISO Shift Registers
3. Implementation of Synchronous Counter
4. Implementation of Asynchronous Counter

Total: 48+32 Hours

Textbook (s)

1. Digital Design 6th Edition by M Morris Mano, PEARSONINDIA,2018.
2. Charles H. Roth, Fundamentals of Logic Design,3rdEdition, ThomsonPublications,2014
3. John F. Wakerly, Digital Design Principles & Practices, 4thEdition, PHI/ Pearson EducationAsia,2008

Reference (s)

1. ZviKohavi, Switching & Finite Automata theory, 2nd Edition,TMH,2008
2. R P Jain,Modern Digital Electronics, 3rd Edition,TMH,2003
3. A. Anand Kumar, Switching Theory and Logic Design, 3rd Edition,PHP,2016.

Internal Assessment Pattern

Cognitive Level	Int. Test 1 (%)	Int. Test 2 (%)	Assignment Test/OBE (%)
Remember	20	10	--
Understand	50	40	--
Apply	30	30	--
Analyze	--	20	--
Evaluate	--	--	--
Create	--	--	--
Total (%)	100	100	--

SAMPLE QUESTION (S)

Remember

1. Retrieve the decimal value of the fractional binary number0.1011
2. List any two postulates of Boolean algebra
3. List the four uses of Multiplexer.
4. Define Flip flop and Latch
5. Define Synchronous sequential circuit

Understand

1. Explain working functionality of Programable Logic Devices(PLD).
2. Represent the following Boolean expression to SOP and POS form: A+BC'+ABD'+ABCD
3. Represent the Boolean function T=F (w, x, y, z) = ∑ m (0,1,2,4,5,7,8,9,12,13) by using 8 to1mux.
4. Illustrate a half adder using NAND –NAND

5. Represent a T flip flop using JK flipflop.
6. Differentiate between Bi-directional and Universal Shift-Registers.

Apply

Find a circuit that has no static hazards and implements the Boolean function F(A,B,C,D) = Σ (0,2,6,7,8,,10,12)..

Construct a four input NAND gate using gates with 2 inputs. Write down the truth table. Do the same for a 4 input NOR gate.

Implement MOD-6 and Johnson Counter.

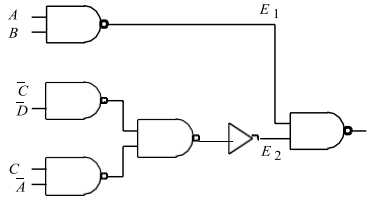
Construct a combinational circuit to convert BCD to EX-3code.

Implement a combinational logic circuit, which can compare two bits binary numbers.

Analyze

1. Analyze the following in two ways to get sums of product and product of sums equations. Then show how you can get one from the other using DeMorgan's Theorem

2. *F*



3. Why the input variables to a PAL are buffered
4. Why the fixed format in not used and floating format issued
5. What happens when an electric signal is grounded?
6. Design a Sequence detector to detect10111001

23DS305 Mathematical Foundation for Data Science

3 0 0 3

Course Outcomes

After undergoing this course, the learners are enabled to

1.Demonstrate Vector Spaces and their applications.
2.Solve Problems involving, Linear Transformations and Inner product spaces
3.Apply generating functions and Characteristic roots to solve recurrence relations
4.Derive statements using the rules of inference for Mathematical logic.
5.Understand the basic knowledge of Graph theory and applications and the rules of inference for Predicate logic.
6.Construct the minimum spanning tree using different algorithms.

CO-PO Mapping

CO's	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	2	2	-	-	-	-	-	-	-	1	-	-
CO2	3	2	3	3	-	-	-	-	-	-	-	1	-	-
CO3	3	3	3	2	-	-	-	-	-	-	-	1	-	-
CO4	3	2	2	1	-	-	-	-	-	-	-	1	-	-
CO5	3	3	2	1	-	-	-	-	-	-	-	1	-	-
CO6	3	3	3	2	-	-	-	-	-	-	-	1	-	-

3-Strongly linked | 2-Moderately linked | 1-Weaklylinked

Unit I

Vector Spaces and Linear Transformations **12 hours**

Vector Spaces: Definition, Properties of Vector Spaces, Subspaces, Algebra of Subspaces, Basis and Dimension, Linear Span, Linear Independence and Dependence of vectors.

Linear Transformations: Definition, Properties, Range and Null Spaces, Rank-Nullity theorem, Matrices of Transformations. (All theorems without Proof)
Singular Value Decomposition

Unit II

Inner Product Space and Recurrence Relations **12 hours**

Inner Product Spaces: Definition, Properties of Inner Product Spaces, Norm of a vector, Cauchy-Schwarz's Inequality (without Proof), Complete orthonormal set, Gram-Schmidt Orthogonalization process.

Recurrence Relations: Generating Functions, Calculating Coefficient of Generating Functions, Solving Recurrence Relations by Substitution method, Solving Recurrence Relations by Method of Characteristic roots, Solving Recurrence Relations by generating functions

Bessel's inequality

Unit III **12 hours**

Mathematical Logic

Propositional Logic: Statements and Notations, Connectives, Well-formed Formulas, Truth Tables, Tautology, Equivalence Implication, Normal Forms, Rules of Inference for Statement Logic, Proof Techniques, Proof by Contradiction, Method of Induction.

Predicative Logic: Quantifiers, Universal Quantifiers, Free & Bound Variables, Rules of Inference for Predicate Logic.
Relations in Databases

Unit IV **12 hours**

Graph Theory

Basics of Graph Theory: Representation of Graphs, Degree of a Graph, Complete Graphs, Regular Graphs,Bipartite Graphs, Walk – Path – Circuit – Cycle, Planar Graphs, Euler's Formula, Isomorphism, Euler Circuit and Hamilton Cycle, Chromatic Number**.**

Trees: Definition and properties of tree, Spanning Tree, Breadth First Search (BFS) algorithm, Depth First Search (DFS) algorithm, Minimum Spanning Tree, Kruskal algorithm, Prim's algorithm. *Applications of Graphs and Trees, Ternary Tree*

Total: 48 Hours

Textbook(s)

1. Gilbert Strang, "Linear Algebra and Learning from Data", Wellesley, Cambridge Press, 2019.
2. Kenneth H. Rosen, Discrete Mathematics and Applications, 7th Edition, Tata McGraw Hill, 2015.
3. J. L. Mott, A. Kandel & T. P. Baker, Discrete Mathematics for Computer Scientists & Mathematicians, Prentice Hall India, 2nd Edition, 2010.

Reference (s)

1. Tremblay J.P. and P. Manohar, Discrete Mathematical Structures with Applications to Computer Science, Tata McGraw Hill,2007.
2. Erwin Kreyszig, Advanced Engineering Mathematics, Wiley India, 2015.
3. K Hoffman and R Kunze, Linear Algebra, Pearson Education, 2017.

Internal Assessment Pattern

Cognitive Level	Int. Test 1 (%)	Int. Test 2 (%)	Assignment Test/ Open Book Exam (%)
Remember	30	30	--
Understand	40	40	--
Apply	30	30	60
Analyze	--	--	40
Evaluate	--	--	
Create	--	--	--
Total (%)	100	100	100

Remember

1. Define Vector Space.
2. State Hand shaking theorem.

Understand

1. Prove that "if n^2 is odd then n is odd" by the method of contradiction.
2. Explain about Euler circuit and Hamiltonian cycle with suitable examples.

Apply

1. Find rank of the transformation $T: R^2 \to R^3$ defined by $T(x, y) = (x + y, x - y, y)$ using Rank-Nullity Theorem.
2. Obtain the matrices U, V and S such that $A = USV^T$, where $A = \begin{bmatrix} 3 & 3 & 2 \\ 2 & 3 & -2 \end{bmatrix}$ using Singular Value Decomposition.

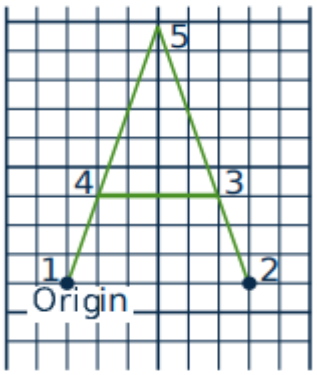
Analyze

1. Prove the validity of the following argument: "If I study, then I will pass in exam. If I do not go to cinema, then I will study. But I failed in exam. Therefore, I went to cinema."

2. Draw a graph which contains an Eulerian circuit but not an Hamiltonian cycle with justification

Open Book Exam

1. Construct an AVL tree by inserting the following elements in the order given: {10, 6, 9, 4, 8, 7, 2, 11, 12, 3, 5, 1}, and balancing as necessary. Delete the elements in the following order {9, 6, 2, 5, 10, 3, 1, 4, 7, 8, 12, 11} balancing as necessary. Show the intermediate steps of deletion when re-balance (rotation) is used and end result of the insertion.
2. Model the matrix for the figure given below.
   i. Draw the image corresponding to the matrix obtained by multiplying $\begin{bmatrix}1 & 0.2\\0 & 1\end{bmatrix}$ with obtained matrix in (i)
   ii. Sketch the image and write the corresponding matrix by rotating the following figure through the angle $\frac{\pi}{6}$ about the point $\begin{bmatrix}4\\5\end{bmatrix}$.



23CS306 Object Oriented Programming with JAVA

3 0 0 3

Course Outcomes

At the end of the course, students will be able to

1. Apply the object oriented programming concepts to develop modular and reusable software solutions.
2. Develop applications using different types of inheritances
3. Implement user-defined packages to promote code reusability and modular programming.
4. Analyze and recover runtime exceptions arise in the applications
5. Apply parallel processing applications using threads
6. Develop interactive applications for standalone and Internet

COs–POs Mapping

CO's	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	1	-	-	-	-	-	-	-	-	2	-	1
CO2	3	3	1	-	-	-	-	-	-	-	-	2	-	1
CO3	3	3	1	-	-	-	-	-	-	-	-	2	-	1
CO4	2	2	2	-	-	-	-	-	-	-	-	2	-	2
CO5	3	2	2	-	-	-	-	-	-	-	-	2	-	2
CO6	3	2	2	-	-	-	-	-	-	-	-	2	-	2

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I **12 Hours`**

Introduction to Java

Overview of Object Oriented Programming principles, Importance of Java to the Internet, Byte code, Methods, classes and instances, Data types, arrays, control statements, simple java program, Classes and Objects– constructors, methods, access control, this keyword, overloading methods and constructors, garbage collection, java.lang – String, StringBuffer, StringBuilder, Type wrappers – Integer, Character, Float, Double, Boolean.

Features of object oriented programming–Java History–Computer Programming Hierarchy–Role of Java Programmer in Industry

Unit II **12 Hours**

Inheritance, Packages & Interface

Inheritance: Basics, Using super, Multilevel Hierarchy, Method overriding, Dynamic Method Despatch, Using Abstract classes, Using final with inheritance, Sealed Classes.

Java Collections – Arrays, ArrayList, Hashtable, StringTokenizer, Vector.

Interface: Defining an interface, differences between classes and interfaces, implementing interface, variables in interface and extending interfaces, Default Methods in Interfaces, Static Methods in Interface

Packages: Defining, Creating and Accessing a Package, Understanding CLASSPATH, importing packages, Member access rules.

Nested–Inner Class & Anonymous Classes–Generic Class Types

Unit III **11 Hours**

Exception Handling & Multithreading

Exception handling: Concepts and benefits of exception handling, exception hierarchy, usage of try, catch, throw, throws and finally, built-in and User Defined Exceptions.

Multithreading: Definition thread, thread life cycle, creating threads, synchronizing threads, daemon threads.

Control Flow in Exceptions– JVM reaction to Exceptions– Inter Communication of Threads– Critical Factor in Thread–Deadlock

Unit IV **13 Hours**

Applets & Event Handling

Applets: Concepts of Applets, life cycle of an applet, types of applets, creating applets, passing parameters to applets, The AWT class hierarchy, user interface components labels, button, Text components.

Event Handling: Events, Delegation event model, handling mouse and keyboard events, Adapter classes, inner classes, Compare basic AWT components with swing components, More user interface components-canvas, scrollbars, check box, choices, lists panels–scroll pane, dialogs, menu bar, layout managers.

Applet Security Policies–Anonymous Inner classes a Short–cut to Event Handling–Java Utilities (java.util Package)

Total: 48 Hours

Textbook (s)

1. H. Schildt, Java: The complete reference, 7th Edition, TMH, 2006
2. T. A. Budd, An Introduction to Object–Oriented Programming, 3rd Edition, Addison Wesley Longman, 2002

Reference (s)

1. Dietal&Dietal, Java: How to Program, 8th Edition, PHI, 2010
2. C. S. Horstmann and G. Cornell, Core Java, Vol 1. Fundamentals, 7th Edition, Pearson Education, 2004
3. C. Horstmann, BIG JAVA Compatible with Java 5 & 6, 3rd Edition, Wiley Publishers, 2008

Internal Assessment Pattern

Cognitive Level	Int. Test 1(%)	Int. Test 2 (%)	Assignment Test/OBE (%)
Remember	25	15	10
Understand	30	15	30
Apply	25	40	30
Analyze	20	20	30
Evaluate	--	10	--
Create	--	--	--
Total (%)	100	100	100

SAMPLE QUESTION (S)

Remember

1. List all OOPs principles
2. Define class and object
3. Illustrate various data types supported by JAVA
4. State various access controls available in JAVA
5. Define inheritance
6. Define thread

Understand

1. Explain byte code
2. Explain the use of super, static and final keywords
3. Illustrate the use of try, catch, throw and throws
4. Java is purely object oriented programming language. Explain
5. Explain types of polymorphism with Example

Apply

1. Write statements that perform the following one-dimensional-array operations:
    a. Set the 10 elements of integer array counts to zero
    b. Add one to each of the 15 elements of integer array bonus
    c. Display the five values of integer array best Scores in column format
2. Create two arrays that store all even no's in one array and all odd no's in another array for a given set of elements
3. Write a java program to implements final variable, Method and classes
4. Explain the reason for the following code giving compiling errors.

```
abstractclassAbstractClass
{
  abstractvoidabstractMethod()
  {
    System.out.println("First Method");
  }
}
```

5. Write a java program to implement the following techniques
    1. Method Overloading
    2. Method Overriding
6. Write a Java program to implement Virtual function technique

Analyze

1. Compare and Contrast an abstract class and an interface
2. Differentiate method overloading and overriding
3. Differentiate abstraction and encapsulation
4. Identify the uses of calling a synchronize method inside a synchronize method
5. Identify memory leak problems in Java objects

23CS307 Design and Analysis of Algorithms Lab

0 0 3 1.5

Course Outcomes

At the end of the course, students will be able to

1. Analyze the performance of algorithms by implementing and measuring their space and time complexity using asymptotic notations.
2. Implement algorithms using brute force and divide and conquer approaches, and assess their efficiency.
3. Implement advanced data structures and graph algorithms to solve complex computational problems, and evaluate their performance.
4. Solve optimization problems efficiently using a greedy approach, and analyze the results.
5. Develop dynamic programming solutions to tackle complex computational problems, and measure their effectiveness.
6. Apply backtracking and branch and bound techniques to solve complex problems.

CO-PO Mapping

CO's	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	1	2	-	-	-	-	-	1	-	2	2	-
CO2	3	2	2	2	-	-	-	-	-	1	-	2	2	-
CO3	3	3	3	3	-	-	-	-	-	1	-	2	2	-
CO4	3	2	3	3	-	-	-	-	-	1	-	2	2	-
CO5	3	2	2	3	-	-	-	-	-	1	-	2	2	-
CO6	3	3	2	2	-	-	-	-	-	1	-	2	2	-

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

List of Experiments

1. Analysis of recursive algorithms
2. Analysis of Non-recursive algorithms
3. Solve the String matching problem using Brute Force approach
4. Solve sorting problem using divide and conquer approach
5. Implement heapsort algorithm using a max-heap.
6. Implement Simple Union and Find operation on disjoint-sets
7. Implement the graph traversal algorithm.
8. Implement topological sorting of a directed acyclic graph (DAG).
9. Find the minimum cost spanning tree for a given graph using greedy approach
10. Implement dijkstra's algorithm using greedy approach
11. Solve matrix chain multiplication problem using dynamic programming
12. Solve the 0/1 knapsack using dynamic programming
13. Implement Floyd algorithm using dynamic programming
14. Find the sum of subsets using backtracking technique
15. Solve N queen problem using the backtracking technique.
16. Find the solution for Assignment Problem using branch and bound method

Indicative list of Augmented Experiments:

1. Basic File System Simulation Using B-Trees
2. Implementation of Huffman Coding for Data Compression
3. Topological Sorting of Courses with Prerequisites
4. Implementation of Tries for Auto-Completion and Spell Checking
5. Route Optimization for a Delivery Service Using Traveling Salesman Problem (TSP) Algorithms
6. Implementation of a Priority Queue Using a Binary Heap for Task Scheduling
7. Graph Coloring Problem for Map Coloring
8. Dynamic Programming for Optimal Binary Search Tree Construction
9. Visualizing Strongly Connected Components in Social Network Analysis
10. Rod Cutting Problem

Reading Material (s)

1. Design and Analysis of Algorithms Lab manual, Department of CSE, AI&ML, AI&DS and IT, GMRIT
2. https://www.leetcode.com

23CS308 JAVA Lab

0 0 3 1.5

Course Outcomes

1. Make use of JAVA SDK environment to create - debug and run java programs
2. Create applications based on code reusability
3. Develop programs using threads
4. Develop and debug real time problems using exception handling
5. Design Graphical User Interface using AWT components
6. Design Graphical User Interface using Swing components

CO-PO Mapping

CO's	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	1	1	-	-	-	-	-	2	-	2	-	2
CO2	3	3	2	1	-	-	-	-	-	2	-	2	-	2
CO3	3	3	2	2	-	-	-	-	-	2	-	2	-	2
CO4	3	3	2	3	-	-	-	-	-	2	-	3	-	3
CO5	3	3	3	2	-	-	-	-	-	2	-	2	-	3
CO6	3	3	3	2	-	-	-	-	-	2	-	3	-	3

3–Strongly linked | 2–Moderately linked| 1–Weakly linked

List of Experiments

1. Case Study: Permutations and Combinations Calculator

User Input:

- The program should prompt the user to enter the total number of elements (n) and the number of elements to be selected (r).
- Validate the input to ensure that both n and r are positive integers, and r is less than or equal to n.

Functional Description:

- Create a method **calculatePermutations** that takes two integers n and r as input and returns the number of permutations.

Formula *nPr = n! / (n - r)!, where "!" denotes factorial*.

- Create a method **calculateCombinations** that takes two integers n and r as input and returns the number of combinations.

Formula *nCr = n! / (r! * (n - r)!).*

- Implement a separate method to calculate the factorial of a given integer.

Sample Output:

Enter the total number of elements (n): 5
Enter the number of elements to be selected (r): 3
Permutations (nPr): 60
Combinations (nCr): 10

2. Conversion Utility that allows users to convert measurements between different units.

User Input:

- The program should prompt the user to enter positive integer value for conversion type.
- The program should prompt the user to enter the positive input (feet/ meters/ miles/ kilometers) based on user option.

Functional Description:

- utilize method overloading to handle conversions between various units, such as feet to meters, miles to kilometers, and vice versa.

formula:feet to meters - feet * 0.3048
miles to kilometers - miles * 1.60934

Sample Output:

Measurement Conversion Utility
1. Feet to Meters
2. Meters to Feet
3. Miles to Kilometers
4. Kilometers to Miles
5. Exit
Enter your choice: 3
Enter the distance in miles: 5
5 miles is approximately 8.046 kilometers.

3. Case Study: **BankAccount** Class with Constructor Overloading

Functional Description:

- Create a **BankAccount** class to represent a bank account. The class should have private attributes such as **accountNumber** (a unique identifier for the account), **accountHolderName**, and **balance** (representing the current account balance).
- The class should have the following constructors:
  - **BankAccount(String accountNumber, String accountHolderName)**: Create an account with a zero balance.
  - **BankAccount(String accountNumber, String accountHolderName, double initialBalance)**: Create an account with the specified initial balance.
- Implement methods to:
  - **deposit(double amount):** Add the given amount to the account balance.
  - **withdraw(double amount):** Subtract the given amount from the account balance if sufficient funds are available.

Sample Output:

Welcome to the Bank Account System
Creating a new account...
Enter the account holder's name: John Doe
Enter the initial balance: 500.0
Account created successfully.
Account Number: ACCT123456
Account Holder: John Doe
Balance: $500.00
What would you like to do?
1. Deposit
2. Withdraw
3. Exit
Enter your choice: 1
Enter the amount to deposit: 250.0
$250.00 deposited successfully.
Updated Balance: $750.00

4. Implement inheritance concept for below case study:

Note: class **FictionBook** extends class **LibraryBook**, class **ClassicBook** and class **ScienceFictionBook** extends from **FictionBook**

- **LibraryBook**: Represents a general library book with attributes like title, author, and year, along with a method **displayBookInfo()** to display book details.
- **FictionBook**: Extends **LibraryBook** and adds an additional attribute genre.
- **ClassicBook**: Extends **FictionBook** and adds an additional attribute publisher.
- **ScienceFictionBook**: Extends **FictionBook** and adds an additional attribute edition

5. Case Study: Abstract Class for Finding Areas of Different Shapes

User Input:

- The program should prompt the user to enter positive integer value to calculate type of area.
- The program should prompt the user to enter the input based on user option.
  - **Circle**: Requires the radius as an input parameter
  - **Rectangle**: Requires the length and width as input parameters.
  - **Triangle**: Requires the base and height as input parameters.
  - **Square**: Requires the side length as an input parameter.

Functional Description:

- Create an abstract class **Shape** that will serve as the base class for different shapes. Include an abstract method **calculateArea()**
- Create concrete subclasses that extend the **Shape** class for different shapes like **Circle**, **Rectangle**, **Triangle** and **Square**.

Sample Output:

Select a shape:
1. Circle
2. Rectangle
3. Triangle
4. Square
5. Exit
Enter your choice: 1
Enter the radius of the circle: 5
The area of the selected shape is: 78.53981633974483

6. Implement Dynamic Method Dispatch in Java.
7. i. Create an **ArrayList** of integers to store a collection of numbers. Implement methods to add elements to the list, remove elements, update elements, and search for elements in the list.
   ii. Create a Java program that accepts a string input from the user. Use the **StringTokenizer** class to tokenize the input string into separate tokens based on specified delimiters (e.g., space, comma, etc.).
8. Demonstrating Mutability Differences between **String**, **StringBuilder**, and **StringBuffer** in Java
   - Explain the Concept in 100 words.
   - Perform string manipulation operation concatenation on each class
   - Print the results of each class after modification operation
9. Demonstrate multiple inheritance-like behavior achieved through interface implementation.
10. Case Study: Java Program Illustrating Multiple Catch Clauses with Finally Block

    Functional Description:
    - Implement exception handling for possible exceptions that may occur during the division, such as **ArithmeticException** and **NumberFormatException**.
    - Use multiple catch clauses to handle **ArithmeticException** and **NumberFormatException** separately.
    - Use a **finally** block to execute code that should be performed regardless of whether an exception occurred or not

    Sample Output:

    Enter the dividend: 10
    Enter the divisor: 0
    Division by zero is not allowed.
    Execution completed.
11. Create your own custom exception classes to handle specific situations or errors that are not covered by the built-in exception classes.

    Functional Description:
    - Create a custom exception class that extends the **Exception** class (or any other appropriate superclass, depending on your requirements).
    - Define the custom exception class to represent a specific error situation, such as **InvalidAgeException**, which is thrown when an invalid age is provided.
    - Use the custom exception class in a method to throw the exception when a certain condition is met, such as when the age is less than zero or exceeds a specific limit.

    Constraints:

    The program should prompt the user to enter age integer value. Here, age<0 or age>120

    Sample Output:

    Enter age:-5
    Exception: Invalid age. Age must be between 0 and 120.
12. Creates three threads by extending the **Thread** class. Each thread will display a specific message at a regular interval. The messages to be displayed are as follows:

    First thread: "**Good Morning**" every 1 second
    Second thread: "**Hello**" every 2 seconds
    Third thread: "**Welcome**" every 3 seconds
    Note: implement the same by using **Runnable** interface.
13. Case Study: Java Program for Producer-Consumer Problem

    User Input:
    - The program should prompt the user to enter positive integer valuesfor **bufferSize**
    - The program should prompt the user to enter positive integer values to **produceCount** and **consumeCount**

    Functional Description:
    - Buffer Implementation:
      - Create a shared buffer with a limited size (e.g., an array or an ArrayList) to hold the produced items.
    - Producer & Consumer Implementation:
      - Create a Producer class & Consumer class that implements the Runnable interface.
      - The Producer class should produce items and put them into the buffer.
      - The Consumer class should consume items from the buffer.
      - Implement synchronization to avoid race conditions when accessing the buffer.

    Sample Output:

    Enter Buffer Size:5
    Enter Procedure Count:5
    Enter Consumer Count:5
    Produced: 0

Consumed: 0
Produced: 1
Consumed: 1
Produced: 2
Consumed: 2
Produced: 3
Consumed: 3
Produced: 4
Consumed: 4

14. Case Study: Java Program for a Basic Calculator using AWT

In this case study, we'll implement a Java program to create a basic calculator using AWT. The calculator will have a simple graphical user interface (GUI) with buttons for numbers, arithmetic operations, and a display to show the result of calculations.

AWT Implementation:

- Create a Java AWT to implement the calculator GUI.
- The AWT should have buttons for digits (0 to 9), arithmetic operations (addition, subtraction, multiplication, division), and an equal (=) button to perform calculations.

Display and Input:

- Implement a display area where the user can see the entered numbers and the result of calculations.
- Provide a way for users to input numbers and arithmetic operations using the GUI buttons.

Basic Calculations:

- Implement methods to handle basic arithmetic calculations (addition, subtraction, multiplication, division).
- Perform calculations based on the user's input and display the result in the GUI.

User Interface:

- Design a user-friendly GUI with appropriate labels, buttons, and display areas.
- Use layout managers to organize the components effectively.

15. Case Study: Java Program for Temperature Converter using AWT

In this case study, we'll implement a Java AWT to create a Temperature Converter. The AWT will allow users to convert temperatures between Celsius and Fahrenheit. The user will input a temperature value in one unit, and the AWT will display the converted temperature in the other unit.

AWT Implementation:

- Create a Java AWTpplet to implement the temperature converter GUI.
- The AWT should have input fields for the temperature value and buttons to select the source and target temperature units (Celsius or Fahrenheit).
- Use appropriate layout managers to organize the components effectively.

Conversion Logic:

- Implement methods to handle temperature conversions between Celsius and Fahrenheit.
- The temperature conversion formulas are:
  - Fahrenheit to Celsius: C = (F - 32) * 5 / 9
  - Celsius to Fahrenheit: F = (C * 9 / 5) + 32

User Interface:

- Design a user-friendly GUI with appropriate labels, input fields, and buttons.
- Display the converted temperature in the GUI after the conversion is performed.

16. Case Study: Java Swing Application for Job Application Form

In this case study, we'll implement a Java Swing application with a JFrame that contains fields for name, age, email, and qualifications. The application will validate the form data when the user clicks the "Submit" button. If all fields are filled, the data will be written to a file named "application_form.txt."

Swing Application:

- Create a Java Swing application with a JFrame to display the job application form.
- Design the form with labels for each field (name, age, email, qualifications) and corresponding input fields (text fields or other appropriate components).
- Add a "Submit" button to the form to allow users to submit their job application.

Data Validation:

- Implement validation for the form data to ensure that all fields are filled before submission.
- Display error messages for missing fields or invalid data.

Data Writing:

- Create a method to write the validated form data to a file named "application_form.txt."
- The file should be created in the current working directory.
- Use appropriate file handling techniques to write the data.

User Interaction:

- Allow users to input their name, age, email, and qualifications using the form fields.

- Trigger the form data validation when the user clicks the "Submit" button.
- Display appropriate messages to inform the user about the success or failure of form submission.

List of Augmented Experiments

1. New Patient Registry Management System
2. Restaurant Billing Management System
3. Library Management System
4. ATM Management System
5. Bus Ticket Booking Management System
6. Movie Ticket Booking Management System
7. Queuing Management System
8. Attendance Management System
9. Medical Store Billing Management System
10. Text Editor Projects in Java
11. Google Search Engine Filter
12. Electronic voting System
13. Day Planner
14. Library management System
15. Personal Finance Management System

Reading Material (s)

JAVA Lab manual, Department of CSE, AI&ML, AI&DS and IT, GMRIT, Rajam

23ESX01 Employability Skills I

0 0 2 0

Course Outcomes

1. Demonstrate oral communication and writing skills as an individual to present ideas coherently
2. Develop life skills with behavioral etiquettes and personal grooming.
3. Assess analytical and aptitude skills.
4. Develop algorithms for engineering applications.
5. Solve engineering problems using software.
6. Utilize simulation tools for testing

COs –POs Mapping

CO's	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	2	1	2	3	2	-	-	-	2	2	2	3	-	-
CO2	3	2	3	2	3	-	-	-	2	2	2	3	-	-
CO3	3	2	3	2	3	-	-	-	2	2	2	3	-	-
CO4	3	2	3	2	2	-	-	-	3	3	3	3	-	-
CO5	3	2	3	2	2	-	-	-	3	3	3	3	-	-
CO6	2	1	2	3	2	-	-	-	3	3	3	3	-	-

3–Strongly linked | 2–Moderately linked| 1–Weakly linked

Unit-I

Communication Skills **8 hours**

Communication Skills & Confidence: How Communication Skills affect Confidence? How to communicate effectively.(with Examples)

Listening: Listening? , Listening Vs Hearing, Possible reasons for why people do not Listen at times, Active Listening Vs Passive Listening, How Listening can affect our relationships? How Listening helps in Campus Placements also? (with Examples)

Goal Setting: SMART Technique to Goal Setting, Putting First things First, SWOT Analysis and Time Management

Attitude & Gratitude: Attitude Vs Skills Vs Knowledge, Attitude Vs Behaviour, How to develop Positive Attitude? Developing the attitude of Gratitude.

Public Speaking: JAM, J2M, Presentations by Students on General Topics.

Quantitative Aptitude **8 hours**

Number system, L.C.M and H.C.F, Problems on Ages, Averages, Time and work, Pipes and cisterns

Unit-II

Competitive Programming Phase – I **16 Hours**

Module-1:

Introduction: Importance of Competitive Programming, Conditional Statements, Loops, Nested Loops. Strings: Declaration of strings, Accessing strings elements, operations**,** String Matching Algorithms, Practice Problems - Codechef, Hackerrank, Codeforces, Leetcode, interviewbit.

Module-2:

Arrays: Basics on Arrays: Creating elements, converting a string into array of characters, converting a number into array of character, accessing array elements, 2D Arrays & Matrix, **Recursion:** Definition, examples and how recursion differs from iteration. Practice Problems - Codechef, Hackerrank, Codeforces, Leetcode, interviewbit.

References:

1. https://leetcode.com
2. https://www.codechef.com
3. https://www.hackerearth.com
4. https://www.interviewbit.com
5. https://www.hackerrank.com

4th Semester

23IT304 Database Management Systems

3 0 0 3

Course Outcomes

At the end of the course, students will be able to

1. Explain the fundamental concepts of database and Various data models
2. Understand the Relational Algebra and integrity constraints for Query execution
3. Apply the relational database to formulate the SQL Commands
4. Make use of Normalization to improve database design and schema refinement
5. Analyze the concept of transaction management, recovery and serializability in databases
6. Classify the concurrency control protocols, different types of data and Indexing techniques

CO–PO Mapping

COs	PO_1	PO_2	PO_3	PO_4	PO_{12}	PSO_2
1	3	3	2	2	1	1
2	3	3	3	2	2	2
3	3	3	2	2	2	2
4	3	3	3	2	1	1
5	3	3	3	3	2	2
6	3	3	2	2	2	2

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I

Introduction to DBMS and ER Model **11 Hours**

DBMS Vs. File System, instance and schema, Data abstraction, Data independence, database users and database administrator, Database system structure, Introduction to Data Models (E-R Model, Relational Model, Hierarchical Model, Network Model, Object Oriented Data Model), Database Design Process, Entities, Attributes, Entity Sets, Relationships, Relationship Sets, Additional features of ER Model.

Applications of DBMS, Object Relational Data Model

Unit II

Introduction to Relational Model and Basic SQL Queries **12 Hours**

Relational Algebra Operations: Selection, Projection, Rename, Set Operators, Joins, Division, Examples of Relational Algebra Queries, Relational Calculus: Tuple Relational Calculus. Integrity Constraints over Relations, Introduction to Views. SQL Queries: Basic Structure, Set Operations, Aggregate Functions, Null values, Sub Queries, Group By And Having Clauses, Outer Joins.

Domain Relational Calculus, Query Optimization

Unit III

Normalization and Transaction Management **13 Hours**

Introduction To Schema Refinement - Problems Caused By Redundancy - Decomposition - Problems Related To Decomposition - Functional Dependency - Closure of a Set of Fds - Attribute Closure - First - Second - Third Normal Forms – BCNF - Multi Valued Dependencies – Fourth Normal Form, Join Dependency, Fifth Normal Form, Transactions: Acid Properties of Transaction - Transaction States - Schedule: Serial Schedule - Concurrent Schedules - Anomalies Associated with Concurrent Schedules (RW - WR - and WW Conflicts) -Serializability - Conflict Serializability - and View Serializability.

EF Codd Rules, Domain Dependency

Unit IV

Locking, Recovery Systems, Indexing, Different Types of Data **12 Hours**

Introduction to Lock Management-Lock Based Concurrency Control: 2pl-Strict 2pl-Concurrency without Locking: Timestamp–Based Concurrency Control, Optimistic Concurrency Control.

Introduction to Aries - the Log - the Write-Ahead Log Protocol-Check Pointing Indexing: Types of Single-Level Ordered Indexes, Multilevel Indexes Different Types of Data: Structured, Semi-Structured and Unstructured Data

Heap File, Hash File Organizations

Total: 48 Hours

Textbook (s)

1. Elmasri & Navatha, Fundamentals of Database Systems, Pearson Education, 7th Edition, 2016
2. Silberschatz Korth, Database System Concepts, McGraw hill, 7th Edition, 2019

Reference (s)

1. Soraya Sedkaoui, Data Analytics and Big Data, Wiley, 1st Edition, 2018.
2. Peter Rob & Carlos Coronel, Database Systems design, Implementation and Management, 9th Edition, 2010.
3. Raghurama Krishnan & Johannes Gehrke, Database Management Systems, TATA McGraw-Hill, 3rd Edition, 2003
4. C.J.Date, An Introduction to Database Systems, Pearson Education, 8th Edition, 2006

Internal Assessment Pattern

Cognitive Level	Int. Test 1 (%)	Int. Test 2 (%)	Open Book Exam (%)
Remember	25	20	--
Understand	30	30	--
Apply	30	20	50
Analyze	15	30	50
Evaluate	--	--	--
Create	--	--	--
Total (%)	100	100	100

Sample Question (S)

Remember

1. List any four application of DBMS
2. Define data model
3. List any four applications for triggers
4. Define functional dependency
5. List the 4 properties of Transaction

Understand

1. Explain E-R Model with suitable example
2. Explain the role of integrity constraints in database design
3. Illustrate the working principle of 'write a head log' protocol
4. Differentiate 3NF and 4NF
5. Explain Two Phase Locking Protocol

Apply

1. When multiple transactions are being executed by the operating system in a multiprogramming environment, there are possibilities that instructions of one transaction are interleaved with some other transaction. Apply the suitable concept to overcome the problem
2. Classify various normal forms according to their applicability
3. Give some real-world applications of Normalization
4. Illustrate the Commit and Rollback operations of Transaction Control
5. Give some real-world applications for Database indexing techniques

Analyze

1. Compare File processing system with DBMS
2. Analyze different locking protocol for concurrency control and serializability
3. Normalization will increase the complexity of the database design. Justify
4. Compare DDL and DML of SQL
5. Compare and Contrast Serializability and Recoverability

Evaluate

1. Is database redesign being necessary? explain
2. How can you evaluate the performance of two data models?
3. Evaluate the performance of query processor and list the corresponding metrics
4. How can you assess the throughput and delay for any DBMS?
5. How can you evaluate the impact of data models on the query processing?

Open Book Examination Questions

1. Anitha has a large CD collection. Her friends like to borrow her CD's, and she has to keep track of who has what. She maintains a list of friends, identified by unique FID's and a list of CD's, identified by CID's. With each friend are the name and telephone numbers which she can call to get the CD back. With each CD is actor name and title. Whenever a friend borrows a CD, She will enter that fact into her database along with the date borrowed. Whenever the CD gets returned, that fact, too, gets noted along with the date returned. Anitha wants to keep a complete history of her friends' borrowing habits so that she can ask favors of the heavy borrowers.
   Draw an ER diagram to figure out the above situation and identify types of attributes and cardinality. Represent this database as a collection of 3NF relational tables.

The relational scheme R(A,B,C,D,E,F) and set of functional dependencies AB -> D, E -> C, AF -> B. From this, find out all super keys for this relation, and which of these super keys form a key.

23IT403 Operating Systems

3 0 0 3

Course Outcomes

At the end of the course, students will be able to

1. Understand the different services provided by operating system at different level.
2. Analyze various CPU Scheduling Algorithms and synchronization techniques.
3. Identify deadlock prevention and avoidance measures.
4. Compare and contrast main memory and virtual memory.
5. Illustrate the functionality of file systems.
6. Evaluation of various disk scheduling algorithms.

CO – PO Mapping

COs	PO_1	PO_2	PO3	PO4	PO_7	PO_{12}
1	3	3	2	2	2	2
2	3	3	2	2	2	2
3	3	3	3	2	2	2
4	3	3	3	2	2	2
5	3	3	2	2	2	2
6	3	2	2	2	2	2

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Syllabus

Unit I **12Hours**

Operating-Systems Overview and Process Management

Operating-System Overview: Computer-System Organization and Architecture, Operating-System Structure, Operating-System Operations & Services, System Calls & its types. Threads: Multi Core Programming, Multithreading Models, Thread Scheduling algorithms. Process Management: Process Concepts, Process Scheduling Criteria, Scheduling Algorithms and evaluation.

Thread issues, Multilevel Queue, Multilevel feedback Queue Scheduling

Unit II **12 Hours**

Inter Process Communication Mechanism

Process Synchronization: Cooperative process, the Critical Section Problem, Peterson's Solution, Synchronization Hardware, Semaphores, Classical Synchronization problems, Monitors.

Deadlocks: Deadlock Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance & Detection, Recovery from Deadlock.

Synchronization Examples-Synchronization in Solaris, Synchronization in Linux.

Unit III **12 Hours**

Memory Management

Main Memory: Contiguous Memory allocation, Swapping, Segmentation, Paging, Segmented paging, Multilevel paging.

Virtual Memory Management: Demand Paging, Page Replacement algorithms, Allocation of Frames.

Structure of page table, Thrashing, Memory-Mapped Files

Unit IV **12 Hours**

File System Interface & I/O Systems

Mass-Storage Structure: Disk structure, Disk Scheduling, Disk management, Raid Structure.

File System: Access Methods, Directory Structures, Allocation Methods, Free-Space Management.

I/O Systems: I/O hardware Application of I/O Interface, Kernel I/O Sub-System.

File Sharing, File System Recovery

Total: 48 Hours

Textbook (s)

1. Operating System Concepts, Abraham Silberschatz, Greg Gagne, Peter B. Galvin, 9th Edition, Wiley, 2016.
2. Operating Systems, Harvey M. Deitel, Paul J. Deitel, David R. Choffnes, 3rd Edition, Pearson Prentice Hall, 2004.

Reference (s)

1. Operating Systems: Internals and Design Principles, William Stallings, 7th Edition, Pearson Prentice Hall, 2013.
2. Operating systems: A Concept based Approach, D. M. Dhamdhere, 2nd Edition, TMH, 2006.
3. Operating System: A Design Approach, Crowley, 1st Edition, TMH, 2001.
4. Modern Operating Systems, Andrew S Tanenbaum, 3rd Edition, PHI, 2009.

Internal Assessment Pattern

Cognitive Level	Int. Test 1 (%)	Int. Test 2 (%)	Open Book Exam (%)
Remember	20	20	--
Understand	30	10	--
Apply	30	30	40
Analyze	10	20	30
Evaluate	10	20	30
Create	--	--	--
Total (%)	100	100	100

Sample Question (S)

Remember

1. List any four operating systems
2. Define operating system
3. List four operating system services

Understand

1. Explain System calls
2. Explain the role memory management in operating system
3. Illustrate the working principle critical section problem

Apply

1. When multiple transactions are being executed by the operating system in a multiprogramming environment, there are possibilities that instructions of one transaction are interleaved with some other transaction. Apply the suitable concept to overcome the problem
2. Give an example of a scenario that might benefit from a file system supporting an append-only access write.

Analyze

1. Context switching between two threads of execution within the operating system is usually performed by a small assembly language function. In general terms, what does this small function do internally?
2. Compare CPU scheduling algorithms
3. Analyze the general strategy behind deadlock prevention and give an example of a practical deadlock prevention method.

Evaluate

1. Assuming the operating system detects the system, is deadlocked, what can the operating system do to recover from deadlock?
2. Describe how to implement a lock using semaphores.

Open Book Exam Questions

1. A file to be shared among different processes, each of which has a unique number. The file can be accessed simultaneously by several processes, subject to the following constraint: the sum of all unique numbers associated with all processes currently accessing the file must be less than n. Write a monitor to co-ordinate the access to the file. Ou may want to write start access and end access monitor procedures.
2. Consider a paging system with the page table stored in memory:
    a. If a memory reference takes 200 nanoseconds, how long does a paged memory reference take?
    b. If we add associative registers, and 75% of all page-table references are found in the associative registers, what is the effective memory reference time?

23CS403 Computer Organization and Architecture

3 0 0 3

Course Outcomes

1. Interpret the functional architecture of computing systems.
2. Summarize the types of instruction and its micro operation with addressing modes
3. Identify various arithmetic operations on fixed, floating point numbers and its representation.
4. Illustrate the concepts of control unit design and I/O processor.
5. Discuss the memory hierarchy concepts.
6. Describe concept of parallelism and types of hazard.

CO-PO Mapping

CO's	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	2	-	-	-	-	-	-	-	-	1	2	-
CO2	3	2	2	-	-	-	-	-	-	-	-	1	2	-
CO3	3	3	2	-	-	-	-	-	-	-	-	1	2	-
CO4	3	2	2	-	-	-	-	-	-	-	-	1	2	-
CO5	3	2	2	-	-	-	-	-	-	-	-	1	2	-
CO6	3	2	2	-	-	-	-	-	-	-	-	1	3	-

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I **13 Hours**

Overview & Micro operation

Components of a computer system – Performance measures - Classifying Instruction Set Architecture- Representing instructions -Micro operation – Logical operations – Shift operations - instruction codes - Computer Registers instruction –memory Reference instruction –Input-Output Reference instruction - Instruction cycle -Addressing and addressing modes.

Trends in Technology-Arithmetic micro-operations

Unit II **11 Hours**

Arithmetic Operations

ALU - Addition and subtraction with Signed Magnitude Data - Hardware Implementation – Multiplication – Hardware Implementation for Signed Magnitude Data – Division - Hardware Implementation for Signed Magnitude Data – Divide Overflow - Floating Point operations – Parallelism and Computer Arithmetic: Sub word Parallelism.

BCD Adder-BCD Subtraction

UNIT III **11 Hours**

Control Unit and Memory Systems

Basic MIPS implementation – Building data path – Control Implementation scheme – Memory hierarchy – Cache basics – Measuring and improving cache performance - Virtual memory- Input/output system- programmed I/O-DMA and Interrupts-I/O processors

Stack organization-RISC Vs CISC Architecture

Unit IV **13 Hours**

Parallelism

Instruction-level-parallelism – Parallel processing challenges – Flynn's classification – Multicore processors- Pipelining – Arithmetic pipeline –Instruction pipeline -Pipelined data path and control – Handling Data hazards & Control hazards – Exceptions.

Vector processing –single processor Vs parallel processor

Total: 48 Hours

Text Book(s):

1. DavidA.PattersonandJohnL.Hennessey,"Computerorganizationanddesign:Thehardware/software interface", Morgan Kauffman / Elsevier, Fifth edition, 2014.

2. M.Morris Mano," Computer System Architecture", 3rd edition, Pearson/PHI,2092.

Reference(s):

1. V.CarlHamacher, Zvonko G. Varanesic and Safat G. Zaky, "Computer Organization ", 6th edition, McGraw-Hill Inc,2012.
2. William Stallings "Computer Organization and Architecture, Seventh Edition, Pearson Education, 2007.
3. Andrew S Tanenbaum "Structured Computer Organization ", 5th edition, Pearson/PHI,2007

Internal Assessment Pattern

Cognitive Level	Int. Test 1 (%)	Int. Test 2 (%)	Assignment Test/OBE (%)
Remember	40	--	--
Understand	40	50	--
Apply	20	50	80
Analyze	--	--	20
Evaluate	--	--	--
Create	--	--	--
Total (%)	100	100	100

SAMPLE QUESTION (S)

Remember

1. What is micro operation?
2. Show the Register Reference Instruction format.
3. Define PC and MAR.
4. What are the two types of data presentation?
5. Define Associative Memory

Understand

1. Identify the basic functional units of the system
2. Explain about logic micro operations and its applications with examples
3. Differentiate RISC and CISC architectures in terms of their instruction set and addressing modes.
4. Compare hardwired control unit is differing from micro programmed control unit designs
5. Demonstrate control memory

Apply

1. Starting from an initial value of R=11011101, determine the sequence of binary values in R after a logical shift-left, followed by a logical shift-right and a circular shift-right.
2. Analyze the contents of Register A that holds 8 bit binary 11011001 and Determine the B- operand and the logic micro operation to be performed in order to change the value in A to: (i) 01101101 (ii) 11111101State the differences between register stack and memory stack.
3. Perform the arithmetic operations (+70) + (+80) and (-70) + (-80) with binary numbers in signed-2's complement representation. Use eight bits to accommodate each number together with its sign. Show that overflow occurs in both cases, that the last two carries are unequal, and that there is a sign reversal.
4. Show the hardware to be used for the addition and subtraction of two decimal numbers with negative numbers in signed- 10's complement representation. Indicate how an overflow is detected. Derive the flowchart algorithm and try a few numbers to convince yourself that the algorithm produces correct results.
5. The procedure for aligning mantissas during addition or subtraction of floating-point numbers can be stated as follows: Subtract the smaller exponent from the larger and shift right the mantissa having the smaller exponent a number of places equal to the difference between the exponents. The exponent of the sum (or difference) is equal to the larger exponents. Without using a magnitude comparator, assuming biased exponents, and taking into account that only the AC can be shifted, derive an algorithm in flowchart form for aligning the mantissas and placing the larger exponent in the AC.

Open book Examination

Apply

1. Smith and Goodman found that for a given small size, a direct-mapped instruction cache consistently outperformed a fully associative instruction cache using LRU replacement.
    a. Explain how this would be possible (*Hint:* You can't explain this with the three C's model because it "ignores" replacementpolicy)
    b. Explain where replacement policy fits into the three C's model, and explain why this means that misses caused by a replacement policy are "ignored"- or, more precisely, cannot in general be definitively classified by the three C's model.
    c. Are there any replacement polices for the fully associative cache that would outperform the direct-mapped cache? Ignore the policy of "do what a direct- mapped cache would do".
    d. Use a cache simulator to see if Smith and Goodman's results hold for memory reference traces that

you have access to. If they do not hold, why not?

2. John takes two numbers in sign magnitude representation (the two numbers are same with different signs), The 1's complement of one number is 6. The difference between 1's complement of these two numbers is 32. Find the numbers. And also find the product of these two numbers using the result of 2's complement value of these two numbers.

Analyze

1. A two-word instruction is stored in memory at an address designated by the symbol W. The address field of the instruction (stored at W + 1) is designated by the symbol Y. The operand used during the execution of the instruction is stored at an address symbolized by Z. An index register contains the value X. State how Z is calculated from the other addresses if the addressing mode of the instruction is
    a. direct
    b. indirect
    c. relative
    d. indexed.
2. An 8-bit computer has a 16-bit address bus. The first 15 lines of the address are used to select a bank of 32K bytes of memory. The high-order bit of the address is used to select a register which receives the contents of the data bus. Explain how this configuration can be used to extend the memory capacity of the system to eight banks of 32K bytes each, for a total of 256K bytes of memory.

23MA404 - Probability and Statistics using Python

3 0 2 4

Course Outcomes

1. List the basic data structures of Python and various libraries of Python.
2. Illustrate basic probability axioms and apply Bayes' theorem related to engineering problems.
3. Differentiate between various distributions of random variables on given data.
4. Examine given dataset based on descriptive statistics.
5. Compare the null or alternative hypotheses using the suitable test statistic.
6. Choose appropriate curve fitting and correlation methods for a given data

CO-PO Mapping

CO's	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	2	3	-	1	-	-	-	-	-	2	-	2	-	-
CO2	3	2	-	3	-	-	-	-	-	3	-	2	-	-
CO3	3	2	-	3	-	-	-	-	-	3	-	1	-	-
CO4	3	1	-	3	-	-	-	-	-	3	-	3	-	-
CO5	3	1	-	2	-	-	-	-	-	3	-	1	-	-
CO6	3	1	-	3	-	-	-	-	-	3	-	3	-	-

3-Strongly linked | 2-Moderately linked | 1-Weaklylinked

Unit I **12+8 Hours**

Fundamental of Python Programming

Strings, Files, Lists, Dictionaries, Tuples**. Introduction to NumPy:** Creating Numpy Arrays, Numpy Data Objects, dtype, Numerical Operations. Numpy Arrays: Concatenating, Flatten, reshape, adding Dimensions, Vector Stacking,Pandas DataFrame, Accessing and Changing values of DataFrames, Pandas Pivot, groupby **Introduction to Pandas:** Reading and Writing Data in Pandas, Multi-level Indexing, Data Visualization in Pandas

Objects and Classes

Practical Components

1. Experiments on Basic experiments on fundamentals on python
2. Exploratory Data Analysis using NumPy
3. Exploratory Data Analysis using Pandas
4. Exploratory Data Analysis using Matplotlib

Unit II 12+6 **Hours**

Random Variables and Probability Distributions

Introduction to Probability: Events, Axioms of Probability and Probabilities of events; Conditional Probability, Bayes Theorem. Discrete and Continuous random variables; Expectation and Variance, Distribution of a Random Variables: **Discrete and Continuous Distributions**- Bernoulli, Binomial, Poison and Normal Distributions (without proofs); Central Limit Theorem (without proof).

Gaussian distribution

Practical Components

1. Simulation of random experiments and plot the empirical probability of each event, alongside the theoretical probability.
2. Python script to find conditional probability of a given event.
3. Simulation of various probability distributions

Unit III **12+10 Hours**

Sampling Theory and Testing of Hypothesis

Sampling: Population and sample, parameter and statistic, standard error. Parameter estimations, Confidence Intervals, Unknown Standard Deviation. **Hypothesis Testing**: Null and alternative hypothesis, level of significance, Type I and Type II errors, one tail and two-tail tests. Hypothesis testing concerning one mean and proportions, two means – Proportions and their differences using Z-test, Student's t-test, F-test, Chi-square test.

One-way ANOVA

Practical Components

1. Python Modules for Descriptive Statistics
2. Write a python program to perform z-test, t- test for testing the Null hypothesis for single mean
3. Write a python program to perform z-test, t- test for testing the Null hypothesis for difference of means
4. Write a python program to explore problems in F-test.
5. Write a python program to perform chi square for determine whether there is a significant association between the two variables

Unit IV **12+8 Hours**

Curve Fitting, Correlation & Regression

Curve Fitting: Introduction-fitting a straight line-second degree curve-exponential curve by method of least squares. **Correlation:** Simple correlation, Karl Pearson's Correlation coefficient and Spearman's rank correlation. Simple regression, multiple regression.

Covariance

Practical Components

1. Develop a straight line curve to fit some real world observations
2. Develop a exponential curve to fit some real world observations
3. Explore problems in Correlation with sample data
4. Explore problems in Regression with sample data

Total: 48+32 **Hours**

Text Books:

1. Probability & Statistics by T.K.V.Iyengar&B.Krishna Gandhi & Others, S.Chand (2016) 6th edition.
2. E. Walpole, R. H. Mayers, S. L. Mayers and K. Ye, (2012), Probability and Statistics for Engineers and Scientists,9th Edition, Pearson Education, ISBN: 978-8-131-71552-9.
3. Wes McKinney, Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython, 2nd Edition, O'Reilly Media, Inc.

Reference Books:

1. Michael Baron, Probability and Statistics For Computer Scientists, 2nd Edition, CRC Press
2. Sheldon M. Ross, (2011), Introduction to Probability and Statistics for Engineers and Scientists, 4th Edition, Academic Foundation, ISBN: 978-8-190-93568-5.

Internal Assessment Pattern

Cognitive Level	Int. Test 1 (%)	Int. Test 2 (%)	Assignment Test/OBE (%)
Remember	10	10	--
Understand	50	30	--
Apply	40	60	--
Analyze	--	--	--
Evaluate	--	--	--
Create	--	--	--
Total (%)	100	100	--

SAMPLE QUESTION (S)

Remember

1. Define variable in python.
2. List any two differences between built-in and user defined functions.
3. List any two python packages used mathematics.

Understand

1. Explain use of probability and statistics in real world.
2. Illustrate about random variables in python.

Apply

1. Find different real time applications which use Markov process
2. Implement a queuing theory to handle multiple processes.

23ML405 Foundations of Machine Learning

3 0 0 3

Course Outcomes

1. Explore the pipeline of machine learning to apply on real world problems
2. Demonstrate the need of Similarity and Dissimilarity between features in data
3. Apply Pre-processing techniques in real-time
4. Analyze the characteristics of various Regression models in real-world
5. Examine the characteristics of various supervised learning and ensemble models
6. Investigate the characteristics of various algorithms of unsupervised learning

COs–POs Mapping

CO's	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	-	1	-	-	-	-	-	-	-	-	1	1
CO2	3	2	-	3	-	-	-	-	-	-	-	-	2	2
CO3	2	2	-	3	-	-	-	-	-	-	-	-	2	2
CO4	2	3	-	3	-	-	-	-	-	-	-	-	3	2
CO5	2	3	-	2	-	-	-	-	-	-	-	-	3	2
CO6	3	3	-	2	-	-	-	-	-	-	-	-	3	2

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

UNIT I **12 hours**

Introduction: Relation among AI, ML and Data Science, Importance of Machine learning, Types of machine learning ,ML Use cases; Phases of ML: Training, Testing and Validation Machine learning Pipeline; Evaluation Machine Learning Models: Error Analysis, Performance Indices

Data Exploration: Objectives of Data Exploration, Forms of Data (Structured, Semi Structured, Unstructured), Datasets (data objects and types of attributes/fields), Characteristics of Datasets and corresponding Statistical Measures.

Discriminant analysis, LDA

UNIT II **12 hours**

Data Similarity/Dissimilarity: Data Matrix, Dissimilarity Matrix, Proximity/Similarity Measures for various types of data (nominal, ordinal, binary, numerical, etc.), Dissimilarity for Attributes of Mixed Types, Cosine Similarity;

Data Preprocessing: Preprocessing Pipeline, Forms of Preprocessing, Data Cleaning, Data Integration, Data Reduction, Data Transformation and Discretization.

Distance based similarity (Euclidean distance, Jaccard Similarity)

UNIT III **12 hours**

Regression (forecast a given numerical quantity): Linear Regression, Logistic Regression (Regression for Classification), Regularization;

Classification: Classification Principles, Classification Model Evaluation Metrics (Confusion Matrix, ROC and AUC, Lift Covers), Classification using Decision Trees, Bayesian Classification, Distance based Classifier(k-NN), Support Vector Machines (SVM), ANN-based Classification

LASSO and Ridge Regression

UNIT IV **12 hours**

Ensemble Learning: Conditions for Ensemble Modeling, Ensemble Techniques (Voting, Bagging, Boosting and Random Forest);

Clustering: Clustering Principles, Applications/Purpose of Clustering- Clustering for description/preprocessing/classification, Types of Clustering, Clustering Evaluation Parameters, Clustering Algorithms (k-Means, Agglomerative/Divisive, DBSCAN and Self Organizing Maps) and Evaluation Metrics; **Data Science Tools: Examples.**

ADABoost, XGBoost

Textbooks:

1. Stephen Marsland, "Machine Learning -An Algorithmic Perspective ", CRC Press, 2009
2. Jlawei Han & Micheline Kamber, "**Data Mining: Concepts and Techniques**", 3rd Edition, Morgan Kaufmann Publishers, 2012
3. Pang-Ning Tan et al., "**Introduction to Data Mining**", 2nd Edition, Pearson Publications, 2019.

References:

1. Tom M. Mitchell, "Machine Learning ",Tata McGraw Hill, 1997
2. Machine Learning" First Edition 2016,Packt Publishing Ltd.

Internal Assessment Pattern

Cognitive Level	Int. Test 1 (%)	Int. Test 2 (%)	Open book Test[1] (%)
Remember	20	20	--
Understand	50	30	--
Apply	30	30	80
Analyze	--	20	20
Evaluate	--	--	--
Create	--	--	--
Total (%)	**100**	**100**	**100**

SAMPLE QUESTION (S)

Remember

1. Define machine learning
2. List the various similarity measures

Understand

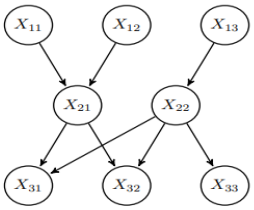
1. Explain the need of pre-processing in machine learning
2. Differentiate between supervised and unsupervised algorithms

Apply

1. Give Decision trees to represent the Boolean Functions:
   - **a.** A && - B
   - **b.** b. A V [B && C]
   - **c.** c. A XOR B
   - **d.** d. [A&&B] V [C&&D]

Open Book Question(s)

1. Suppose we clustered a set of N data points using two different clustering algorithms: k-means and Gaussian mixtures. In both cases we obtained 5 clusters and in both cases the centers of the clusters are exactly the same. Can 3 points that are assigned to different clusters in the kmeans solution be assigned to the same cluster in the Gaussian mixture solution? If no, explain. If so, sketch an example or explain in 1-2 sentences
2. Assume we have a set of data from patients who have visited UPMC hospital during the year 2011. A set of features (e.g., temperature, height) have been also extracted for each patient. Our goal is to decide whether a new visiting patient has any of diabetes, heart disease, or Alzheimer (a patient can have one or more of these diseases). Suggest a suitable machine learning approach for the above situation
3. Consider a Bayesian network B with boolean variables



   a) Is there any variable(s) conditionally independent of X33 given X11 and X12? If so, list all
   b) Is there any variable(s) conditionally independent of X33 given X22? If so, list all
   c) Write the joint probability P(X11, X12, X13, X21, X22, X31, X32, X33) factored according to the Bayes net. How many parameters are necessary to define the conditional probability distributions for this Bayesian network?
   d) Write an expression for P(X13 = 0, X22 = 1, X33 = 0) in terms of the conditional probability distributions given in your answer to part (c).

23IT308 Database Management Systems Laboratory

0 0 3 1.5

Course Outcomes

At the end of the course, students will be able to

1. Demonstrate ER Modelling concepts to design the Database
2. Apply integrity constraints on a database
3. Apply different DDL, DML, DCL, TCL commands in creation and manipulation of Database
4. Implement subqueries to make the complex queries more readable and apply joins to combine different relations to get required data from database
5. Implementation of SQL conversion, number and string functions
6. Experiment with triggers, cursors and functions to maintain referential integrity of data

CO-PO Mapping

CO	PO_1	PO_2	PO_3	PO_4	PO_5	PO9	PSO2
1	3	3	2	1	1	1	2
2	3	3	3	2	2	2	1
3	3	3	2	2	2	1	2
4	3	3	3	1	1	2	2
5	3	3	3	2	2	2	1
6	3	3	2	2	2	2	2

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

List of Experiments

Experiment 1

Application: 1 "Established in 1977, "Roadway Travels" has been actively operating for decades, providing connectivity across various destinations in India through a fleet of several buses. Headquartered in Hyderabad, the company now aims to modernize its operations in the following key areas:

1. Reservation System:

Managing reservations is the responsibility of the booking office. Customers can reserve their seats up to 60 days in advance, utilizing either cash or credit payment methods. In instances where tickets are unavailable, customers receive wait-listed tickets, which can be confirmed once cancellations occur.

2. Ticketing:

The booking office handles ticketing processes, including sales and issuance. Both cash and credit payments are accepted for this purpose.

3. Cancellation and Modification:

Cancellation services are also facilitated at the booking office. Applicable cancellation charges will be levied as per company policy. Furthermore, modifications to reservations can be processed within this domain. Notably, wait-listed tickets that remain unconfirmed are eligible for full refunds."

The Following are the entities:

Bus (Busno, Source, Destination, coachtype)

Reservation (PNRNo, Busno, NoofSeats, JourneyDt, ContactNo, Address)

Ticket (TicketNo, BusNo, JourneyDt, Source, Destination, Dept_Time, Age, Sex

Passenger (TicketNo, PNRNo, Name, ContactNo, Age, Sex)

Cancellation (PNRNo, JourneyDt, SeatNo, ContactNo)

Application: 2 Online Learning Platform: An online learning platform requires an ER model to handle courses, students, instructors, and enrollments. Courses have titles, descriptions, and durations. Instructors have names, expertise, and contact details. Students have profiles with names, email addresses, and student IDs. Enrollments link students to courses, including enrollment dates.

Entities and Attributes:

Course (Course_ID (Primary Key), Title, Description, Duration (in weeks or hours) Level (Beginner, Intermediate, Advanced), Category (e.g., Programming, Mathematics, Language), Price, Language, Thumbnail Image URL,)

Instructor (Instructor_ID (Primary Key), Name, Bio, ContactInfo (Email, Phone), Expertise (Areas of specialization), ProfilePictureURL)

Student (Student_ID (Primary Key), FirstName. LastName, Email, DOB, Address, Contact)

Enrollment (Enrollment_ID (Primary Key), Course_ID (Foreign Key referencing Course), Student_ID (Foreign Key referencing Student), EnrollmentDate, Status (Enrolled, Completed, Dropped, etc.))

Lesson (Lesson_ID (Primary Key), Course_ID (Foreign Key referencing Course), Title

Order (Sequence within the course), Duration (in minutes or hours), Content (Text, Video URL, PDF URL,

etc.))
Quiz (Quiz_ID (Primary Key), Course_ID (Foreign Key referencing Course), Title, Number of Questions, Passing Score)
Question (Question_ID (Primary Key), Quiz_ID (Foreign Key referencing Quiz), Question Text, Question Type (Multiple Choice, True/False, Short Answer, etc.))
Answer (Answer_ID (Primary Key), Question_ID (Foreign Key referencing Question), AnswerText, Correct_Answer_Indicator)
Application: 3 Music Streaming Service: A music streaming service needs an ER model to organize its content and user interactions. The system should handle songs, albums, artists, playlists, and users. Songs have titles, durations, and release years. Albums include album names, release dates, and cover art. Artists have names, genres, and biographies. Users have profiles with names, email addresses, and subscription details. Playlists contain songs and are created by users.
Application: 4 Company Database: The Company is organized into departments. Each department has a unique name, a unique number, and a particular employee who manages the department. We keep track of the start date when that employee began managing the department. A department may have several locations. A department controls several projects, each of which has a unique name, a unique number, and a single location. We store each employee's name, social security number, address, salary, sex, and birthdate. An employee is assigned to one department but may work on several projects, which are not necessarily controlled by the same department. We keep track of the number of hours per week that an employee works on each project. We also keep track of the direct supervisor of each employee. We want to keep track of the dependents of each employee for insurance purposes. We keep each dependent's name, sex, birthdate, and relationship to the employee.
Entities are:
Company Database:
EMPLOYEE (SSN, EName, Address, Sex, Bdate, Salary, SuperSSN, DNo)
DEPARTMENT (DNo, DName, MgrSSN, MgrStartDate)
DLOCATION (DNo,DLoc)
PROJECT (PNo, PName, PLocation, DNo)
WORKS_ON (SSN, PNo, Hours)
DEPENDENT (Essn, Dependent_name, Sex, Bdate, Relationship)
Apply ER Model for the above applications and identify the relations present in each application represent them in ER diagram. & Convert ER model to Relational Model.
Note: Students need to form a group minimum of 3 students and maximum of 5, and should design at least two applications into ER model by using EdrawMax software and convert them into relational model

Experiment 2

Creating users, roles and grant privileges to users and relations.

Experiment 3

Create and alter tables for Company Database in SQL using Integrity Constraints.

Integrity Constraints are CHECK, NOT_NULL, DEFAULT, UNIQUE, PRIMARY KEY, FOREIGN KEY. Use atleast one constraint in one relation to understand the usage of constraints on database.

Schema: CREATE TABLE employee (Fname varchar (15) NOT NULL, Minit varchar(1), Lname varchar(15) NOT NULL, SSN varchar(9) NOT NULL, Bdate date, Address varchar(30), Sex varchar(1), Salary float, Super_ssn varchar(9), Dno int NOT NULL, PRIMARY KEY (ssn), FOREIGN KEY (Super_ssn) REFERENCES employee(ssn));

CREATE TABLE department (Dname varchar(15) NOT NULL, Dnumber int NOT NULL, Mgr_ssn varchar(9) NOT NULL, Mgr_start_date date, PRIMARY KEY (Dnumber), UNIQUE(Dname), FOREIGN KEY (Mgr_ssn) REFERENCES employee(ssn));

CREATE TABLE dept_locations (Dnumber int NOT NULL, Dlocation varchar(15), PRIMARY KEY (Dnumber, Dlocation), FOREIGN KEY (Dnumber) REFERENCES department (Dnumber));

CREATE TABLE project (Pname varchar(15) NOT NULL, Pnumber int NOT NULL, Plocation varchar(15), Dnum int NOT NULL, PRIMARY KEY (Pnumber), UNIQUE (Pname), FOREIGN KEY (Dnum) REFERENCES department (Dnumber));

CREATE TABLE works_on (Essn varchar(9) NOT NULL, Pno int NOT NULL, Hours float NOT NULL, PRIMARY KEY (Essn, Pno), FOREIGN KEY (Essn) REFERENCES employee(ssn), FOREIGN KEY (Pno) REFERENCES project(Pnumber));

CREATE TABLE dependent (Essn varchar(9) NOT NULL, Dependent_name varchar(15) NOT NULL, Sex

varchar(1), Bdate date, Relationship varchar(8), PRIMARY KEY (Essn, Dependent_name), FOREIGN KEY (Essn) REFERENCES employee(ssn));

Experiment 4: Execute basic SQL queries on Employee Database

To Create a database called Employee which consists of EMP, Dept, Salgrade, Project, Works_on and Dependent tables as relations in the database use Data Definition Language command (Create, Alter).

Run sql queries on database using DML commands like Insert, Update, Delete and for retrieval of data from database use Select statement.

Employee Database:

EMP (Empno, Ename, Job, Mgr, Hiredate, sal, comm, deptno)

DEPT (Deptno, Dname, Location)

SALGRADE (Grade, Losal, Hisal)

PROJECT (Pno, Pname, Plocation, Deptno)

WORKS_ON (Empno, Pno, Hours)

DEPENDENTS (EMPNO, DependentName, Sex, Bdate, Relation)

4.1 Insertion:

Insert rows into table by using datatypes like varchar, int, date, float etc.,

Syn: INSERT INTO employee (Fname, Minit, Lname, SSN, Bdate, Address, Sex, Salary, Super_ssn, Dno) values ('James', 'E', 'Borg', 888665555, '10-NOV-1937′, '430 Stone, Houston, TX', 'M', 55000, NULL, 1);

4.2 Retrieval (Set operations)

Syn: Select Name from EMP Union Select Salary from EMP;

4.3 Updating

Update rows by changing the values in a row using update command syntax.

4.4 Deletion

Delete an entire row or multiple rows from the table using this command

Experiment 5

Implement the SQL operations on Employee Database created in Experiment 4.

Operations are: In, Any, All, Between, Like, Exists, Group by, Order by, having.

Employee Database:

EMP (Empno, Ename, Job, Mgr, Hiredate, sal, comm, deptno)

DEPT (Deptno, Dname, Location)

SALGRADE (Grade, Losal, Hisal)

PROJECT (Pno, Pname, Plocation, Deptno)

WORKS_ON (Empno, Pno, Hours)

DEPENDENTS (EMPNO, DependentName, Sex, Bdate, Relation)

Experiment 6

Implementing the concepts of Rollback – commit, checkpoints, and Views.

Rollback – commit, checkpoints:

Create Teacher table with the following fields(Name, DeptNo, Date of joining, DeptName, Location, Salary)

1. Insert five records
2. Give Increment of 25% salary for Mathematics Department
3. Perform Rollback command
4. Give Increment of 15% salary for Commerce Department
5. Perform commit command.

Views:

To create different views, use the Employee Database

1. Create a view to display Hire date of all employees.
2. Create a view to display no of working hours of all employees with Location.

Experiment 7

Apply different types of joins available in SQL on company database created in Experiment 4.

Types of Joins:

- Inner Join
- Outer Join
- Letf Join
- Right Join

Experiment 8

Apply subqueries and nested queries on company database (refer to experiment 4).

Experiment 9

Implement built in functions in oracle (Numeric, Aggregate functions) on Dual table (provided by Oracle) and Employee database.

Numeric Functions: ABS, CEIL, FLOOR, SQRT etc.,

Aggregate Functions: MIN, MAX, SUM, COUNT, AVG

Experiment 10

Implement String handling and Conversion functions on Dual table (provided by Oracle) and Employee database.

String Functions: Lowercase, Uppercase, Initcap, Rtrim, Ltrim, Lpad, Rpad etc.,
Conversion Functions: TO_CHAR(), TO_DATE(), TO_NUMBER().

Experiment 11

Sql Practice links from online platforms:

Employee Bonus	https://leetcode.com/problems/employee-bonus/
Big Countries	https://leetcode.com/problems/big-countries/
Employees Earning more than their managers	https://leetcode.com/problems/employees-earning-more-than-their-managers/
Managers with at least 5 direct reports	https://leetcode.com/problems/managers-with-at-least-5-direct-reports/
Second Highest Salary	https://leetcode.com/problems/second-highest-salary/
Count Salary Categories	https://leetcode.com/problems/count-salary-categories/

Experiment 12

Triangle	https://www.hackerrank.com/challenges/what-type-of triangle/problem?isFullScreen=true
Average Population of each country in a continent	https://www.hackerrank.com/challenges/average-population-of-each-continent/problem?isFullScreen=true
Top Earners	https://www.hackerrank.com/challenges/earnings-ofemployees/problem?isFullScreen=true
Occupations	https://www.hackerrank.com/challenges/occupations/problem?isFullScreen=true
New Company	https://www.hackerrank.com/challenges/the-ompany/problem?isFullScreen=true
15Days of Learning SQL	https://www.hackerrank.com/challenges/15-days-of-learning-sql/problem?isFullScreen=true

Experiment 13

Implement PL/SQL operations on Employee Database to retrieve one row as output.

Experiment 14

Implementing functions, stored procedures on company database using PL/SQL

Experiment 15

Implement cursors on Employee database using PL/SQL to retrieve multiple rows as output.

Experiment 16

Creating triggers using PL/SQL

List of Augmented Experiments

1. Inventory control management System
2. College Management System
3. Hospital management System
4. Library management System
5. Payroll management System
6. Health care organization Management System
7. Restaurant Management System
8. Blood Donation Management System
9. Art Gallery Management System
10. Hotel Management System
11. School Management System
12. Salary Management System
13. Wholesale Management System
14. Timetable Management System
15. Website Management

Reading Material (s)

1. Database Management Systems Lab Manual, Department of CSE, AI&ML, AI&DS and IT, GMRIT, Rajam

E-RESOURCES AND OTHER DIGITAL MATERIAL

[1] Prof Arnab Bhattacharya IIT Kanpur, SQL Introduction https://nptel.ac.in/courses/106104135/10
[2] Prof Arnab Bhattacharya IIT Kanpur SQL: Updates, Joins, Views, and Triggers
https://nptel.ac.in/courses/106104135/11

23ML407 Foundations of Machine Learning Lab 0 0 3 1.5

Course Outcomes

1. Analyze the relationship between AI, ML, and Data Science, and explore real-world problems using various types of data.
2. Perform basic statistical analysis and visualization techniques to explore and understand different datasets.
3. Apply similarity measures and data preprocessing techniques to prepare data for machine learning models.
4. Build and evaluate regression and classification models using algorithms like linear regression, logistic regression, decision trees, SVM, and ANN.
5. Apply ensemble learning techniques such as bagging, boosting, and random forests to improve model accuracy.
6. Implement clustering algorithms and explore data science tools to complete a simple data science workflow.

CO–PO Mapping

CO's	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	2	3	2	-	-	-	-	-	-	-	1	1
CO2	3	2	2	3	3	-	-	-	-	-	-	-	2	2
CO3	3	2	1	3	1	-	-	-	-	-	-	-	3	3
CO4	3	2	2	2	2	-	-	-	-	-	-	-	3	3
CO5	3	2	2	2	2	-	-	-	-	-	-	-	3	3
CO6	3	2	1	1	2	-	-	-	-	-	-	-	3	3

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Experiments:

1. **Explore the relationship between AI, ML, and Data Science with real-world use cases**
   (Identify and analyze data science problems from business, healthcare, or finance.)
2. **Load and explore structured, semi-structured, and unstructured datasets**
   (Perform basic exploration on different data formats (CSV, JSON, text).)
3. **Analyze datasets using statistical measures**
   (Calculate mean, median, mode, variance, standard deviation, skewness, and kurtosis.)
4. **Create visualizations for data exploration**
   (Plot univariate (histograms, boxplots), multivariate (scatter plots, pair plots), and high-dimensional visualizations using libraries like Matplotlib or Seaborn.)
5. **Compute similarity and dissimilarity measures for various data types**
   (Calculate cosine similarity, Jaccard index, Euclidean distance, and dissimilarity matrices for different attribute types.)
6. **Handle mixed-type data for similarity analysis**
   (Implement Gower's similarity or other techniques to compute distance for mixed attribute types.)
7. **Build a preprocessing pipeline**
   (Apply data cleaning, missing value treatment, and data integration on a dataset.)
8. **Perform data transformation and reduction**
   (Use normalization, discretization, and PCA for dimensionality reduction.)
9. **Implement Linear and Logistic Regression models**
   (Predict housing prices (linear) and classify customers (logistic) using real datasets.)
10. **Apply Regularization techniques (Lasso and Ridge)**
    (Compare results of models with and without regularization on regression tasks.)
11. **Build and evaluate classification models using Decision Trees and k-NN**
    (Train and test models; analyze confusion matrix, accuracy, and ROC-AUC.)
12. **Apply SVM for classification problems**
    (Implement SVM using kernel tricks)
13. **Build ensemble models using Voting, Bagging, and Boosting**
    (Compare performance of individual models vs ensemble on a classification dataset.)
14. **Implement and evaluate Random Forest Classifier**
    (Analyze feature importance and compare results with basic decision trees.)
15. **Apply Clustering algorithms (K-Means, Agglomerative, DBSCAN)**
    (Visualize clustering results and evaluate using silhouette score and Davies–Bouldin index.)
16. **Explore and use Data Science tools (Jupyter, Google Colab, R, Orange)**
    (Demonstrate a simple data science pipeline using one or more of these tools.)

Reading Material (s)

1. Foundations of Machine Learning Lab Manual, Department of CSE-AI&ML, GMRIT, Rajam

23ESX01 Employability Skills I

0 0 2 2

Course Outcomes

1. Demonstrate oral communication and writing skills as an individual to present ideas coherently
2. Develop life skills with behavioral etiquettes and personal grooming.
3. Assess analytical and aptitude skills.
4. Develop algorithms for engineering applications.
5. Solve engineering problems using software.
6. Utilize simulation tools for testing

COs –POs Mapping

CO's	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	2	1	2	3	2	-	-	-	2	2	2	3	-	-
CO2	3	2	3	2	3	-	-	-	2	2	2	3	-	-
CO3	3	2	3	2	3	-	-	-	2	2	2	3	-	-
CO4	3	2	3	2	2	-	-	-	3	3	3	3	-	-
CO5	3	2	3	2	2	-	-	-	3	3	3	3	-	-
CO6	2	1	2	3	2	-	-	-	3	3	3	3	-	-

3–Strongly linked | 2–Moderately linked| 1–Weakly linked

Unit-I

Communication Skills **8 hours**

Building Confidence: Fear? Steps to Overcoming the Fear of Public Speaking? Self Esteem: Definition? Types of Self Esteem, Causes of Low Self Esteem, Merits of Positive Self Esteem and Steps to build a positive Self Esteem.

Group Discussions (Practice): GD? GD Vs Debate, Overview of a GD , Skills assessed in a GD, Dos & Don'ts, & Conducting practice sessions (Simple Topics).

Motivational Talk: Team Work: Team Vs Group? Stages in Team Building, Mistakes to avoid and Lessons to Learn (Through Stories or Can be a Case Specific)

Quantitative Aptitude **8 hours**

Percentages, Profit and loss, Mixtures and Allegations, Simple Interest, Compound Interest

Unit-II

Competitive Programming Phase – II **16 Hours**

Module-1:

Fundamental mathematical concepts: Basic Mathematics, Number Theory, Geometry - Bit manipulation - Two-pointer technique.

Practice Problems: Codechef, Hackerrank, Codeforces, Leetcode, interviewbit

Module-2:

Data Structures: Searching - Linked List - Stack – Queue – Graph Traversal Methods - Hashing Techniques.

Practice Problems: Codechef, Hackerrank, Codeforces, Leetcode, interviewbit

References:

1. https://leetcode.com
2. https://www.codechef.com
3. https://www.hackerearth.com
4. https://www.interviewbit.com
5. https://www.hackerrank.com

5th Semester

23IT405 Web Technologies

3 0 2 4

Course Outcomes

At the end of the course, students will be able to

1. Illustrate client-side scripting tools to create and modify web pages.
2. Apply CSS framework to design responsive and interactive web pages
3. Examine jQuery for efficient element manipulation and work with JSON objects for seamless data exchange
4. Demonstrate server-side scripting languages to develop business logic.
5. Make use of database connectivity to communicate database server from web server.
6. Test for client-server web application development in real time internet domains

CO–PO Mapping

COs	PO_1	PO_2	PO_3	PO_5	PSO_1	PSO_2
1	3	3	3	2	3	3
2	3	3	3	2	3	3
3	2	2	2	3	3	3
4	2	2	2	3	2	3
5	3	3	3	3	3	3
6	2	2	2	2	3	3

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Syllabus

Unit- I **12+8 Hours**

HTML Tags

Introduction, Links, Lists, Tables, Forms, Canvas, Semantics, Plug-ins.**CSS:** Introduction, CSS Properties, Selectors, Combinators, Controlling Fonts, Forms, Pseudo classes, Transitions, Animations (data-aos), 2D and 3D Transforms, Navigation Bar, Layouts,

Java Script: Introduction, Functions, Events, Validations, Objects, Document Object Model (DOM), Division Replacement, Browser Object Model (BOM)

CSS Tooltips, Font Combinations.

Practical Components

1. Design the following static web pages required for Online Examination System web site.
    a) Home Page
    b) Login Page: a. Student Login Page b. Admin Login Page
    c) Registration page
    d) Test Page
    e) Results Page.
2. For Specific pages of Online Examination System like Test page and Result page Apply different font styles, font families, font colors, animations and other formatting styles to the above static web pages.
3. Validate login page and registration page using Java Script.
4. Create an HTML page with any required JavaScript that takes a number from text field in the range of 0 to 999 and shows it in words. It should not accept four and above digits, alphabets and special characters.

Unit- II **12+8 Hours**

Bootstrap

Responsive Design, Layouts, Grids, Media Queries, **Components**- Forms, Drop Downs, Cards, Input Groups, Collapse, List Groups, Navbar, Popovers, Flex Box. **Utilities**- Borders, Positions, Scroll spy, Visibility, CSS Components, SVG, Java Script Components. **JSON:** Introduction, Parsing, Objects, JSON arrays, Dynamic HTML Data. **jQuery:** Selectors, Filters, Form Plug-ins, Auto Validations

Bootstrap icons, Jumbotron.

Practical Components

1. Design an interactive area using HTML and CSS. Use JavaScript to implement the color change effect during mouseover and reversion during mouse out.
2. **Inner Function with Constant Addition**: Design a function “createAdderWithConstant” that takes a constant as a parameter and returns an inner function. The inner function, when given a number, adds the constant to it and returns the result. Use the inner function to add the same constant to different numbers.
3. Make the static pages Responsive and attractive using Bootstrap components (***Mobile View)
4. Validate login page and registration page using JQuery**.**

Unit- III **12+8 Hours**

PHP

Introduction, Modal View Architecture (MVC), Creating PHP script, Running PHP script, Introduction to PHP, Arrays, Functions, sending parameters in URL, working with Forms, **working with Database:** Running SQL Queries, connecting with Databases, Uploading Files, PHP Sessions, Redirecting, Routing, Connecting PHP with JavaScript and jQuery, Authentication.

PHP Cookies, PHP Exception Handling, Introduction to PHP Laravel

Practical Components

1. Write a PHP to connect to the database, Insert the details of the student who registered through Online Examination System student login page including photograph.
2. Develop and demonstrate PHP Script for the following problems:
    a. Write a PHP Script to find out the Sum of the Individual Digits.
    b. Write a PHP Script to check whether the given number is Palindrome or not
3. Write a PHP program to insert the questions and respective answers into the database through admin page, sending mails to registered students through admin.
4. Write a PHP program, assuming four users user1, user2, user3 and user4 having the passwords pwd1, pwd2, pwd3 and pwd4 respectively. Create a Cookie and add these four user ID"s and passwords to this Cookie. Read the user id and passwords entered in the Login form (week1) and authenticate with the values (user id and passwords) available in the cookies.
    a. If he is a valid user (i.e., user-name and password match) you should welcome him by name (user-name) else you should display "You are not an authenticated user"

Unit- IV **12+8 Hours**

Flask

Introduction to Flask, Virtual Environment, features of flask, url building, routing, Templates and Jinja Code, Rendering Templates, Static files, Building Forms, Sending Form data to Templates, Template Inheritance (header, footer, etc), Session Tracking, connecting database (SQLite), Retrieving database values to templates, File uploading, Sending Mails, Deployment of website.

wsgi file, csrf token, http methods.

Practical Components

1. Write a Flask code to connect to the database, and authenticate Login pages with jQuery and start a Session for Student, Change password for Student
2. After user login, display Questions from the database into test page in shuffled manner using flask and store in database.
3. Write a Flask code to do the following
    a) Evaluate the answers of the test which is given by the student in the Test page.
    b) Calculate the total score of the student and store it into the database and display score in dashboard of student.
4. Using flask retrieve results in a responsive table format of all students with filtering. Deploy the application developed in real time environment

Total: 48+32 Hours

Textbook(s):

1. Programming the World Wide Web, 8th edition Robert W. Sebesta, pearson.
2. Bootstrap: Responsive Web Development, 1st Edition, jake spurlock foreword by dave winer, O'Reilly publications.

Reference Book(s):

1. Web programming with HTML, XHTML and CSS, 2e, Jon Duckett, Wiley India
2. Web programming Bai, Michael Ekedahl, CENAGE Learning, India edition.
3. An Introduction to Web Design + Programming, Paul S.Wang, India Edition
4. Flask Web Development 2e Miquel Grinberg O'Reilly publications.

Internal Assessment Pattern

Cognitive Level	Int. Test 1 (%)	Int. Test 2 (%)	Lab Examination (%)
Remember	20	20	--
Understand	50	40	--
Apply	30	40	40
Analyze	--	--	20
Evaluate	--	--	40
Create	--	--	--
Total (%)	100	100	100

Sample Question (S)

Understand

1. What is the use of cascading style sheet? Explain different style sheets used in HTML.
2. Explain string object with suitable example.
3. Illustrate how work java script events.
4. Describe procedure for how to connect PHP to MYSQL.
5. Explain AJAX technologies

Apply

1. Write an HTML program to display your class timetable.
2. Write a java script to find reverse of a given number using functions.
3. Design a responsive registration page using bootstrap.
4. Insert an image into a web page. Write a script which displays a message when the mouse is over the image. The co-ordinates of the mouse should be displayed if click is attempted on the image.
5. A person X has created the table (emp) and inserted the data in the table. After the completion of insertion, he found that one of the inserted record is invalid. So, write a PHP program to update the table by deleting the invalid record. [Assume database table name is emp (empno, ename, dept, age)]

Analyse

1. Compare traditional web application development with AJAX and analyze what way AJAX useful for real time internet application.

23ML502 Neural Networks

3 0 0 3

Course Outcomes

1. Illustrate the basics of the Neural Networks and their roles in building intelligent machines
2. Analyse the behaviour of basic neural network models in solving linearly and non-linearly separable problems.
3. Design feedforward neural networks using backpropagation and gradient descent techniques.
4. Describe associative memory networks and demonstrate pattern storage and retrieval using feedback models like Hopfield Networks.
5. Explain different unsupervised learning models and their learning mechanisms..
6. Apply neural network models to solve real-world problems such as character recognition, image classification, and signal processing.

COs–POs Mapping

COs	PO1	PO2	PO3	PO5	PO12	PSO1	PSO2
1	3	2	2	1	1	2	2
2	3	2	2	1	1	3	3
3	3	3	3	1	1	3	3
4	3	3	3	1	2	3	3
5	3	3	3	3	2	3	3
6	3	3	3	3	3	3	3

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I **12 Hours**

Introduction

Introduction to Neural Networks, Benefits of Neural Networks, Biological Neural Networks Vs Artificial Neural Networks.

Artificial Neural Networks: Introduction to Artificial Neural Networks, Neural Networks Terminologies, Activation Functions and its types, Basic Models of Neural Networks-McCulloch-Pitts Neuron and Hebb Network.

Types of Learning Laws, *Evolution of Neural Network Models*

Unit II

Supervised Learning Networks **12 Hours**

Perceptron Networks: Perceptron, Perceptron Learning Algorithm, Linearly Separable Problem (AND, OR, XOR), Perceptron Convergence Theorem.

Multi-Layer Feedforward Networks: Introduction to Feed Forward Neural Networks, Multilayer Perceptron Architecture and Layers, Cross Validation, Gradient Descent Algorithm, Stochastic Gradient Descent Algorithm, Back Propagation Algorithm.

Overfitting and Underfitting in Neural Networks, Vanishing and Exploding Gradient Problem

Unit III

Associative and Feedback Networks **12 Hours**

Associative Memory: Hetero-associative Memory Neural Network, Auto-associative Network, Bidirectional Associative Memory (BAM).

Feedback Networks: Hopfield Networks - Energy Function and Convergence - Pattern Storage and Retrieval

Limitations of Associative Memories and Hopfield Networks

Unit IV **12 Hours**

Un-Supervised Learning Networks: Competitive Learning, Kohonen Self-Organizing Maps, Learning Vector Quantization (VLQ), Adaptive Resonance Theory (ART1).

Applications of Neural Network: Character Recognition, Signal Processing, Image Classification.

Comparison of SOM, and ART1, Applications of SOM

Total: 48 Hours

Textbook (s)

1. Laurene Fausett, "Fundamentals of Neural Networks" , Pearson Education, 2004.
2. Simon Haykin, "Neural Networks - A comprehensive foundation", Pearson Education, 2003.
3. S. N. Sivanandam, S N Deepa, Principles of Soft Computing, 2ndEditionWiley India, 2007.

4. Bengio, Yoshua, Ian J. Goodfellow, and Aaron Courville. "Deep Learning." An MIT Press book in preparation, 2015.

Reference (s)

1. Satish Kumar,"Neural Networks: A Classroom Approach" Tata McGraw Hill Education, 2004.
2. Simon Haykin, " Neural networks and Learning Machines", Prentice Hall, 2008
3. Josh Patterson and Adam Gibson "Deep Learning A Practitioner's Approach" O'Reilly Media, Inc. 2017

Internal Assessment Pattern

Cognitive Level	Int. Test 1 (%)	Int. Test 2 (%)	Open Book Exam (%)
Remember	20	20	--
Understand	40	40	--
Apply	20	20	40
Analyze	20	20	40
Evaluate	--	--	20
Create	--	--	--
Total (%)	100	100	100

Sample Questions

Remember

1. Define the simple artificial neuron and the calculation of net input.
2. Define the bias and threshold
3. list out various activation functions

Understand

1. Explain the basic models of Artificial Neural Networks.
2. List the terminologies of artificial neural networks
3. Explain about the Neural Network with backpropagation algorithm and its architecture in detail.

Apply

1. Demonstrate the perceptron network architecture with training algorithm in detail
2. Demonstrate the Full Counter Propagation Network (CPN) algorithm with neat architecture.
3. Construct the convolution neural network with neat diagram.

Analyze

1. Justify the Back propagation network architecture and analyse its training algorithm
2. Analyse different layers of convolution neural network in detail.

Evaluate

1. Kohonen Self – Organizing net with two cluster units and five inputs. The weight vector foe cluster units are given by $W = \begin{matrix} 1.0 & 0.9 & 0.7 & 0.5 & 0.3 \\ 0.3 & 0.5 & 0.7 & 0.9 & 1.0 \end{matrix}$

   Use the square of the Euclidean distance to find the winning cluster unit for the input pattern x= [0.0 0.5 1.0 0.5 0.0].

   Using learning rate 0.25, find the new weights for the winning unit.

2. Consider the CPN net with using the input pair x= [0110] and y=[01] perform Phase 1 of training (one step only).Find the activation of the cluster layer units and update the weights using learning rates $\alpha = 0.2$ and $\beta = 0.3$.

   $V = \begin{bmatrix} 0.7 & 0.5 \\ 0.7 & 0.5 \\ 0.5 & 0.7 \\ 0.5 & 0.7 \end{bmatrix}$ $W = \begin{bmatrix} 0.2 & 0.2 \\ 0.2 & 0.2 \end{bmatrix}$

23DS503 Data Analytics and Visualization Techniques

3 0 2 4

Course Outcomes

1. Understand the Data Analysis Fundamentals
2. Apply data transformation techniques for preprocessing and feature engineering in datasets.
3. Analyze and visualize data using descriptive statistics, correlation measures, and advanced plotting techniques.
4. Apply data transformation, statistics, and plots like violin and heatmaps for data analysis.
5. Analyze and visualize associations using scatterplots, correlograms, and treemaps.
6. Analyze and visualize individual and multivariate time series data.

COs–POs Mapping

COs	PO1	PO2	PO3	PO4	PO5	P012	PSO1	PSO2
1	2	3	1	-	2	1	3	1
2	2	2	1	2	2	-	3	2
3	2	3	1	1	3	-	3	3
4	2	2	1	1	3	-	3	3
5	2	2	3	-	2	2	3	3
6	2	3	3	-	2	2	3	3

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I **12+8 Hours**

Data Analysis Fundamentals

Understanding data science, significance of EDA, steps in EDA. types of analysis (univariate, bivariate, multivariate). Making sense of data: Numerical data-Discrete &continuous data, categorical data, Measurment scales-Nominal, Ordinal, Interval, Ratio. Comparing EDA with classical and Bayesian analysis, getting started with EDA: Numpy, Pandas,Scipy, and Matplotlib.

Grouping of data: Groupby mechanics, rearranging, reshaping data structures, data aggregation methods, and cross-tabulation methods.

Objectives of Exploratory data Analysis, The applications of EDA

Practical Components

1: Explore Scipy Library
2: Explore Stat Library
3: Experiment with all group by functionalities & cross tabulation methods
4: Analyzing and Visualizing Real-World Data.

Unit II **12+8 Hours**

Data transformation and descriptive statistics

Transformation techniques- performing data deduplication, replacing values, handling missing data, renaming axis indexes, outlier detection and filtering. Permutation and random sampling, computing indicators/dummy variables, string manipulations.

Correlation and Inferences: Introducing correlation, covariance, Pearson's Correlation, Spearman's Rank Correlation.**Descriptive statistics:** Understanding statistics, measure of central tendenacy, measure of dispersion (standard deviation, variance, skewness, kurtosis, percentiles, quartiles)

Violin Plots, Inter Quartile Range, Discretization & binning, heatmaps

Practical Components

1: Experiment with various data pre-processing & Transformation techniques
2:Experiment and Analyse the correlations among attributes.
3: Experiment with the statistical measures
4: Outlier Detection and Handling.

Unit III **12+8 Hours**

Visualizing Distributions

Visualizing Data : Mapping data on to Aesthetics, Color Scales, Directory of Visualizations, Visualizing Amounts-Bar Plots, Grouped and Stacked Bars, Dot Plots and Heatmaps, Visualizing Distributions: Histograms and Density Plots-Visualizing a Single Distribution, Visualizing Multiple Distributions at the Same Time, Visualizing Distributions: Empirical Cumulative Distribution Functions and Q-Q Plots-Empirical Cumulative Distribution Functions, Highly Skewed Distributions, Quantile Plots, Visualizing Many Distributions at Once-Visualizing Distributions Along the Vertical Axis, Visualizing Distributions Along the Horizontal Axis

Kernel Density Estimation (KDE), Geospatial Distributions, Interactive Visualization

***Practical* Components**

Download the House Pricing dataset from Kaggle
1: Use different Color scales on the Rainfall Prediction dataset
2: Create different Bar plots for variables in any dataset
3: Visualization of different Distributions in various ways
4: Visualizing Empirical Cumulative Distribution Functions (ECDF) for Real-world Data.

Unit IV **12+8 Hours**

Visualizing Associations & Time Series

Stacked Densities, Visualizing Proportions Separately as Parts of the Total ,Visualizing Nested Proportions- Nested Proportions Gone Wrong, Mosaic Plots and Treemaps, Nested Pies ,Parallel Sets. Visualizing Associations Among Two or More Quantitative Variables-Scatterplots, Correlograms, Dimension Reduction, Paired Data. Fundamentals on Time Series Data, Visualizing Time Series and Other Functions of an Independent Variable-Individual Time Series , Multiple Time Series and Dose–Response Curves, Time Series of Two or More Response Variables

Multidimensional Scaling, MANOVA vs ANOVA

Practical Components

1: Visualization through Pie charts, Stacked Bars & Stacked Densities
2: Show an example of Skewed data and Experiment with different skewness removal techniques
3: For a sales dataset do a Time Series Visualization
4: Geospatial Visualization for Regional Sales Analysis

Total: 45+32 Hours

Textbook (s)

1. Daniel J. Denis: Univariate, Bivariate, and Multivariate Statistics Using R: Quantitative Tools for Data Analysis and Data Science, Wiley,2020
2. Mukhiya Suresh Kumar Mukhiya, Ahmed Usman Ahmed: Hands-On Exploratory Data Analysis with Python: Perform EDA techniques to understand, summarize, and investigate your data, Packt, 2020
3. Claus Wilke, "Fundamentals of Data Visualization: A Primer on Making Informative and Compelling Figures", 1st edition, O'Reilly Media Inc, 2019.
4. Downey, Allen. Think stats: exploratory data analysis. " O'Reilly Media, Inc.", 2014.
5. Neil H. Spencer: Essentials of Multivariate Data Analysis, CRC Press,2014

Reference (s)

1. Wes McKinney : Python for Data Analysis 2nd Edition,Wiley,2013
2. Glenn J. Myatt, Wayne P. Johnson: Making Sense of Data I: A Practical Guide to Exploratory Data Analysis and Data Mining, 2nd Edition,Wiley,2014
3. 1 Tony Fischetti, Brett Lantz, R: Data Analysis and Visualization,O'Reilly ,2016 2 OssamaEmbarak, Data Analysis and Visualization Using Python: Analyze Data to Create Visualizations for BI Systems,Apress, 2018
4. Wendy L. MartinezAngel R. MartinezJeffrey L. Solka: Exploratory Data Analysis with MATLAB, 2nd Edition,CRC Press,2011
5. Radhika Datar, Harish Garg : Hands-On Exploratory Data Analysis with R , Packt,2020
6. Joseph F Hair, Barry J. Babin, Rolph E. Anderson, William C. Black: Multivariate Data Analysis Cengage, 2018

SAMPLE QUESTION (S)

Internal Assessment Pattern

Cognitive Level	Int. Test 1 (%)	Int. Test 2 (%)	Open Book Exam (%)
Remember	50	40	--
Understand	30	40	--
Apply	20	20	50
Analyze	--	--	50
Evaluate	--	--	--
Create	--	--	--
Total (%)	100	100	100

Remember

1. What is data science.
2. Define purpose of different data visualization aids
3. List any two steps involved in EDA.

Understand

1. Explain different data representation
2. Explain the role of various data grouping methods
3. Explain purpose of hypothesis testing in detail

Apply

1. Apply various visual aids to identify behavior of data
2. Make use of Multivariate Analysis methods for data nanlysis
3. Examine data with various distribution function

Analyze

1. Compare and contrast various data grouping methods.
2. Distinguish various methods to address problems with dummy variables
3. Examine various string handling methods

Evaluate

1. Evaluate the statistics of a given dataset
2. Justify the importance of data skewness
3. Measure data dispersion methods on a dataset

Open Book Exam Questions

Question 1:

Perform below analysis on a house price prediction dataset.

a) Univariate Analysis
   i. Analysis of a numerical feature
   ii. Analysis of a categorical feature
b) Bivariate Analysis
   i. Relationship of a numerical feature with another numerical feature
   ii. Relationship of a numerical feature with a categorical feature

Question 2:

Perform below analysis on weather prediction dataset.

1. Correlation Analysis
   i. Correlation Heat Map
   ii. Zoomed Heat Map
2. Investigation of missing values
   1. What's missing? to what extent?
   2. Visualizing missing values in a DataFrame

23ML504 Computer Networks

3 0 0 3

Course Outcomes

1. Describe the basics of computer networks including hardware, software, transmission media, and switching methods.
2. Explain how the data link layer works, including error detection/correction and simple communication protocols.
3. Analyze how MAC protocols like ALOHA, CSMA, and various Ethernet types manage access to network media.
4. Illustrate how the network layer handles routing, congestion control, and IPv4/IPv6 addressing.
5. Compare TCP and UDP protocols, focusing on their features and control mechanisms for flow, errors, and congestion.
6. Demonstrate how application layer protocols like HTTP, Email, TELNET, and DNS work, including their structure and security aspects.

CO-PO Mapping:

CO's	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	1	-	2	1	-	2	-	-	-	-	2	1	-
CO2	3	2	-	2	1	-	2	-	-	-	-	2	2	-
CO3	2	3	-	2	2	-	2	-	-	-	-	2	1	-
CO4	3	3	-	3	3	-	2	-	-	-	-	2	1	-
CO5	2	3	-	3	3	-	2	-	-	-	-	2	2	-
CO6	3	1	-	3	2	-	2	-	-	-	-	2	2	-

3- Strongly linked, 2- Moderately linked, 1- Weakly linked

SYLLABUS:

UNIT I Network Hardware-Network Software-and Transmission Media **12 Hours**

Network Hardware-Network Software-and Transmission Media OSI-TCP/IP reference models, ARPANET, Network Topologies,

Physical Layer: Transmission media: Guided Media-Twisted Pair-Coaxial Cable-Fiber optics-Unguided Media: Electromagnetic Spectrum, Radio Transmission and Microwave Transmission. Switching Techniques: Circuit Switching-Packet Switching-Message Switching.

Taxonomy of networking devices, Network Types, LAN, MAN, WAN

Unit II Data link layer

12 Hours

Design issues, Framing: fixed size framing, variable size framing, error control, error detection and correction codes, CRC, Checksum: idea, one's complement internet checksum, services provided to Network Layer, Elementary Data Link Layer protocols: simplex protocol, Simplex stop and wait, Simplex protocol for Noisy Channel. Sliding window protocol: One bit, Go back N, Selective repeat-Stop and wait protocol,

Media Access Control(MAC): Random Access: ALOHA, Carrier sense multiple access (CSMA), CSMA with Collision Detection, CSMA with Collision Avoidance, Controlled Access: Reservation, Polling, Token Passing, Channelization: frequency division multiple Access(FDMA), time division multiple access(TDMA), code division multiple access(CDMA). Wired LANs: Ethernet, Ethernet Protocol, Standard Ethernet, Fast Ethernet(100 Mbps), Gigabit Ethernet, 10 Gigabit .

Data link layer in HDLC, Point to point protocol (PPP), Ethernet.IEEE Standard 802.3 and Ethernet-IEEE Standard 802.4: Token bus.

Unit III Network Layer

12 Hours

The Network Layer Design Issues – Store and Forward Packet Switching Services Provided to the Transport layer-Implementation of Connectionless Service, Implementation of Connection Oriented Service- Comparison of Virtual Circuit and Datagram Networks, Routing Algorithms-The Optimality principle-Shortest path, Flooding, Distance vector, Link state, Hierarchical, Congestion Control algorithms-General principles of congestion control, Congestion prevention polices, Approaches to Congestion Control-Traffic Aware Routing- Admission Control-Traffic Throttling-Load Shedding.

Internet Working: How networks differ- How networks can be connected- Tunnelling, internetwork routing-, Fragmentation, network layer in the internet – IP protocols-IP Version 4 protocol-IPV4 Header Format, IP addresses,

Class full Addressing, CIDR, Subnets-IP Version 6-The main IPV6 header, Transition from IPV4 to IPV6, Comparison of IPV4 & IPV6.
Traffic Control Algorithm-Leaky bucket & Token bucket.

Unit IV Transport Layer

12 Hours

Transport layer protocols: Introduction-services- port number-User data gram protocol-User datagram-UDP services-UDP applications Transmission control protocol: TCP services- TCP features- Segment- A TCP connection- windows in TCP- flow control-Error control, Congestion control in TCP.
Application Layer- World Wide Web: HTTP, Electronic mail-Architecture- web based mail- email security-TELENET-local versus remote Logging-Domain Name System.
ATM Transport protocol-Proxy Servers-Multimedia: Data compression-digital representation of analog signals

Total: 48 Hours

Textbook (s)

1. Andrew S Tanenbaum, Computer Networks, 4th Edition, Pearson Education /PHI, 2002.
2. Behrouz A. Forouzan, Data Communications and Networking, 3rdEdition, Tata McGraw Hill Higher Education, 2003

Reference (s)

1. Willam Stallings, Data and Computer Communications,8th Edition, Pearson Prentice Hall, 2007.
2. W.A. Shay, Thomson, Understanding communications and Networks, 3rd Edition, Cengage Learning, 2005

SAMPLE QUESTION (S)

Internal Assessment Pattern

Cognitive Level	Int. Test 1 (%)	Int. Test 2 (%)	Open Book Exam (%)
Remember	25	20	--
Understand	40	40	20
Apply	30	40	40
Analyze	--	--	40
Evaluate	--	--	--
Create	--	--	--
Total (%)	100	100	100

Sample Questions

Remember

1. List the applications of Computer Networks
2. Write two differences between OSI and TCP/IP models.
3. State 5 key assumptions in Dynamic channel allocation?
4. State the purpose of DNS.
5. Define congestion.

Understand

1. Illustrate OSI Reference model.
2. Explain various design issues of data link layer
3. What are the responsibilities of Data Link layer and explain Pure Aloha and Slotted Aloha protocols.
4. Describe Distance Vector routing algorithm with example and explain count to infinity problem 5. Represent the Manchester encoding for the bit stream: 0001110101.

Apply

1. What is the check summed frame transmitted if the message is 1101011011 and the generator polynomial is x 4 + x + 1 using CRC
2. Can you think of any circumstances under which an open-loop protocol, (e.g., a Hamming code) might be preferable to the feedback-type protocols?
3. Assuming that all routers and hosts are working properly and that all software in both is free of all errors, is there any chance, however small, that a packet will be delivered to the wrong destination?
4. The following data fragment occurs in the middle of a data stream for which the byte-stuffing algorithm described in the text is used: A B ESC C ESC FLAG FLAG D. What is the output after stuffing?
5. The following character encoding is used in a data link protocol: A: 01000111; B: 11100011; FLAG: 01111110; ESC: 11100000 Show the bit sequence transmitted (in binary) for the four-character frame: A B ESC FLAG when each of the following framing methods are used:
    1. Character count.
    2. Flag bytes with byte stuffing.
    3. Starting and ending flag bytes, with bit stuffing.

23MLC11 Computer Vision & Pattern Recognition (ML-Ops Career path)

3 0 0 3

Course Outcomes

1. Explain fundamental computer vision concepts, including applications and challenges.
2. Understand image processing techniques and their application in computer vision.
3. Apply feature detection and matching techniques to analyze image data.
4. Understand pattern recognition and machine learning concepts and their application in image analysis.
5. Describe and implement segmentation techniques to partition images into meaningful regions.
6. Understand and apply object detection and recognition techniques for identifying and analyzing objects in images or sequences.

CO–PO Mapping

CO	PO1	PO2	PO3	PO12	PSO1
1	3	2	1	1	1
2	3	2	2	1	1
3	3	3	3	1	3
4	3	3	3	2	3
5	2	3	3	2	3
6	2	3	3	2	3

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I **12 Hours**

Introduction to Computer Vision and Image Processing

Computer Vision: Introduction to computer vision - applications of computer vision - Image formation, Geometric primitives, 2D transformations, and photometric image formation - Sampling and aliasing in images: Image processing Techniques: Point operators: Thresholding, Image Enhancement - Linear filters - Non-Linear filters - Multi-resolution representations : Gaussian pyramid, Laplacian Pyramid - wavelets: 1D Haar wavelet transform , 2D Haar wavelet transform.

Neighborhood operators (morphological operations) and content-based image retrieval.

Unit II **12 Hours**

Feature Detection and Matching

Feature detection : Image features - Edge Detection - Overview of edge detection methods, Prewitt, Sobel, and Canny Edge detectors - Line detection: Hough Transform for line detection - Feature descriptors: Histogram of Oriented Gradients (HOG), Local Binary Pattern (LBP), SIFT and SURF- Feature Matching techniques - Performance Analysis.

Harris corner detection, and Piecewise Linear Approximation

Unit III **12 Hours**

Pattern Recognition and Machine Learning

Introduction to pattern recognition: Pattern, Features, RST invariant features - Pattern recognition systems - Different approaches to pattern recognition - Statistical Pattern Recognition: Bayesian decision theory and discriminant functions. Supervised learning: Parametric methods (linear regression), Nonparametric methods (k-nearest neighbors, decision trees) - Unsupervised learning: Clustering algorithms (k-means, mean shift clustering). Neural pattern recognition: Structure and types of neural networks, Learning algorithms for neural networks.

Dimensionality reduction methods- PCA (Principal Component Analysis) and LDA (Linear Discriminant Analysis)

Unit IV **12 Hours**

Image Analysis and Object Recognition

Segmentation techniques: Region-based segmentation: Split and merge, region growing, Contour-based segmentation: Active contours, Clustering-based segmentation: K-means, mean shift. Object detection: Techniques for object detection - Face recognition: Methods and challenges in face recognition - Instance recognition Techniques – Image Classification – Semantic segmentation - Motion estimation: Optical flow, activity recognition, motion estimation.

Medical image segmentation and Deep learning object detection

Total: 48 Hours

Textbook (s)

1. Richard Szeliski, Computer Vision: Algorithms and Applications , Springer, 2022, ISBN:978-1848829343
2. Rober.J. Shelkoff, John Wiley &Sons,Pattern Recognition- Statistical, Structural and Neural Approaches,Wiley, 2007, ISBN: 978-8126513703.

Reference (s)

1. R.O.Duda, P.E.Hart and D.G.Stork, Pattern Classification, Wiley India, 2006, ISBN: 978-8126511167.
2. Simon J. D. Prince, Computer Vision: Models, Learning, and Inference, Cambridge University Press, 2012,78-1107011793

3. Milan Sonka, Vaclav Hlavac, Roger Boyle, Image Processing, Analysis, and Machine Vision, Brooks/Cole, 2007, ISBN:978-0495082521.

Internal Assessment Pattern

Cognitive Level	Int. Test 1 (%)	Int. Test 2 (%)	Open Book Exam
Remember	30	30	-
Understand	50	50	-
Apply	20	20	50
Analyze	--	--	50
Evaluate	--	--	-
Create	--	--	-
Total (%)	**100**	**100**	**100**

Remember

1. What is computer vision and what is its role in image processing?
2. Define Hough transform.
3. List the roles of linear filtering in image processing.
4. What is Instance recognition?

Understand

1. How does sampling affect the quality of an image?
2. Explain the working principles of K-nearest neighbors (KNN) and decision trees in pattern recognition.
3. Explain the concept of wavelets and their use in multi-resolution representations.
4. Describe the process of feature detection in computer vision.
5. Describe the process of optical flow estimation and its applications in motion tracking.

Apply

1. Discuss the application of mean shift and mode finding in image segmentation.
2. Explain the concept of multi-resolution representations in image analysis. How are wavelets used in this context?

Analyze

1. Analyze the impact of aliasing in images and discuss methods to mitigate its effects.
2. Compare and contrast parametric and nonparametric methods in supervised learning for pattern recognition.
3. Analyze the challenges and potential solutions for object tracking in crowded or occluded scenes.

23CSC21 Backend Programming Languages (Full Stack Developer Career Path)

3 0 0 3

Course Outcomes:

1. Demonstrate a comprehensive understanding of web fundamentals.
2. Apply RESTful API for modern web applications.
3. Develop and deploy basic web applications using Node.js and Express.js.
4. Demonstrate the ability to understand C# concepts and build web applications using ASP.NET MVC.
5. Create efficient and dynamic RESTful APIs with CRUD functionality.
6. Design and develop web applications using Python.

CO-PO Mapping:

COs	PO1	PO2	PO3	PO5	PSO1	PSO2
1	3	2	1	1	2	1
2	3	3	3	2	3	3
3	3	3	3	3	3	3
4	3	3	3	3	3	3
5	3	3	3	3	3	3
6	3	3	3	3	3	3

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit-I **12 Hours**

Overview of the web: Basic concepts of web, Anatomy of HTTP transaction, Request and response structures, HTTP status codes, REST APIs, Browser storage - Local storage & Session storage

RESTful programming: Introduction to Restful API, REST Principles, Resource Identification, Representation of Resources: JSON, XML and Plain Text/HTML, HTTP Methods: GET, POST, PUT and DELETE, Stateless Communication, Versioning, Error Handling, Pagination.

Basic HTML Tags, Basic CSS concepts, Authentication and Authorization in RESTful

Unit-II **12 Hours**

Programming with JavaScript

Introduction to Node.js, Features of Node.js, npm, understanding package. json, Modules: fs, http, path and os, Express.js: Introduction to Express.js, Setting up an Express server, Handling GET and POST requests, Middleware in Express, Template Engine: Embedded JavaScript Templates, rendering dynamic HTML, building a REST API: Creating routes for CRUD operations, understanding authentication vs. authorization, Implementing basic JWT.

Basics of JavaScript and ES6 JavaScript, Node Packages, async, await

Unit-III **14Hours**

Programming with C#

Introduction to .NET: Introduction, .NET Framework vs .NET Core vs .NET, components of .NET, CLR, FCL, Managed code and Unmanaged code, C# Concepts: Introduction, Features, Datatypes, Functions, Arrays, Classes and Objects, Properties, Inheritance, Polymorphism, Interfaces, Namespaces, Collections, Delegates, Events, Exception Handling, Introduction to ASP.NET: MVC, Routing, Razor pages vs MVC, Create RESTful endpoints, Model Binding

C# variables, LINQ, async/await, Multithreading, ORM, ASP.NET life cycle, web forms, ADO.NET introduction

Unit-IV **10 Hours**

Programming with Python

Introduction to Python, Classes and objects, Constructors, Encapsulation, Inheritance, Polymorphism, Decorators and Closures, Abstract Base Classes, Generators and Iterators, Metaclasses, Functional Programming: map(), filter(), reduce(), Python's HTTP package, building a REST API: creating routes for CRUD operations.

Basic concepts of python, Python packages

Total: 48 Hours

Textbook (s)

1. Restful Web APIs, Leonard Richardson, Mike Amundsen, Sam Ruby, O'Reilly Media.
2. Web Development with Node and Express, Ethan Brown, O'Reilly Media.
3. C# Programming, Wikibooks Contributors, Platypus Global Media.
4. Python Programming: A Modern Approach, Vamsi Kurama, Pearson.

Reference (s)

1. Pro ASP.NET Core 6, Adam Freeman, 9th edition, Apress.
2. Automate the Boring Stuff with Python, Al Sweigart, No Starch Press, 2nd Edition.

Internal Assessment Pattern

Cognitive Level	Int. Test 1 (%)	Int. Test 2 (%)	Open Book Exam (%)
Remember	30	30	--
Understand	40	40	--
Apply	30	30	50
Analyze	--	--	50
Evaluate	--	--	--
Create	--	--	--
Total (%)	100	100	100

SAMPLE QUESTION (S)

Remember:

1. What are the advantages of Restful API?
2. What is the difference between Node.js and Express.js?
3. Define C# Function?
4. Write a short note about Abstract Base Classes.
5. Write a short note on Browser Storage.

Understand:

1. Explain about request and response structure.
2. What is Restful API? Explain about various Rest principles.
3. Describe the procedure to use middleware in Express.js.
4. Explain in detail about CLR.
5. Discuss about map(), filter(), reduce() in Python.

Apply:

1. Implement a basic Node.js program that reads and writes a file using the fs module.
2. Create a simple HTTP server using the http module that responds with "Welcome to Node.js!".
3. Write a simple C# program to handle an Exception.
4. Write a simple C# code that calculates and displays the sum of two numbers entered by the user.
5. Implement a simple HTTP server in Python that handles GET and POST requests.

Analyze:

1. How does the client-server architecture impact the way web applications handle user requests?
2. Compare the http module in Node.js with Express.js in terms of request handling and ease of use.
3. Compare how Razor pages are different from MVC.
4. Analyze the difference between .NET Framework and .NET Core?
5. Analyze the difference between synchronous and asynchronous HTTP requests in Python.

23MLC31 Fundamentals of Cloud Computing (Cloud Computing Career Path)

3 0 0 3

Course Outcomes

1. Interpret the architecture and infrastructure models of cloud computing, strengths, and limitations of cloud computing.
2. Understand the techniques of virtualization, including virtual machines and virtualized datacenters.
3. Apply design principles and architectural patterns to develop scalable, resilient, and cloud-native applications.
4. Compare and contrast various cloud service provider implementations and deployment models based on performance, cost, and features.
5. Understand the strategies for cloud scaling, load balancing, fault tolerance, and disaster recovery in cloud environments.
6. Identify and assess security, privacy, and risk management issues associated with cloud computing platforms and services.

COs–POs Mapping

COs	PO_1	PO_2	PO_3	PO_4	PO_5	PSO_1	PSO_2
1	3	2	2	1	2	2	1
2	3	3	2	2	3	2	2
3	2	2	3	2	3	3	2
4	3	2	3	2	2	2	2
5	3	3	3	2	2	2	2
6	2	2	2	2	2	2	2

3–Stronglylinked|2–Moderatelylinked|1–Weakly linked

Unit I **12 Hours**

Cloud Computing

Cloud computing: Introduction, SOA, Cloud computing architectures, Value of cloud computing, Cloud Infrastructure models including SaaS, PaaS, IaaS, Storage-as-a-Service, Database-as-a-Service, Information-as-a-Service,Process-as-a-Service,Application-as-a-Service, Integration-as-a-Service, and Security-as-a-Service. Public cloud, private cloud and hybrid cloud, Cloud Services.

Before moving into the cloud: Know Your Software Licenses, The Shift to a Cloud Cost Model. History of Cloud Computing, Advantages of Cloud Computing, and Disadvantages of Cloud Computing.

Companies in the Cloud Today, Amazon Web Services, Google services, IBM Cloud, Windows Azure, Tata Cloud, Salesforce.com

Unit II **13 Hours**

Virtualization & Design

Virtualization,Virtualmachine, Implementation Levels of Virtualization,Virtualization Structures/Tools and Mechanisms, Virtualization of CPU, Memory, and I/O Devices, Virtual Clusters and Resource Management, Datacentre, Virtualization for Data-Centre Automation. Service Levels for Cloud Applications Ready for the cloud: Web Application Design, Machine Image Design, Privacy Design, Database Management.

*Various hypervisors like VMware, KVM,*oracleVM,

Unit III **12 Hours**

Cloud centres, Scaling a Cloud Infrastructure and Disaster Recovery:

Cloud centres in detail: Comparing approaches, Xen, Eucalyptus, CloudStack, and OpenStack. Cloud Scaling:Capacity Planning, Cloud Scale, Types of cloud scaling.

Disaster Recovery:Disasters in cloud,Disaster Recovery Planning, Cloud Disaster Management.

Requirements for modern data centres-high availability and Service Orientated-Infrastructures(SOI).Modern data centre use case studies.

Unit IV **11 Hours**

Cloud Computing Software Security Fundamentals

Data Security, Network Security, Host Security, Compromise Response.

Cloud information Security Objectives, Cloud Security Services, Relevant Cloud Security Design Principles,

Secure Cloud Software Requirements, Approaches to Cloud Software Requirements Engineering, Cloud Security Policy Implementation.
Cloud Computing Risk Issues: The CIA Triad, Privacy and Compliance Risks, Threats to Infrastructure Data and Access Control, Cloud Access Control Issues, Cloud Service Provider Risks.
MicrosoftAzure, IBMBluemix

Total:48 Hours

Textbook(s)

1. George Reese,Cloud Application Architectures,1st Edition O'ReillyMedia,2009
2. RonaldL.Krutzand Russell DeanVines,Cloud Security,1st Edition,WileyPublishing,2010

Reference(s)

1. MichaelMiller,CloudComputing-WebBasedApplicationsthatchangethewayyouworkandcollaborateonline, 1st Edition, Pearson Education, Publishing, 2011
2. KaiHwang,GeoffreyCFoxandJackJ.Dongarra,Distributed&CloudComputingfromParallelProcessingto the Internet ofThings,1st Edition, MK Publishing,2010
3. DavidSLinthicum,CloudComputingandSOAConvergenceinYourEnterprise:AStep-by-StepGuide,1st Edition, Addison-Wesley, 2009

SAMPLEQUESTION(S)

Internal Assessment Pattern

Cognitive Level	Int.Test1 (%)	Int.Test2 (%)	AssignmentTest3(%)
Remember	40	45	20
Understand	40	45	60
Apply	20	10	
Analyze	--	--	20
Evaluate	--	--	--
Create	--	--	--
Total(%)	100	100	100

Remember

1. Define Cloud Computing.
2. List types of virtualization.
3. Define proactive scaling.
4. What is CIAT RAID?

Understand

1. Explain different cloud Infrastructure models.
2. Explain different levels of virtualization.
3. Explain about Recovery Point Objective.
4. With neat diagram explain the functioning of Xen Architecture.
5. List and explain various cloud service providers risks.

Analyze

1. Compare cloud center and service infrastructure.
2. Analyze different cloud services provided by Amazon

23CS004 Principles of Programming Languages

3 0 0 3

Course Outcomes

1. Explain the concepts of programming languages.
2. Describe syntax and semantics of programming languages.
3. Explain data types, and basic statements of programming languages.
4. Illustrate the concept of subprogram.
5. Demonstrate the concepts of ADT and OOP.
6. Summarize the concept of exception handling and event handling.

CO–PO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	1	2	2	-	-	-	-	-	-	-	-	3	-
CO2	3	2	1	2	-	-	-	-	-	-	-	-	3	-
CO3	3	1	3	1	-	-	-	-	-	-	-	-	3	-
CO4	3	2	2	2	-	-	-	-	-	-	-	-	3	-
CO5	3	2	2	3	-	-	-	-	-	-	-	-	3	-
CO6	3	2	2	3	-	-	-	-	-	-	-	-	3	-

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I

Preliminary Concepts **12 Hours**

Reasons for studying concepts of programming languages, programming domains, language evaluation criteria, influences on language design, language categories, language design trade-offs, implementation methods, programming environments, Evolution of Major Programming Languages.

Syntax and Semantics: General problem of describing syntax, formal methods of describing syntax, attribute grammars, describing the meanings of programs.

Unit II

Data types, Expressions and Statements, Control Structure **12 Hours**

Names, Bindings, and Scopes: Introduction, names, variables, concept of binding, scope, scope and lifetime, referencing environments, named constants

Data types: Introduction, primitive, character, string types, user defined ordinal types, array, associative arrays, record, tuple types, list types, union types, pointer and reference types, type checking, strong typing, type equivalence

Expressions and Statements: Arithmetic expressions, overloaded operators, type conversions, relational and Boolean expressions, short- circuit evaluation, assignment statements, mixed-mode assignment

Control Structures: introduction, selection statements, iterative statements, unconditional branching, guarded commands.

Unit III

Subprograms **12 Hours**

Subprograms: Fundamentals of subprograms, design issues for subprograms, local referencing environments, parameter passing methods, parameters that are subprograms, calling subprograms indirectly, overloaded subprograms, generic subprograms, design issues for functions, user defined overloaded operators, closures, co routines

Implementing subprograms: General semantics of calls and returns, implementing simple subprograms, implementing subprograms with stack-dynamic local variables, nested subprograms, blocks, implementing dynamic scoping.

Unit IV

ADT, OOP, Exception and Event Handling **12 Hours**

Abstract Data types: The concept of abstraction, introductions to data abstraction, design issues, language examples, parameterized ADT, encapsulation constructs, naming encapsulations

Object Oriented Programming: Design issues for OOP, OOP in Smalltalk, C++, Java, Ada 95, C#, Implementation of Object-Oriented constructs.

Exception Handling and Event Handling: Introduction, exception handling in Ada, C++, Java, introduction to event handling, event handling with Java and C#.

Total: 48 Hours

Textbook (s)

1. Robert W. Sebesta, “Concepts of Programming Languages”, Tenth Edition, Addison Wesley, 2012.
2. Programming Languages, Principles & Paradigms, 2ed, Allen B Tucker, Robert E Noonan, TMH

Reference (s)

1. R. Kent Dybvig, "The Scheme programming language", Fourth Edition, MIT Press, 2009.
2. Jeffrey D. Ullman, "Elements of ML programming", Second Edition, Prentice Hall, 2098.
3. Richard A. O'Keefe, "The craft of Prolog", MIT Press, 2009.
4. W. F. Clocksin and C. S. Mellish, "Programming in Prolog: Using the ISO Standard", Fifth Edition, Springer, 2003

Internal Assessment Pattern

Cognitive Level	Int. Test 1 (%)	Int. Test 2 (%)	Open Book Exam (%)
Remember	50	40	--
Understand	50	50	--
Apply	--	10	50
Analyze	--	--	50
Evaluate	--	--	--
Create	--	--	--
Total (%)	100	100	100

SAMPLE QUESTION (S)

Remember

1. Define programming language?
2. Define named constants?
3. Define subprogram?

Understand

1. Explain numeric types?
2. Summarize grammars for simple assignment statements.
3. Discuss language evaluation criteria and the characteristics that affect them.

Apply

1. Identity the exceptions occurred in C++ and Ada.
2. Construct BNF notation for following:
    a. For loop
    b. If-else condition
    c. Structure definition
3. Solve the given grammar is ambiguous or not.
   <assign> -><id>=<expr>
   <id>->A|B|C
   <expr>-><expr>+<expr>
   |<expr>*<expr>
   |(<expr>)
   |<id>

23CS005 Mobile Computing

3 0 0 3

Course Outcomes

1. Explain the basic concepts and fundamentals of mobile computing and telecommunication systems along with various standards.
2. Illustrate the techniques, protocols related to GSM and GPRS architecture to perform requirements analysis.
3. Explain major components of Mobile IP to improve the service qualities of a network
4. Compare various ad hoc routing protocols to examine the performance of network
5. Explain the architecture of Wireless Sensor Network and WLAN design issues and limitations.
6. Outline the basic knowledge in developing smart phone applications using various platforms, toolkits, APIs and third party libraries

CO–PO Mapping

CO's	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	2	-	3	-	-	-	-	2	-	2	-	2	-	2
CO2	2	-	2	-	-	-	-	2	-	2	-	2	-	3
CO3	2	-	1	-	-	-	-	2	-	2	-	2	-	3
CO4	2	-	2	-	-	-	-	2	-	2	-	2	-	3
CO5	2	-	2	-	-	-	-	2	-	3	-	2	-	3
CO6	2	-	2	-	-	-	-	2	-	3	-	2	-	3

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I

Introduction to Mobile Communications **12 Hours**

Overview:Mobile Communication and Mobile Computing – Architecture; Generations of Telecommunications – 1G, 2G, 3G, 4G and 5G; Wireless Technologies – WPAN, WLAN, WMAN; Wireless Medium Access Control (MAC) – Overview, SDMA, TDMA, FDMA, CDMA;

2G Technologies: GSM – Services, Architecture, Radio Interface, Protocol Stack, Localization, Call Handling, Handover, Security; 2.5G – GPRS and 3G – EDGE.

Networking: Comunication Modes, Basic Network Designs, Cellular Infrastrures

Unit II

Mobile Network Layer and Transport Layer **12 Hours**

Mobile Network Layer: Mobile IP Overview, IP Packet delivery, Agent discovery and advertisement, Registration, Tunneling and Encapsulation, Optimizations, Security, and Dynamic Host Configuration Protocol (DHCP).

Mobile Transport Layer – Motivation, Traditional TCP, Classical TCP Approaches: Indirect TCP, Snooping TCP, Mobile TCP, Transaction-oriented TCP; Optimizations, TCP for 2.5G/3G.

Multi Task gadget: wide area mobile data - air link standards for data - wireless application environment

Unit III

Mobile Ad-hoc Network (MANET) **12 Hours**

Introduction to Mobile ad-hoc networks, Characteristics and features, Applications, Limitations; Routing protocols – Design Issues, Routing algorithms: Proactive (DSDV & OLSR) and Reactive (DSR & AODV), Security in ad hoc networks; Wireless LAN – IEEE 802.11 – System Architecture, Protocol Layers.

Wireless Sensor Network (WSN): Introduction, Architecture, Applications, Properties and Security.

Satellite systems: history - applications - basics - broadcast systems

Unit IV

Mobile Platforms and Applications **12 Hours**

Mobile OS: Overview on Mobile Device Operation Systems (Android, iOS, Black Berry) – Architecture, App development Kit; Introduction to Network Simulators: Charactestics, Applications, Limitations, Types of Simulators: Wireless Application Protocol (WAP): Introduction, Architecture, Applications.

Application layer Protocols – FTP, SMTP, HTTP, DNS; Windows 10.

Total: 48 Hours

Textbook (s)

1. Raj Kamal, Mobile Computing, Oxford press, Third Edition, 2018
2. Jochen Schiller, Mobile Communications, Pearson Eduction, Secon Edition, 2020

Reference (s)

1. Asoke K Talukder, Hasan Ahmad and RoopaYavagal, Mobile Computing, Second Edition, McGraw Hill, 2010
2. Prasant Kumar Pattnail and Rajib Mall, Fundamentals of Mobile Computing, Second Edition, PHI Learning Pvt. Ltd., 2015
3. Frank Adelstein, et al., Fundamentals of Mobile and Pervasive Computing, McGraw Hill, 2005
4. http://www.isi.edu/nsnam/ns/doc/ns_doc.pdf (NS2 manual)

Internal Assessment Pattern

Cognitive Level	Int. Test 1 (%)	Int. Test 2 (%)	Open Book Exam (%)
Remember	30	25	10
Understand	25	15	20
Apply	25	20	30
Analyze	20	20	30
Evaluate	--	10	10
Create	--	10	--
Total (%)	100	100	100

SAMPLE QUESTION (S)

Remember

1. Define Mobile Computing
2. List the applications of mobile computing
3. List the limitations of mobile computing
4. Define Mobile Adhoc Network
5. List the MANET routing issues

Understand

1. Explain mobile computing with architecture
2. Explain tTDMA and FDMA with neat diagrams
3. Illustrate the different services provided by the GSM, Explain with system architecture
4. Explain IP Packet delivery with diagram
5. Describe DSSM (Direct Sequence Spread Spectrum)

Apply

1. Illustarte Registration process achieved in mobile network layer
2. Discuss the feature of tunneling and encapsulation
3. Illustrate the function of Dynamic Host Configuration Protocol (DHCP)
4. Discuss about security in Ad-hoc network
5. Illustarte the properties of MANETS

Analyze

1. Compare Snooping TCP and Indirect TCP
2. Analyze the performance of Dynamic Source Routing protocol
3. Compare about iOS and Blackberry Operating Systems
4. Compare NS2 and NS3 simulators
5. Explain about Android Operating System with architecture

Evaluate

1. Evaluate the performance of Wireless Datagram protocol
2. Evaluate the performance of DSDV
3. Evaluate the performance of DSR
4. How Traditional TCP is different from I-TCP
5. How localization and Call handling is performed

Open Book Exam Questions

Q1. Describe the functions of the MS and SIM. Why does GSM separate the MS and SIM? How and where is user-related data represented/stored in the GSM system? How is user data protected from unauthorized access, especially

over the air interface? How could the position of an MS (not only the current BTS) be localized? Think of the MS reports regarding signal quality.

Q2. Describe the Concept of GPRS. Using the best delay class in GPRS and a data rate of 115.2 kbit/s – how many bytes are in transit before a first acknowledgement from the receiver could reach the sender (neglect further delays in the fixed network and receiver system)? Now think of typical web transfer with 10 Kbyte average transmission size – how would a standard TCP behave on top of GPRS (see chapters 9 and 10)? Think of congestion avoidance and its relation to the round-trip time. What changes are needed?

23CS006 Distributed Operating Systems

3 0 0 3

Course Outcomes

1. Summarize the fundamental concepts of Distributed Operating Systems.
2. Illustrate the Concepts of Message passing system and Remote Procedure Calls.
3. Design and Implementation of Distributed Shared Memory and Structures.
4. Utilize the Synchronization and Distributed File Systems.
5. Make use of Resource Management and Process Management Concepts.
6. Outline the concept of Naming and Security in Distributed Operating Systems.

CO–PO Mapping

CO's	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	2	-	-	-	-	-	-	-	-	1	1	2
CO2	3	2	2	-	-	-	-	-	-	-	-	1	2	2
CO3	3	2	2	-	-	-	-	-	-	-	-	1	2	2
CO4	2	2	2	-	-	-	-	-	-	-	-	1	2	2
CO5	3	2	2	-	-	-	-	-	-	-	-	1	2	2
CO6	3	1	2	-	-	-	-	-	-	-	-	1	2	2

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I

Introduction and Communication in Distributed System **12 Hours**

Introduction: Design Issues - Distributed Computing Environment - Message Passing - Features of Good Message Passing System-Issues in IPC by Message Passing-Synchronization- Buffering

Remote Procedure Calls: The RPC Model - Transparency of RPC- Implementing RPC Mechanism - Stub Generation - RPC Messages - Marshaling Arguments and Results - Server Management - Parameter-Passing Semantics- Call Semantics

Communication Protocols for RPCs - Complicated RPCs - Client-Server Binding-Case Study: Sun RPC

Unit II

Distributed Shared Memory and Synchronization **12 Hours**

Distributed Shared Memory: General Architecture of DSM Systems- Design and Implementation Issues of DSM – Granularity - Structure of Shared Memory Space

Synchronization: Clock Synchronization - Event Ordering - Mutual Exclusion –Deadlocks- Election Algorithms

Consistency Models-Replacement Strategy - Thrashing

Unit III

Resource Management and Distributed FileSystems **12 Hours**

Resource Management: Features of a Good Global Scheduling Algorithm - Task Assignment Approach – Load Balancing Approach – Load Sharing Approach

Distributed File Systems: Introduction -Desirable Features of a Good Distributed File System - File Models – File Accessing Models – File Sharing Semantics – File Caching Schemes - File Replication

Process Management - Process Migration- Threads- Case Study: DCE Distributed File Service

Unit IV

Naming and Security **12 Hours**

Naming: Desirable Features of a Good Naming System - Fundamental Terminologies and Concepts - System-Oriented Names - Object-Locating Mechanisms - Human-Oriented Names - Name Caches

Security: Potential Attacks to Computer Systems - Cryptography – Authentication

Access Control - Digital Signatures - Design Principles- Case Study: DCE Directory Service

Total: 48 Hours

Textbook (s)

1. Pradeep K Sinha, "Distributed Operating Systems: Concepts and Design", Prentice Hall of India, 2009
2. Andrew S Tannebaum, "Distributed Operating Systems", Pearson Education, 2007

Reference (s)

1. George Coulouris, Jean Dollimore and Tim Kindberg, "Distributed Systems Concepts and Design", Fifth Edition, Pearson Education, 2012
2. Tanenbaum A.S., Van Steen M., "Distributed Systems: Principles and Paradigms", Pearson Education, 2007
3. Liu M.L., "Distributed Computing, Principles and Applications", Pearson Education, 2004.
4. Nancy A Lynch, "Distributed Algorithms", Morgan Kaufman Publishers, USA, 2003

Internal Assessment Pattern

Cognitive Level	Int. Test 1 (%)	Int. Test 2 (%)	Open Book Exam (%)
Remember	40	40	--
Understand	30	40	--
Apply	30	20	50
Analyze	--	--	50
Evaluate	--	--	--
Create	--	--	--
Total (%)	100	100	100

SAMPLE QUESTION (S)

Remember

1. List any four advantages of Distributed operating systems
2. Define distributed operating systems.
3. List any five reasons why to build distributed operating System.
4. What are the features of good distributed file systems?
5. List the features of a Good Naming System

Understand

1. Explain the design issues of distributed operating systems.
2. Explain the role of Naming in distributed operating systems
3. Summarize the internal and external synchronization of Physical clocks
4. Differentiate Resource Management and Process Management
5. Explain in detail about security and authentication.

Apply

1. Construct the different protocols of RPC.
2. Identify the File shearing Semantics and file caching Schemes.
3. Develop the Bully and Ring Algorithm
4. Develop the scenario how might the clocks in two computers that are linked by a local network be synchronized without reference to an external time source
5. Utilize the various deadlock prevention techniques with suitable example?

Analyze

1. Analyze the Concept of exception handling in RPC.
2. Examine the Concept of Thread.
3. Compare Fault tolerance and Thrashing.
4. Classify the deadlock avoidance techniques.
5. Simplify an algorithm using multicast and logical clocks for mutual exclusion.

Open Book Exam Questions

Question 1:

A distributed system has DSM facility. The process-scheduling mechanism of this system selects another process to run when a fault occurs for the currently running process, and theCPU is utilized while the block is being fetched. Two system engineers arguing about how to better utilize the CPUs of this system have the following opinions:

(a) The first one says that if a large number of processes are scheduled for execution at anode, the available memory space of the node can be distributed among these processesso that almost always there will be a ready process to run when a page fault occurs. Thus, CPU utilization can be kept high.

(b) The second one says that if only a few processes are scheduled for execution at a node, the available memory space of the node can be allocated to each of the few processes, and each process will produce fewer page faults. Thus, CPU utilization can be kept high. Whose argument is correct? Give reasons for your answer.

Question 2:
The password mechanism is used in a distributed system to authenticate users at login time. State the most suitable locations (according to you) for storing the login program and the password file in the following cases:

(a) The distributed system is based on the workstation-server model with each workstation having a small hard disk of about 20 megabytes capacity.

(b) The distributed system is based on the workstation-server model. Some of the workstations are diskless and others have a small hard disk of about 20 megabytes capacity.
(c) The distributed system is based on the processor-pool model.

23ML507 Neural Networks Lab

0 0 3 1.5

Course Outcomes

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

1. Build and test basic neuron models and activation functions to perform logic operations and understand how simple neurons work.
2. Train neural networks with one or more layers using learning methods like Perceptron and Backpropagation.
3. Check how well neural networks learn by using error correction and testing with different training samples.
4. Create memory-based neural networks like Hopfield and BAM to store and recall patterns.
5. Use unsupervised learning models like SOM, LVQ, and ART1 to group or cluster similar data without labels.
6. Use neural networks in real-world tasks such as recognizing images or classifying signals using real datasets.

CO–PO Mapping

COs	PO1	PO2	PO4	PO5	PO8	PSO1	PSO2
1	2	2	3	2	2	2	1
2	3	2	3	3	2	2	2
3	2	2	3	3	2	3	2
4	2	2	2	2	3	3	3
5	2	2	2	2	2	3	3
6	2	2	1	1	3	3	3

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

List of Experiments

1. Implement the following types of Activation Functions in neural network.
    a. Binary Step Activation Function
    b. Linear Activation Function
    c. Sigmoid Activation Function
    d. Tanh Activation Function
    e. RELU Activation Function
    f. Softmax Activation Function
2. Create a basic Neuron Model to implement basic logic gates like AND, OR, and NOT
3. Implementation of McCulloch Pitts Neuron Model.
4. Implementation of Hebbian Learning Algorithm.
5. Implement the Single-layer perceptron network and test the same using appropriate dataset.
6. Implement the multi-layer perceptron network and test the same using the appropriate dataset.
7. Implement Back propagation algorithm in neural network.
8. Implement K-fold cross validation on any dataset.
9. Implement Auto-Associative Memory Network
10. Implement Hetero-Associative Memory (One-to-One Mapping)
11. Simulate Bidirectional Associative Memory (BAM)
12. Implement Hopfield Network for Pattern Storage and Retrieval
13. Implement Kohonen Self-Organizing Map (SOM)

14. Implement Learning Vector Quantization (LVQ)
15. Simulate Adaptive Resonance Theory (ART1)
16. Mini-Project: Character/Image Classification using MLP or CNN

Lab Manual

1. Neural Networks Laboratory– AIML Department, GMRIT.

Textbook (s)

1. Simon Haykin, "Neural Networks - A comprehensive foundation", Pearson Education, 2003.
2. Bengio, Yoshua, Ian J. Goodfellow, and Aaron Courville. "Deep Learning." An MIT Press book in preparation,2015.

Reference (s)

1. SatishKumar,"Neural Networks: A Classroom Approach" Tata McGraw Hill Education, 2004.
2. Josh Patterson and Adam Gibson "Deep Learning A Practitioner's Approach" O'Reilly Media, Inc. 2017

23TPX01 Term Paper

0 0 3 1.5

Course Outcomes

At the end of the course, students will be able to

1. Interpret the li**t**erature to link the earlier research with the contemporary technologies
2. Communicate effectively as an individual to present ideas clearly and coherently
3. Review the research findings and its correlation to the latest applications
4. Prepare documents and present the concepts clearly and coherently
5. Inculcate the spirit of enquiry for self-learning
6. Identify interdisciplinary oriented topics

COs – POs Mapping

CO's	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	-	3	2	3	-	3	2	3	-	3	2	2
CO2	2	2	-	3	3	3	-	2	3	3	-	3	2	2
CO3	3	3	-	3	2	3	-	3	2	3	-	3	2	2
CO4	3	3	-	3	2	3	-	3	3	3	-	3	2	2
CO5	2	2	-	3	3	3	-	2	2	3	-	3	2	2
CO6	3	2	-	3	1	3	-	3	3	3	-	3	2	2

3–Strongly linked | 2–Moderately linked| 1–Weakly linked

Term Paper: The Term Paper is a self-study report and shall be carried out either during 5th or 6th semester in choice with Mini Project. Every student will take up this term paper individually and submit a report. The scope of the term paper could be an exhaustive literature review choosing any engineering concept with reference to standard research papers or an extension of the concept of earlier course work in consultation with the term paper supervisor. The report will be evaluated by a committee as nominated by HoD with the approval of CoE

23ESX02 Employability Skills II

0 0 2 0

Course Outcomes

1. Demonstrate oral communication and writing skills as an individual to present ideas coherently
2. Develop life skills with behavioral etiquettes and personal grooming.
3. Assess analytical and aptitude skills.
4. Develop algorithms for engineering applications.
5. Solve engineering problems using software.
6. Utilize simulation tools for testing

COs –POs Mapping

CO's	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	1	1	1	1	1	-	-	-	2	3	3	3	-	-
CO2	1	1	3	1	1	-	-	-	2	3	3	3	-	-
CO3	1	1	3	1	1	-	-	-	2	2	3	3	-	-
CO4	2	2	3	2	2	-	-	-	3	3	2	3	-	-
CO5	3	2	3	2	2	-	-	-	3	2	2	3	-	-
CO6	2	2	2	2	3	-	-	-	3	2	3	3	-	-

3–Strongly linked | 2–Moderately linked| 1–Weakly linked

Soft Skills:

Sl No.	**5th Semester (Topic & Content)**	**No. of Periods**
1.	**Introduction to Campus Placements:** Stages of Campus Placement, Skills assessed in Campus Placements & How to get ready?	01
2.	**Motivational Talk on Positive Thinking:** Beliefs, Thoughts, Actions, Habits & Results (Success)	01
3.	**Resume Preparation:** Resume? Templates? Mistakes to be avoided in a Resume, Steps to be followed in preparing it.(with examples)	02
4.	**Group Discussions** (Recap)**:** GD? Stages of a GD, Skills assessed in a GD, Blunders to be avoided, How to excel in a GD? (through Practice Sessions)	02
5.	**Psychometric Tests:** Definition, Types of Psychometric Tests: Numerical Computation, Data Interpretation, Verbal Comprehension, Verbal Critical Reasoning and Personality Questionnaires	01
6.	**Exercises related to Communication:** Story Writing, TAT etc	01
	Total Periods	**08**

Quantitative Aptitude:

Sl No.	**Semester-V (Topic)**	**No. of Periods**
1.	Square &Cube roots	01
2.	Partnership	01
3.	Partnership	01
4.	Logarithms	01
5.	Progressions	01
6.	Mensuration	01
7.	Mensuration	01
8.	Data Sufficiency	01
	Total Periods	**08**

Domain Specific: (Comprehensive Coding Skills- Phase 1) 16 Hours

Module-1: Arrays, Bit Manipulation, Hashing, Searching & Recursion (Weeks 1–8)

1. Time & Space Complexity, Brute Force to Optimization
   Problems: Two Sum, Best Time to Buy/Sell Stock, Move Zeroes
2. Two Pointer Techniques
   Problems: Container with Most Water, Longest Substring without Repeating Characters
3. Sliding Window and Prefix Sum techniques
   Problems: Maximum Average Subarray I, Subarray Sum Equals K, Minimum Size Subarray Sum
4. Binary Search Variants
   Concepts: Lower/Upper Bound, Binary Search on Answer
   Problems: Search Insert Position, Koko Eating Bananas
5. Bit Manipulation Basics
   Problems: Single Number, Counting Bits, Bitwise AND Range
6. Hashing Techniques
   Problems: Group Anagrams, Top K Frequent Elements
7. Sorting with Logic and Custom Comparators
   Concepts: Custom Sort, Comparator Functions, Frequency-Based Sorting
   Problems: Merge Intervals, Sort Colors, Largest Number
8. Recursion & Backtracking Basics
   Problems: Subsets, Permutations, Generate Parentheses

Module-2: Stack, Heap, Trees, Graphs (Weeks 9–16)

1. Stacks & Monotonic Stack
   Problems: Valid Parentheses, Daily Temperatures, Largest Rectangle in Histogram
2. Heaps & Priority Queues
   Problems: Kth Largest in Array, Merge K Sorted Lists
3. Trees – Traversals and Properties
   Problems: Symmetric Tree, Diameter of Binary Tree, Level Order Traversal
4. Binary Search Tree – Insert/Search/LCA(Lowest Common Ancestor)
   Problems: Kth Smallest, Lowest Common Ancestor
5. Tree Serialization & View Problems
   Problems: Serialize/ Deserialize, Right Side View, Zigzag Traversal
6. Graphs – DFS/BFS in Grids & Adjacency List
   Problems: Number of Islands, Number of Provinces, Flood Fill
7. Graphs – Applications
   Concepts: Topological Sort, Cloning
   Problems: Course Schedule, Clone Graph, Word Ladder
8. Greedy Techniques and Introduction to Dynamic Programming(DP)
   Problems: Jump Game, Trapping Rain Water, House Robber (DP Problem)

References:

1. https://leetcode.com
2. https://www.geeksforgeeks.org/
3. https://www.codechef.com
4. https://www.hackerearth.com
5. https://www.interviewbit.com
6. https://www.hackerrank.com

23SIX01 Summer Internship I

0 0 0 1

Course Outcomes

At the end of the course, students will be able to

1. Demonstrate the application of knowledge and skill sets acquired from the course and workplace in the assigned job function/s
2. Solve real life challenges in the workplace by analyzing work environment and conditions, and selecting appropriate skill sets acquired from the course
3. Articulate career options by considering opportunities in company, sector, industry, professional and educational advancement
4. Communicate and collaborate effectively and appropriately with different professionals in the work environment through written and oral means
5. Demonstrate the ability to harness resources by examining challenges and considering opportunities
6. Demonstrate appreciation and respect for diverse groups of professionals by engaging harmoniously with different company stakeholders

COs – POs Mapping

CO's	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	2	1	2	1	2	3	1	3	3	3	3	3	3	3
CO2	3	3	3	3	2	2	1	3	3	1	1	3	3	1
CO3	3	2	3	2	2	1	1	3	2	3	3	2	3	3
CO4	3	3	3	3	3	3	2	3	3	3	3	3	3	3
CO5	3	3	3	3	3	3	3	3	3	3	3	3	3	3
CO6	3	3	3	3	3	3	3	2	3	3	3	3	3	3

Summer Internship: As a part of curriculum in all branches of Engineering, it is mandatory for all students to undergo summer internship Programme at industries (core or allied) / R & D organization to get practical insight of their subject domain during summer break after the 4th semester. This summer internship Programme shall be availed to a maximum duration of 4 weeks and the assessment shall be carried out with both internal and external experts leading to "Satisfactory" and "Non-Satisfactory Performance", and it will not be accounted for the calculation of CGPA.

23ATX-- AUDIT COURSE

0 0 0 0

Course Outcomes

At the end of the Audit Course the students will be able to

1. Interpret the meaning of values and select their goals by self- Investigation based on personal values.
2. Interpret the major events and issues related to a period in Indian history.
3. Assess the benefits and limitations of science and its application in technological developments towards human welfare
4. Check the awareness regarding basic human rights and to uphold the dignity of every individual.
5. Assess the individual and group behaviour and understand the implications of organizational behaviour on the process of management.
6. Determine the appropriateness of various leadership styles and conflict management strategies used in organizations.

COs –POs Mapping

CO's	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	-	-	-	-	-	2	2	3	1	2	-	2	-	-
CO2	-	-	-	-	-	3	2	1	1	1	-	1	-	-
CO3	-	-	-	-	-	2	3	1	1	1	-	2	-	-
CO4	-	-	-	-	-	3	2	3	1	2	-	2	-	-
CO5	-	-	-	-	-	2	2	1	3	2	-	1	-	-
CO6	-	-	-	-	-	2	2	3	1	2	-	2	-	-

3–Strongly linked | 2–Moderately linked| 1–Weakly linked

Audit Courses: Audit courses are among the compulsory courses and do not carry any credits. All the students shall register for one Audit courses in the beginning of 3rd semester. List of the courses will be notified at the beginning of the third semester for all students and the student has to choose one audit course under self-study mode at the beginning of third semester. All the students (regular and lateral entry students) shall complete the audit course similar to other regular courses and the results will be indicated with "Satisfactory" or "Not Satisfactory" performance.

6th Semester

23ML601 Deep Learning Techniques

3 0 0 3

Course Outcomes

1. Explain the fundamentals of deep learning and Deep Neural networks
2. Give Differences between MLP and Deep Neural Networks.
3. Make use of different CNN models.
4. Compare different character encoding techniques and Recurrent Neural networks.
5. Exemplify different Sequence models and Deep unsupervised models.
6. Examine different deep learning applications**.**

CO–PO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2
1	3	2	2	1	3	2	2
2	3	3	2	1	2	3	3
3	3	3	3	2	3	3	3
4	3	2	3	2	3	3	3
5	3	2	2	2	2	3	3
6	3	3	3	3	3	3	3

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I **12 Hours**

Unit I

Introduction Deep Learning: Introduction, Need of Deep learning, Applications of Deep Learning, Relu function, types of Relu function, Model Parameters vs Hyper parameters.

Deep neural networks (DNNs): Difficulty of training DNNs, Greedy layer wise training, Optimization for training DNNs, Optimization methods for neural networks (AdaGrad, RMSProp, Adam), Second order methods for training, Regularization methods (dropout, drop connect, batch normalization).

Types of learning laws, limitation of Deep Learning.

Unit II **12 Hours**

Convolutional networks: Building blocks of CNNs, Filters and Feature Maps, Pooling layers, Convolutions over volumes, Softmax regression, hyper-parameter tuning.
CNN models: Lenet, Alex Net, VGG –16, Residual Networks, YOLO.
Transfer Learning: introduction, types of transfer learning, transfer learning with CNN models.

Linear Least Square Filters, Least Mean Square Algorithm, XOR Problem

Unit III **12 Hours**

Recurrent Networks: One-hot encoding of words and characters, using word embeddings, Recurrent Neural Network Model, vanishing gradients with RNNs, Gated Recurrent Unit (GRU) LSTM (long short-term memory).

Keras ,tensorflow, data augmentation, Batch Normalization, Dropout

Unit IV **12 Hours**

Sequence Models: Encoder Decoder sequence to sequence architectures
Deep Unsupervised Learning: Auto encoders, Variational Auto encoders, Generative Adversarial Network, Deep Boltzmann Machines.
Case study: Image segmentation, object detection,

Total: 48 Hours

Textbook (s)

1. Bengio, Yoshua, Ian J. Goodfellow, and Aaron Courville. "Deep Learning." An MIT Press book in preparation. (2015).
2. S.N. Sivanandam, S.N. Deepa. “Principals of Soft Computing”, An WILLY publications 3rd edition 2009.

3. Fundamentals of Deep Learning: Designing Next-generation Machine Intelligence Algorithms by Nicholas Locascio and Nikhil Buduma O'Reilly Media; 1 edition (June 29, 2017)
4. Simon S. Haykin, Neural Networks, Prentice Hall, 2nd edition
5. B. Yegnanrayana , "Artificial Neural Networks" , PHI.

Reference (s)

1. Franchois Chollet, Deep Learning with Python
2. Deep Learning: A Practitioner's Approach by Adam Gibson and Josh Patterson Shroff/O'Reilly; First edition (2017)
3. Python Deep Learning by Daniel Slater and Gianmario Spacagna, Packt Publishing; 2/e (January 16, 2019)
4. Bishop, C., M., Pattern Recognition and Machine Learning, Springer, 2006
5. Kevin P. Murphy.,Machine Learning:A Probabilistic Perspective

Internal Assessment Pattern

Cognitive Level	Int. Test 1 (%)	Int. Test 2 (%)	Open Book Exam (%)
Remember	20	20	--
Understand	50	50	--
Apply	30	30	50
Analyze	--	--	50
Evaluate	--	--	--
Create	--	--	--
Total (%)	100	100	100

SAMPLE QUESTION (S)

Remember

1. What is neuron.
2. List any two learning rules.
3. Define Convolution neural network
4. Define one-hot encoding technique
5. Define LSTM

Understand

1. Explain Multilayer perceptron.
2. Explain data representation through tensors.
3. Outline different character encoding techniques.
4. Illustrate usage of autoencoders
5. Construct Gated Recurrent Unit (GRU)

Apply

1. Apply 2D tensors to represent image data
2. Develop CNN model to classify digits.
3. Build a CNN model to recognize images
4. Apply LSTM to extract text from a video
5. Build an appropriate DL model for image segmentation

Analyze

1. Examine differences between ANN and CNN.
2. Compare ML with DL
3. Compare and contract different activation functions.

Evaluate

1. Evaluate performance of DL and ML for an image classification problem.

Open Book Exam Questions

Question 1:

List out any four datasets avaialbe in keras API and explain the features of each dataset and take handwritten "**mnist**" numeric numbers dataset from keras API to classify handwritten digits.

Question 2:

Take a dataset which consist student photos and apply appropriate CNN models to disply name of each student or object which appears on image. Note: Each image in a dataset consists group of mixed objects.

23ML602 AUTOMATA THEORY AND LANGUAGE PROCESSORS

3 0 0 3

Course Outcomes

1. Understand State machines, languages and computations
2. Understand regular grammars and regular languages
3. Design Pushdown automata for Context free languages
4. Acquire knowledge of compiler & its phases
5. Construct parse table for a given context free grammar
6. Apply code optimization techniques to improve the performance of a program

CO–PO Mapping

COs	PO1	PO2	PO3	PO5	PO12
1	3	2	2	1	2
2	3	3	2	--	1
3	3	3	3	1	2
4	2	3	1	2	2
5	3	2	2	2	2
6	3	3	3	3	3

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I **12 Hours**

Finite Automata: Finite automata model-Deterministic Finite Automata – Nondeterministic Finite automata - Recognition of a language by an Automaton - Equivalence of DFA and NFA, Minimization of Finite Automata - **Regular sets:** Regular Expressions-Arden's theorem –Pumping Lemma for Regular Languages - Closure Properties of Regular sets.

Applications of finite automata, Equivalence of finite automata.

Unit II **12 Hours**

Context free language- Chomsky Classification of languages- Context-free grammar -simplification of Context free Grammar- Chomsky Normal Form-Closure Properties of Context free Languages.

Pushdown Automata: PDA Model-Design-Acceptance by empty stack and final state- Non-deterministic PDA.

Context Sensitive Languages, Linear bounded automata

Unit III **12 Hours**

Language Processing-Phases of a Compiler-Top-down Parsing: Derivation of a string- ambiguous grammar-Left recursion-Left factoring-First and Follow-Non-Recursive Predictive Parsing- LL(1) Grammar-Bottom-up parsing: Model of an LR Parsers-Construction of SLR Parsing Table.

Shift-Reduce parsing – Regular definitions

Unit IV **12 Hours**

Intermediate Code generation and optimization: Three Address Code- Quadruples-Triples-Indirect Triples-Machine Independent Code Optimization: Common Sub-expression Elimination-Constant Folding-Copy Propagation-Dead Code Elimination-Strength Reduction-Loop Optimization -Basic Blocks-Flow Graph-DAG for basic Blocks-Machine Dependent Code Optimization: Peephole Optimization

Syntax directed translation – DAG for expressions **Total:48 Hours**

Text Books:

1. J. E. Hopcroft and J. D. Ullman, Introduction to Automata Theory, Languages and Computation, 3rd Edition, Pearson/Addison Wesley
2. Mishra & Chandra Sekharan, Theory of Computer Science& Automata Language and Computation, 3rd Edition, Prentice Hall of India
3. Alfred V Aho, Monica S Lam, Ravi Sethi, Jeffrey D. Ullman, Compilers, Principles Techniques an Tools,2nd Edition, Pearson.

Reference books:

1. P. Linz, Introduction to Formal Language and Computation, 2nd Edition, Narosa, 2006.
2. V. Raghavan, Principles of Compiler Design, 2nd Edition, TMH, 2011.

Internal Assessment Pattern

Cognitive Level	Int. Test 1 (%)	Int. Test 2 (%)	Open Book Exam (%)
Remember	20	20	--
Understand	30	40	--
Apply	50	40	60
Analyze			40
Evaluate	--	--	--
Create	--	--	--
Total (%)	100	100	100

SAMPLE QUESTION (S)

Remember

1. Write the analytical representation of NFA.
2. Define context free grammar.
3. Define token.

Understand

1. Explain the closure properties of regular languages.
2. Explain the phases of compiler.
3. Explain about code optimization techniques.

Apply

1. Design pushdown automata for L={$a^n b^n$/n>=1}.
2. Calculate the first and follow functions for the given grammar-
   S → (L) / a
   L → SL'
   L' → ,SL' / ∈
1. Construct SLR(1) parsing table for the following grammar
   S' → S
   S → C C
   C → c C
   C → d

Analyse

1. Differentiate Chomsky classification of languages.
2. How does machine dependent code optimization is different from machine independent code optimization?

Open Book Exam Questions

1. Suppose r1 and r2 are regular expressions over the same alphabet Σ. We say r1 = r2 to denote equality of the languages represented by r1 and r2. In other words, every string in the language represented by r1 is also included in the language represented by r2 and vice versa. For each of the following pairs of regular expressions over Σ = {0, 1}, either prove that they represent the same language, or give a string that is present in the language of one but not in the language of the other. In the latter case, you must also describe why your solution string is in the language of one regular expression, but not in that of the other.
   (a) r1 = 1 ∗ (1 + 0) ∗0 ∗ and r2 = (0 ∗1 ∗) ∗
   (b) r1 = ((0 + 1) ∗0) ∗0 and r2 = (0 + 1) ∗0 ∗0
   (c) r1 = (0 + 1) ∗01(0 + 1) ∗ and r2 = 1 ∗ (0 + 1) ∗0(0 + 1) ∗1

2. Construct the basic block and flow graph for the following C fragments

```
 a=3;
b=4;
for(i=0;i<n;i++)
   {
          a=a*b+5;
          a=a-4;
 }
c=a+b*3;
```

23CS603 Software Engineering

3 0 0 3

Course Outcomes

1. Explain the need of Software Life Cycle Models
2. Build end-user requirements into system and software requirements,
3. Summarize the system models of software engineering
4. Identify and apply appropriate software architectures and patterns to carry out high level design
5. Choose various testing techniques during software development
6. Categorize Risk management and Software quality for software products

CO–PO Mapping

CO's	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	2	-	2	-	-	2	-	-	2	-	3	-
CO2	3	2	3	-	2	-	-	2	-	-	3	-	3	-
CO3	3	2	3	-	3	-	-	-	-	-	3	-	3	-
CO4	3	2	3	-	2	-	-	2	-	-	3	-	3	-
CO5	3	2	3	-	2	-	-	1	-	-	3	-	3	-
CO6	3	2	2	-	2	-	-	1	-	-	3	-	3	-

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

UNIT – I **12 Hours**

Introduction to Software Engineering and SDLC, Software Myths, CMMI, Process models: Linear Sequential model, Prototyping model, Evolutionary models: Spiral model, Agile developmental methodologies-Scrum & XP

Incremental model, software development: Product based and application based

Unit II **12 Hours**

Software Requirements: Functional and non-functional requirements, user requirements, system requirements, interface specification.

Software Requirements Engineering Process, Feasibility studies, Requirements elicitation and analysis, requirements validation.

System models: Context models, behavioral models, data models, object models.

Structure of Software Requirements Document, Structured analysis methods

Unit III **12 Hours**

Design concepts, data design, software architecture, Architectural styles and patterns, User interface design - Golden rules, User interface analysis and design and steps. Conceptual model of UML, basic structural modeling, Sattic and Dynamic UML diagrams: class diagrams, sequence diagrams, collaboration diagrams, use case diagrams, etc.,

Data Acquisition System - Monitoring and Control System

Unit IV **12 Hours**

Testing strategies and Risk Management: Testing levels: Unit testing, integration testing, system testing – alpha and beta testing, Testing Types: black box and white box testing techniques, Cyclomatic Complexity, debugging, Risk management - Risk types, strategies, estimation and Planning. Software Quality - Quality assurance and its techniques

Software measurement, metrics for software quality

Total: 48 Hours

Textbook (s)

1. Roger S. Pressman, Software Engineering, A practitioner's Approach, 8th Edition, McGraw-Hill International Edition, 2015
2. I. Sommerville, Software Engineering, 7thEdition, Pearson education, 2004.
3. Rajib Mal, Fundamentals of software Engineering, 4th Edition, Eastern Economy Edition, 2014.

Reference(s)

1. K. K. Aggarwal and Yogesh Singh, Software engineering,3rd Edition, New Age InternationalPublication,2008

Internal Assessment Pattern

Cognitive Level	Int. Test 1 (%)	Int. Test 2 (%)	Open Book Exam (%)
Remember	30	30	--
Understand	40	40	--
Apply	30	30	50
Analyze	--	--	50
Evaluate	--	--	--
Create	--	--	
Total (%)	100	100	100

SAMPLE QUESTION (S)

Remember

1. Define software engineering.
2. List different types software myths.

Understand

1. Describe software architecture styles and patterns .
2. Illustrate golden rules for user interface design.

Apply

1. Applying the process of requirement analysis, discuss how the requirements can be collected for a project.
2. Applying debugging strategy fin an error from a code?

Analyze

1. Compare and Contrast software life cycle models.
2. Analyze risk types in the risk management.

Open Book Exam Questions

Assume that 10 errors have been introduced in the requirements model and that each error will be amplified by a factor of 2:1 into design and an addition 20 design errors are introduced and then amplified 1.5:1 into code where an additional 30 errors are introduced. Assume further that all unit testing will find 30 percent of all errors, integration will fi nd 30 percent of the remaining errors, and validation tests will fi nd 50 percent of the remaining errors. No reviews are conducted. How many errors will be released to the field.

23MLC12 Machine Learning for Business Intelligence

3 0 2 4

Course Outcomes

1. Understand the basic concepts of business analytics.
2. Identify the application of business analytics and use tools to analyze business data.
3. Apply data mining techniques to find useful patterns in large data and help make better business decisions.
4. Become familiar with various metrics, measures used in business analytics.
5. Illustrate various descriptive, predictive, and prescriptive methods and techniques.
6. Model the business data using various business analytical methods and techniques.

CO–PO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2
1	3	1	2	1	1	2	2
2	3	1	3	1	3	3	3
3	3	3	1	2	3	2	2
4	1	3	1	3	3	2	2
5	1	1	3	3	3	3	3
6	1	2	2	1	3	3	3

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

UNIT I **11+ 9 Hours**

Introduction to Business Analytics: Introduction to Business Analytics, need and science of data driven (DD) decision making, Descriptive, predictive, prescriptive analytics and techniques, big data analytics, Web and Social media analytics, Machine Learning algorithms, framework for decision making, challenges in DD decision making and future.
Recommender Systems, Automated Decision-Making, Reinforcement Learning for Decision-Making

Practical Components

1. Descriptive, Predictive analysis on numeric data
2. Implementation of central tendency models on an appropriate dataset
3. Sample case study on Types of Social Media Analytics Report
4. Sample case study on Decision making

UNIT II **12+ 9 Hours**

Data mining, Text mining, Web mining, Spatial mining, Process mining, Data warehouse and datamarts. Data mining process KDD, CRISP-DM, SEMMA and Domain-Specific, Classification and Prediction performance measures - RSME, MAD, MAP, MAPE, Confusion matrix, Receiver Operating Characteristic curve & AUC; Validation Techniques - hold-out, k-fold cross-validation, LOOCV, random subsampling, and bootstrapping.

Nested Cross-Validation, Stratified Cross-Validation, Time Series Cross-Validation

Practical Components

1. Measure Prediction performance with confusion matrix on sample dataset
2. Implement ROC curve and AUC
3. Implementation of K-fold cross validation for sample dataset
4. Implementation of bootstrapping with suitable datasets

UNIT III **11+ 6 Hours**

Forecasting Techniques: Introduction, time-series data and components, forecasting accuracy, moving average method, single exponential smoothing, Holt's method, Holt-Winter model, Croston's forecasting method, regression model for forecasting, Auto regression models, autoregressive moving process, ARIMA, Theil's coefficient.
Coefficient of Variation, Shannon Diversity Index, Simpson's Diversity Index, Gini Coefficient

Practical Components

1. Implementation of various smoothing techniques on sample datasets
2. Implement Moving Average forecasting technique
3. Implement Holt-Winter forecasting technique
4. Implementation of regression for forecasting
5. Measure the Theil's coefficient for suitable data

UNIT IV **11+ 6 Hours**

Six Sigma: Introduction, origin, 3-Sigma Vs Six-Sigma process, cost of poor quality, sigma score, industry applications, six sigma measures, DPMO, yield, sigma score, DMAIC methodology, Six Sigma toolbox
Statistical Process Control, Design of Experiments, Theory of Constraints (TOC

Practical Components

1. Illustrate difference between 3-sigma vs six-sigma process
2. Implementation of DMAIC method with suitable scenario
3. Case study on Enhancing Employee Management using Lean Six Sigma toolbox

48+32 Hours
Textbooks:

1. Jaiwei Ham and Micheline Kamber, Data Mining concepts and techniques, Kauffmann Publishers 2006
2. Efraim Turban, Ramesh Sharda, Jay E. Aronson and David King, Business Intelligence, Prentice Hall, 2008.
3. W.H.Inmon, Building the Data Warehouse, fourth edition Wiley India pvt. Ltd. 2005
4. Marc J. Schniederjans, Dara G. Schniederjans, Christopher M. Starkey, "Business Analytics Principles, Concepts, and Applications with SAS", Associate Publishers,2015.

References:

1. S. Christian Albright, Wayne L. Winston, "Business Analytics - Data Analysis andDecision Making", 5th Edition, Cengage, 2015.
2. U Dinesh Kumar, "Data Analytics", Wiley Publications, 1st Edition, 2017.
3. Ralph Kimball and Richard Merz, The data warehouse toolkit, John Wiley, 3rd edition,2013.
4. Michel Berry and Gordon Linoff, Mastering Data mining, John Wiley and Sons Inc,2nd Edition, 2011
5. https://machinelearningmastery.com/time-series-forecasting-methods-in-python-cheat-sheet/
6. https://leansixsigmatoolbox.com/2023/06/08/ongoing-education-lean-six-sigma-employee-engagement/

Internal Assessment Pattern

Cognitive Level	Int. Test 1 (%)	Int. Test 2 (%)	Open Book Exam
Remember	30	30	-
Understand	50	50	-
Apply	20	20	50
Analyze	--	--	50
Evaluate	--	--	-
Create	--	--	-
Total (%)	**100**	**100**	**100**

Remember

1. What is business intelligence?
2. Define predictive analysis

Understand

1. Explain the k-fold cross validation with a suitable machine learning algorithm
2. Explain forecasting techniques for time-series data

Apply

1. Illustrate difference between 3-sigma vs six-sigma process
2. Implementation of DMAIC method with suitable scenario

23CSC22 Web Application Frameworks

3 2 0 4

Course Outcomes:

1. Describe the fundamentals of web framework.
2. Design and implement Angular-based applications.
3. Design and implement React-based applications.
4. Build robust and secure web applications using Spring Boot.
5. Create efficient and dynamic REST APIs with CRUD functionality
6. Design and develop dynamic web applications using MVT architecture

CO-PO Mapping:

COs	PO1	PO2	PO3	PO5	PSO1	PSO2
1	3	3	1	1	2	1
2	3	3	3	3	2	3
3	3	3	3	3	2	3
4	3	3	3	3	3	2
5	2	3	2	3	2	3
6	3	3	3	2	2	3

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit-I **13+10 Hours**

Fundamentals of Web Framework

Introduction to frameworks-Framework applications, General-purpose website frameworks-Server-side, Client-side features, Types of framework architecture.

Angular.js

Introduction – Angular MVC, Ajax, Data binding, Angular concepts - Directives, Components & Modules, Templates & Data Binding, Routing & Navigation, Modules, Expressions, HTTP and API integration.

JavaScript ES6 concepts, MVC, Three-tier organisation, Ajax, Scopes, Controllers

Practical Components

1. Implementation of angular directives
2. Implementation of angular components
3. Implementation of custom/user-defined directive
4. Create a TODO Application using Angular JS (Single page application)
5. Implement routing and navigation in Angular JS

Unit-II **13+12 Hours**

React.js

Introduction to React: What is Full-Stack Web Development?, Introduction to React, React App Overview, Core Components: Introduction to JSX, React Components, State and Props, Lifecycle Methods: class and function components, use of fetch and axios, Hooks, React Router and Single Page Applications: Presentational and Container Components, React Virtual DOM, React Router: Parameters, React Forms, Flow Architecture and Introduction to Redux: Controlled Forms, Uncontrolled Components, The Flux Architecture, Introduction to Redux, React Redux Forms

JavaScript ES6 concepts, React Events, State Management, Introduction to React Native

Practical Components

1. React Components
2. React Router and Single page applications
3. Controlled Form Validation, Uncontrolled Forms
4. Create a TODO Application using React JS
5. Creating a movie list Application in React JS
6. Creating a Progressive web application with Reusable React Components

Unit-III **10+4 Hours**

Spring Boot

Introduction to Spring Boot, Application Structure, Framework Theory, Annotations, Bean Management, Dependency Injection, Spring Data: JDBC, ORM, JPA, Building Rest APIs, Starter Dependencies, database Integration, Creating Model Class.

Spring MVC, Error Handling, Spring Boot Security, Asynchronous Programming

Practical Components

1. Configuration in Spring Boot and Create a Simple Spring Boot Application

2. Apply Validation to given input data with REST endpoint and test with valid or invalid data using any API tester.

Unit-IV

Django **12+6 Hours**

Introduction, virtual environment, install Django, Create Django App, App Life Cycle, Project Structure, URL mapping, Django MVT - model and CRUD operations on model, Django view and Django template and configuration, Static File Handling, Django Admin- create user, Response and Redirect, Django Forms

Basic concepts of python, Python packages, Django Cookies and sessions handling

Practical Components

1. Creating models for database queries
2. Writing Django template for rendering data
3. Implementation of REST API using Django

Total: 48+32 Hours

Textbook (s)

1. Angular: Up and Running, ShyamSeshadri, 1st Edition, O'Reilly, 2018
2. Struts the Complete Reference, James Holmes, 2nd Edition, Mc. Graw Hill Professional, 2006.
3. Professional Java Development with the Spring Framework, Alef Arendsen, Thomas Risberg, Rod Johnson, Wiley publications, 2007.
4. Programming with Django, Wiley Publishing

Reference (s)

1. Angular 6 for Enterprise-Ready Web Applications, DoguhanUluca, 1st edition, 2018
2. The Definitive Guide to Django, Adrian Holovaty, Jacob Kaplan-Moss, Apress, 2009.
3. Struts 2 In Action, Donald Brown, Chad Michael Davis, Scott Stanlick, Dreamtech press, 2008.

Internal Assessment Pattern

Cognitive Level	Int. Test 1 (%)	Int. Test 2 (%)
Remember	50	40
Understand	30	40
Apply	20	20
Analyze	--	--
Evaluate	--	--
Create	--	--
Total (%)	100	100

SAMPLE QUESTION (S)

Remember:

1. What is a web framework?
2. Define Full-Stack Web Development.
3. What is Spring Boot?
4. What is Django?
5. What command is used to create a Django project?

Understand:

1. Explain the concept of Model-View-Controller (MVC) in a framework.
2. Explain the difference between state and props in React.
3. Describe the role of JSX in React.
4. Explain the purpose of Spring Boot Annotations with examples.
5. What is the difference between Django MVT and traditional MVC architecture?

Apply:

1. Create a simple HTTP request using a client-side framework.
2. Write a simple React component using JSX.
3. Write a simple Spring Boot application that prints "Hello, World!".
4. Implement a simple CRUD application using Spring Data JPA.
5. Develop a Django model for a product catalog and apply CRUD operations.

Analyze:

1. Compare and contrast server-side and client-side frameworks.
2. Compare useEffect and lifecycle methods in class components.
3. How does Spring Boot handle Bean Management and Dependency Injection?
4. How does Django's URL dispatcher work?
5. Compare Django ORM vs. raw SQL queries.

23MLC32 Cloud Services using AWS

3 0 2 4

Course Outcomes

1. Describe various services offered by Amazon Web Services (AWS) and their applications in cloud computing.
2. Understand the concepts and implementation of Identity and Access Management (IAM) for secure AWS resource access.
3. Demonstrate the usage of AWS Directory Services and AWS Artifact for identity integration and compliance.
4. Deploy and manage scalable computing resources using Amazon Elastic Compute Cloud (EC2).
5. Illustrate the features and functionalities of Amazon Elastic File System (EFS) for scalable storage.
6. Compare AWS storage solutions such as S3, EBS, and Glacier based on use cases and performance needs.

COs–POs Mapping

COs	PO_1	PO_2	PO_3	PO_4	PO_5	PSO_1
1	2	2	2	2	2	2
2	2	2	2	2	2	2
3	2	2	3	2	2	2
4	3	3	3	3	3	2
5	2	2	2	2	2	2
6	2	2	2	2	2	2

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit-I **12+8 Hours**

Introduction to AWS:

Classic Data Center , Virtualization, Service Comparison: AWS, Azure, and GCP, Amazon Web Services (AWS) and its Benefits, AWS Global Infrastructure, Signup an AWS Free Tier Account.

Different Amazon Web Services, Ways to access AWS: CLI, Console, and SDKs, Explore Management Console and Configure AWS CLI

AWS CloudShell

Practical Components

1. Signup an AWS Free Tier Account
2. Explore Management Console and configure AWS CLI
3. Access AWS: CLI, Console, and SDKs
4. Working with AWS services in AWS CloudShell

Unit-II **12+8 Hours**

Identity and Access Management in AWS:

Identity and Access Management (IAM), Managing Users with IAM, Managing Permissions with Groups, IAM Policy and its Elements, IAM Roles ,Password Policy, Best Practices for IAM

AWS Security Token Services, AWS Single Sign-on, AWS Resources Access Manager, Active directory, Microsoft Active directory, AWS Directory Services, AWS Artifact, AWS Audit Manager, Accessing Billing and Alerts.

AWS Certificate Manager

Practical Components

1. Creating New Users, Roles, and Policies on AWS Console
2. Creating an user group for a collection of IAM users
3. Login to AWS Console via MFA
4. AWS Artifact

Unit-III **12+8 Hours**

Amazon Elastic Compute Cloud (EC2):

Amazon Elastic Compute Cloud (EC2) and Its Benefits, Amazon Machine Image (AMI), Security Groups in AWS, Authentication through Key-pair, Hardware Tenancy – Shared vs. Dedicated, Networking Layer in EC2: VPC, Elastic Network Interface (ENI) and Its Attributes

Public IP vs. Elastic IP, Instance Store

Elastic Block Store (EBS), Its Features and Volume Types

Solid State Drive: General Purpose SSD and Provisioned IOPS

Hard Disk Drive: Throughput Optimized HDD and Cold HDD

Snapshots, Elastic File System (EFS) and Its Features, Amazon Batch, Windows File Server

Amazon FSx for Windows File Server

Practical Components

1. Creating an EC2 instance and Custom AMI
2. Host the Website Inside EC2 instance
3. Create an Elastic IP and attaching an EBS Volume Externally
4. Creating a Snapshot

Unit-IV **12+8 Hours**

AWS Storage Options:

Traditional Storage, Need to Move to Cloud Storage, Traditional vs. Cloud Storage Cost, Cloud Storage, Different Storage Options Available on AWS, Simple Storage Service (S3) and Its Components, Working of S3, Bucket Policy, Access Control List (ACL)

Cross-Region Replication (CRR) and Its Use Case, Amazon S3 Transfer Acceleration, Choice of Storage Classes on S3, Lifecycle Policy of S3 Bucket, AWS Backup.

CDN: CloudFront, Storage Gateway

Practical Components

1. Hosting a Static Website on Amazon S3
2. Replicating Data Across Regions
3. Transfer and Retrieve Data from Glacier
4. Accessing Website Using CloudFront

Total:48+32 Hours

Textbook(s):

1. Amazon Web Services (AWS) 2022 Everything You Need To Know About The Amazon Web Service From Start To End, 2022, Maxwell Harris, Maxwell Harris.
2. Implementing AWS: Design, Build, and Manage your Infrastructure, Yohan Wadia, Rowan Udell, Lucas Chan, Udita Gupta, packt publishers.

Internal Assessment Pattern

Cognitive Level	Internal Test 1 %	Internal Test 2 %	Lab Examination %
Remember	25	25	---
Understand	50	25	---
Apply	25	50	40
Analyze	---	---	20
Evaluate	---	---	40
Create			
Total (%)	**100**	**100**	**100**

Sample Question(s)

Remember

1. List any two cloud services provided by Amazon AWS.
2. What is AWS Artifact?
3. Define EC2.
4. What is Access Control List?

Understand

1. What is Virtualization and explain about that.
2. Explain about Identity and Access Management (IAM).
3. Write and explain about AWS Directory Services.
4. Explain about Access Control List (ACL).

Apply

1. How do we sign up for an AWS Free Tier Account?
2. Illustrate the process of Move to Cloud Storage from normal storage.
3. How Cross-Region Replication (CRR) works?

23CS007 Cloud Computing Essentials

3 0 2 4

Course Outcomes

1. Interpret the architecture and infrastructure models of cloud computing, strengths, and limitations of cloud computing.
2. Explain the virtualization concepts of machines and data centers.
3. Infer the design concepts of cloud ready applications
4. Compare different cloud centre's implementation
5. Discuss the concepts of cloud scaling and disaster recovery
6. Analyze the performance, scalability, and availability of the underlying cloud technologies and software

CO–PO Mapping

CO's	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	1	-	3	-	-	3	2	3	1	-	-	-	-	-
CO2	2	-	3	-	-	1	1	2	3	-	-	-	-	-
CO3	3	-	2	-	-	2	3	1	2	-	-	-	-	-
CO4	4	-	2	-	-	1	3	3	1	-	-	-	-	-
CO5	5	-	2	-	-	1	3	1	2	-	-	-	-	-
CO6	6	-	2	-	-	3	3	3	3	-	-	-	-	-

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I **12+6 Hours**

Characterization of Distributed Systems: Introduction, Examples of Distributed Systems, Resource Sharing and the Web, Challenges. System Models: Introduction, Architectural Models- Software Layers, System Architecture, Variations, Interface and Objects, Design Requirements for Distributed Architectures, Fundamental Models- Interaction Model, Failure Model, Security Model.

*Communication between Distributed Objects- Object Model, Distributed Object Modal.***Practical Components:**

1. Study of Cloud Computing & Architecture.
2. Install KVM/VMware Workstation for creating a virtual environment on windows platform. Concept: Virtualization
3. Install a C compiler in the virtual machine created using virtual box and execute Simple Programs.

Unit II **12+8 Hours**

Introduction to Cloud Computing

Overview of Computing Paradigm: Recent Trends in Computing, Evolution of Cloud Computing. Introduction to Cloud Computing: Cloud Computing (NIST Model), Properties, Characteristics & Disadvantages, Role of Open Standards. Cloud Computing Architecture: Cloud Computing Stack, Service Models (XaaS), Deployment Models. Infrastructure as a Service (IaaS): Introduction to IaaS, Resource Virtualization. Platform as a Service (PaaS): Introduction to PaaS, Cloud Platform and Management. Software as a Service (SaaS): Introduction to SaaS, Web services, Web 2.0, Web OS.

Companies in the Cloud Today, Amazon Web Servies, Google services, IBM Cloud, Windows Azure, Tata Cloud, Salesforce.com

Practical Components:

1. Install Google App Engine. Create hello world app and other simple web applications using python/java.
2. Use GAE launcher to launch the web applications.
3. Simulate a cloud scenario using CloudSim and run a scheduling algorithm that is not present in CloudSim.
4. Signup an AWS Free Tier Account and Explore Management Console and configure AWS CLI

Unit III **12+10 Hours**

Virtualization & Design

Virtualization, Virtual machine, Implementation Levels of Virtualization, Virtualization Structures/Tools and Mechanisms, Virtualization of CPU, Memory, and I/O Devices, Virtual Clusters and Resource Management, Data centre, Virtualization for Data-Centre Automation. Service Levels for Cloud Applications Ready for the cloud: Web Application Design, Machine Image Design, Privacy Design, Database Management.

various hypervisors like VMware, KVM, oracle VM, HTTPS,Electronic Payment

Practical Components:

1. Working on to transfer the files from one virtual machine to another virtual machine.
2. Working on to launch virtual machine using trystack (Online Openstack Demo Version)
3. Set up a pseudo-distributed, single-node Hadoop cluster backed by the Hadoop Distributed File System, running on Ubuntu Linux. After successful installation on one node, configuration of a multi-node Hadoop cluster(one master and multiple slaves).Hadoop file management: Adding files and directories, Retrieving files, Deleting files
4. Run Wordcount application using Hadoop single node cluster.
5. Move files between regular Linux file-system and HDFS using put and get commands

Unit IV

Cloud Service Providers **12 + 8 Hours**

EMC, EMC IT, Captiva Cloud Toolkit, Google, Cloud Platform, Cloud Storage, Google Cloud Connect, Google Cloud Print, Google App Engine, Amazon Web Services, Amazon Elastic Compute Cloud, Amazon Simple Storage Service, Amazon Simple Queue, service, Microsoft, Windows Azure, Microsoft Assessment and Planning Toolkit, SharePoint, IBM, Cloud Models, IBM Smart Cloud, SAP Labs, SAP HANA Cloud Platform, Virtualization Services Provided by SAP, Sales force, Sales Cloud.

Service Cloud: Knowledge as a Service, Rack space, VMware, Manjra soft, Aneka Platform

Practical Components:

1. Working and installation of Microsoft Azure
2. Working with Mangrasoft Aneka Software
3. Installation and Configuration of Justcloud
4. Working with AWS services in AWS CloudShell

Total: 48+32 Hours

Textbook (s)

1. George Coulouris, Jean Dollimore, Tim Kindberg, "Distributed Systems- Concepts and Design", Fourth Edition, Pearson Publication
2. Cloud Computing Bible, Barrie Sosinsky, Wiley-India, 2010
3. Cloud Computing: Principles and Paradigms, Editors: RajkumarBuyya, James Broberg, Andrzej M. Goscinski, Wile, 2011
4. Cloud Computing: Principles, Systems and Applications, Editors: Nikos Antonopoulos, Lee Gillam, Springer, 2012
5. Cloud Security: A Comprehensive Guide to Secure Cloud Computing, Ronald L. Krutz, Russell Dean Vines, Wiley-India, 2010

Reference (s)

1. Michael Miller, Cloud Computing-Web Based Applications that change the way you work and collaborate online, 1st Edition, Pearson Education, Publishing, 2011
2. Kai Hwang, Geoffrey C Fox and Jack J.Dongarra, Distributed & Cloud Computing from Parallel Processing to the Internet of Things , 1st Edition, MK Publishing, 2010
3. David S Linthicum, Cloud Computing and SOA Convergence in Your Enterprise: A Step-by-Step Guide, 1st Edition, Addison-Wesley, 2009
4. Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance, Tim Mather, SubraKumaraswamy, ShahedLatif, O'Reilly, SPD, rp2011.

Internal Assessment Pattern

Cognitive Level	Int. Test 1 (%)	Int. Test 2 (%)
Remember	30	30
Understand	40	40
Apply	30	30
Analyze	--	--
Evaluate	--	--
Create	--	--
Total (%)	100	100

SAMPLE QUESTION (S)

Remember

1. Define Cloud Computing.
2. List types of virtualizations.
3. Define proactivescaling.
4. What is CIATriad?

Understand

1. Explain different cloud Infrastructure models.
2. Explain different levels of virtualization.
3. Explain about cloud Network security in detail.
4. Explain about Recovery Point Objective.
5. With neat diagram explain the functioning of Xen Architecture.
6. List and explain various cloud service providers risks.

Analyze

1. Compare cloud center and service infrastructure.
2. Analyze different cloud services provided by Amazon

23CS008 Cryptography and Network Security (Elective III)

3 0 2 4

Course Outcomes

1. Explain the fundamentals of Cryptography, encryption and decryption algorithms
2. Make use of the symmetric and public key cryptographic algorithms
3. Choose the various authentication applications for security
4. Interpret the functionalities of IP and web security.
5. Demonstrates the functionalities of firewalls
6. Explain various non-cryptographic protocol vulnerabilities

CO–PO Mapping

CO's	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	1	2	-	-	2	-	3	-	-	-	2	-	-
CO2	3	2	2	-	-	2	-	3	-	-	-	2	-	-
CO3	3	2	2	-	-	2	-	3	-	-	-	2	-	-
CO4	3	1	2	-	-	2	-	2	-	-	-	2	-	-
CO5	3	1	2	-	-	2	-	3	-	-	-	2	-	-
CO6	3	1	2	-	-	2	-	2	-	-	-	2	-	-

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I **12+10 Hours**

Introduction: Security Attacks, Security Goals, Computer criminals, Methods of Defence, Security Services, Security Mechanisms. **Basics of Cryptography:** Symmetric Cipher Model, Substitution Techniques, Transportation, Techniques, Other Cipher Properties- Confusion, Diffusion, Block and Stream Ciphers. Block Cipher Design Principles and Modes of Operations, **Symmetric Key Cryptosystems:** Principles of Private Key System, Data Encryption Standard (DES), Strength of DES, Triple DES, International Data Encryption Algorithm(IDEA), Advanced Encryption Standard (AES) *Blowfish, CAST-128.*

Practical Components:

1. Write a program that contains a string (char pointer) with a value "Hello world". The program should XOR each character in this string with 0 and displays the result.
2. Write a program that contains a string (char pointer) with a value "Hello world". The program should AND or and XOR each character in this string with 127 and display the result.
3. Write a program to perform encryption and decryption using the following algorithms
    a. Ceaser cipher
    b. Substitution cipher
4. Write a program to perform encryption and decryption using the following algorithms
    a. Mono-alphabetic cipher
    b. Hill Cipher
5. Write a program to perform encryption using the following algorithms
    a. Railfence Technique
    b. Transposition Technique

Unit II **12+8 Hours**

Public Key Cryptography: Principles of Public Key Cryptosystems, RSA Algorithm, Diffie-Hellman Key Exchange. Cryptographic Hash Functions: Principles of Cryptographic Hash functions, Applications of Cryptographic Hash Functions, Secure Hash Algorithm (SHA), Message Authentication Codes – Message Authentication Requirements and Functions, HMAC, Digital Signatures, Elgamal Digital Signature Schemes

Digital Signature Standards

Practical Components:

1. Write a program to implement the Play-fair Cipher algorithm.
2. Write a program to implement the DES algorithm logic.
3. Write a program to implement the Blowfish algorithm logic.

Unit III **12+8 Hours**

Authentication Applications: Kerberos, Key Management and Distribution, X.509 Directory Authentication service, Public Key Infrastructure, Electronic Mail Security: Pretty Good Privacy, S/MIME. **IP Security:** Overview, Architecture, Authentication Header, Encapsulating Security Payload, Combining security Associations, Internet Key Exchange,

Web Security: Web Security Considerations, Secure Sockets Layer and Transport Layer Security,.

HTTPS, Electronic Payment

Practical Components:

1. Write a Java program to implement RSA algorithm.
2. Calculate the message digest of a text using the SHA-1 algorithm.
3. Write a program to implement the Triple-DES algorithm logic.
4. Write a program to implement the AES algorithm logic.

Unit IV

Locking, Recovery Systems, Indexing, Different Types of Data **12 + 6 Hours**

IDS and Firewalls: Intruders, Intrusion Detection, Password Management, Firewalls-Characteristics, Types of Firewalls, Placement of Firewalls, Firewall Configuration, Trusted Systems. Non-cryptographic protocol Vulnerabilities: DoS, DDoS, Session Hijacking and Spoofing, Software Vulnerabilities- Phishing, Buffer Overflow.
Format String Attacks, SQL InjectionCybercrime and Computer Crime, Intellectual Property.

Practical Components:

1. Implement the Diffie-Hellman Key Exchange mechanism.
2. Write a program that can encrypt and decrypt using a general substitution block cipher.
3. Write a program that can encrypt and decrypt in 4-bit in cipher clock chaining mode using DES.

Total: 48+32Hours

Textbook (s)

1. William Stallings, "Cryptography And Network Security – Principles and Practices", 7th edition, Pearson Education Limited 2017.
2. AtulKahate, "Cryptography and Network Security", 2nd edition Tata McGraw-Hill, 2003.
3. Behourz A Forouzan, Cryptography and Network Security, 2nd edition, Tata McGraw-Hill 2011

Reference (s)

1. Matt Bishop ,"Computer Security art and science ", Second Edition, Pearson Education, 2002
2. Wade Trappe and Lawrence C. Washington, "Introduction to Cryptography with Coding Theory" Second Edition, Pearson Education, 2007
3. Jonathan Katz, and Yehuda Lindell, Introduction to Modern Cryptography, CRC Press, 2007
4. Douglas R. Stinson, "Cryptography Theory and Practice", Third Edition, Chapman & Hall/CRC, 2006
5. Wenbo Mao, "Modern Cryptography – Theory and Practice", Pearson Education, First Edition, 2006.
6. OWASP top ten security vulnerabilities: http://xml.coverpages.org/OWASPTopTen.pdf

Internal Assessment Pattern

Cognitive Level	Int. Test 1 (%)	Int. Test 2 (%)
Remember	30	30
Understand	40	40
Apply	30	30
Analyze	--	--
Evaluate	--	--
Create	--	--
Total (%)	100	100

SAMPLE QUESTION (S)

Remember

1. Mention any two security attacks
2. List any two goals of security
3. Define Hash function

Understand

1. Differentiate between asymmetric and symmetric key cryptography
2. How do we achieve authentication?
3. Differentiate between the two applications of hash function.

Apply

1. How do we apply PGP to the Email Security?
2. Implement firewall using iptables command.
3. Can message encryption itself provide measure of authentication?

Open Book Exam Questions

Question 1:

A man named Jones wanted to chat with his girlfriend Goldie. But he can see that all his family is around him and even his girlfriend is also with her parents. So, he thought to send a secret message to his girlfriend. They usually love to meet in the "Central Park". Now Jones wants to send a message to Goldie as "Hi Goldie. How are you. Because you are with your parents, and I am with my parents we cannot speak with each other. But I want to meet you at our favourite place central park tomorrow after your class". Help jones to convert the message to unreadable format using the key of their favorite place. Also suggest him how would jones tell Goldie that the letter is originated from Jones only.

Question 2:

You have decided to start a startup after graduation. But you alone cannot be doing this, so you have to ask help from your friends. Suggest what techniques you require and which sort of people you would select so that your company will be a huge success. Explain the techniques clearly to your friends and tell them what they have to do in detail.

23ML606 Deep Learning Techniques Lab

0 0 3 1.5

Course Outcomes

1. Design the basic Neural Network and Multilayer Perceptron model.
2. Compare the effects of different optimizers and regularization methods.
3. Develop different CNN and transfer learning models.
4. Implement different Character Encoding techniques.
5. Apply different Deep Unsupervised models.
6. Design Auto Encoders, Generative Adversarial Network and Deep Boltzmann Machine.

CO–PO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO8	PSO1	PSO2
1	2	2	3	3	3	2	3	3
2	2	2	3	3	3	2	3	3
3	3	2	3	3	3	2	3	3
4	3	2	3	2	3	3	3	3
5	3	2	3	2	3	2	3	3
6	3	2	3	1	3	3	3	3

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

List of Experiments

1. Design the structure and initialize the parameters of a neural network model.
2. Perform forward computation and compute the cost function in a multilayer perceptron (MLP).
3. Compare training performance using SGD, Adam, and RMSprop optimizers on the same dataset.
4. Evaluate the impact of Dropout, DropConnect, and Batch Normalization on the performance of a deep neural network.
5. Develop a digit recognition system using a Convolutional Neural Network (CNN) on the MNIST dataset.
6. Implement a character recognition system using CNN on a suitable dataset.
7. Apply transfer learning using the VGG-16 model for an image classification task.
8. Use transfer learning with the YOLO model for real-time object detection.
9. Apply one-hot encoding and word embedding techniques for text representation.
10. Use pretrained word embeddings to represent and analyze textual data.
11. Implement a Recurrent Neural Network (RNN) model for text classification tasks.
12. Understand and implement Long Short-Term Memory (LSTM) networks for sequence prediction.
13. Build and train an autoencoder model for data compression and reconstruction.
14. Implement a variational autoencoder to learn latent representations of data.
15. Understand and implement a Generative Adversarial Network (GAN) for image generation.
16. Explore and implement Deep Boltzmann Machines for unsupervised learning tasks.

Lab Manual

1. Deep Learning Laboratory– AIML Department, GMRIT.

Textbook (s)

1. Bengio, Yoshua, Ian J. Goodfellow, and Aaron Courville. "Deep Learning." An MIT Press book in preparation. (2015).
2. Fundamentals of Deep Learning: Designing Next-generation Machine Intelligence Algorithms by Nicholas Locascio and Nikhil Buduma O'Reilly Media; 1 edition (June 29, 2017)
3. Simon S. Haykin, Neural Networks, Prentice Hall, 2nd edition
4. B. Yegnanrayana , “Artificial Neural Networks” , PHI.

Reference (s)

1. Franchois Chollet, Deep Learning with Python
2. Deep Learning: A Practitioner's Approach by Adam Gibson and Josh Patterson Shroff/O'Reilly; First edition (2017)
3. Python Deep Learning by Daniel Slater and Gianmario Spacagna, Packt Publishing; 2/e (January 16, 2019)
4. Bishop, C., M., Pattern Recognition and Machine Learning, Springer, 2006
5. Kevin P. Murphy.,MachineLearning:A Probabilistic Perspective

23MPX01 Mini Project

0 0 3 1.5

Course Outcomes

1. Identify a contemporary engineering application to serve the society at large
2. Use engineering concepts and computational tools to get the desired solution
3. Justify the assembled/fabricated/developed products intended
4. Organize documents and present the project report articulating the applications of the concepts and ideas coherently
5. Demonstrate ethical and professional attributes during the project implementation
6. Execute the project in a collaborative environment

COs – POs Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	2	3	2	3	2	2	2	2	3	3	3	3
CO2	3	3	3	2	3	3	3	3	2	3	3	3	3	3
CO3	2	3	3	3	3	3	3	2	2	2	3	3	3	3
CO4	2	2	3	3	3	3	3	3	3	3	3	3	3	3
CO5	3	3	2	2	3	3	3	3	3	2	3	3	3	3
CO6	3	3	2	2	3	3	3	3	3	3	3	3	3	3

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Mini Project: The curriculum offers Mini Projects in two different forms viz: (i) Mini Project as a mandatory component in all lab courses (ii) 2 credit Mini Project during 5th or 6th semester. With respect to second one (ii) student will take mini project batch wise and the batches will be divided as similar to lab courses. The report will be evaluated by a committee as nominated by CoE constituted with internal & external panels

23ESX02 Employability Skills II

0 0 2 2

Course Outcomes

1. Demonstrate oral communication and writing skills as an individual to present ideas coherently
2. Develop life skills with behavioral etiquettes and personal grooming.
3. Assess analytical and aptitude skills.
4. Develop algorithms for engineering applications.
5. Solve engineering problems using software.
6. Utilize simulation tools for testing

COs –POs Mapping

CO's	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	1	1	1	1	1	-	-	-	2	3	3	3	-	-
CO2	1	1	3	1	1	-	-	-	2	3	3	3	-	-
CO3	1	1	3	1	1	-	-	-	2	2	3	3	-	-
CO4	2	2	3	2	2	-	-	-	3	3	2	3	-	-
CO5	3	2	3	2	2	-	-	-	3	2	2	3	-	-
CO6	2	2	2	2	3	-	-	-	3	2	3	3	-	-

3–Strongly linked | 2–Moderately linked| 1–Weakly linked

Soft Skills:

Sl No.	Topic & Content	No. of Periods
1.	**Resume (Recap):** Resume? Templates? Mistakes to be avoided in a Resume and Steps to be followed in preparing it.	**01**
2.	**Group Discussions (Recap) & Practice:** GD? Stages of a GD, Skills assessed in a GD, Blunders to be avoided, How to excel in a GD? Practice sessions and sharing Feedback. (Screening sample Videos)	**01**
3.	**Interview Skills:** Interview? Types of Interviews, Dos & Don'ts, Skills assessed in an Interview, Mistakes to be avoided, How to equip oneself to excel? How to handle the Typical Interview Questions? (with Examples)	**03**
4.	**Mock Interviews**: Practice sessions with Feedback.	**02**
5.	**Exercises related to Communication:** Email Writing, Voice Versant., etc.	**01**
	Total Periods	**08**

Quantitative Aptitude:

Sl. No.	Topic	No. of Periods
1.	Time and Distance	01
2.	Time and Distance	01
3.	Problems on Trains	01
4.	Problems on Trains	01
5.	Blood relations	01
6.	Ratio and Proportions	01
7.	Calendars	01
8.	Clocks	01
	Total Periods	**08**

Domain Specific: (Comprehensive Coding Skills- Phase 2) **16 Hours**

Module-1: Advanced Algorithms (Weeks 1–8)

1. Graph Representation and Traversals
   Problems: Flood Fill, Number of Provinces, Detect Cycle in Directed Graph
2. Shortest Path and Connectivity Planning
   Problems: Dijkstra's Algorithm for Road Navigation, Prim's Algorithm for Cable Laying, Optimize Delay in Networks
3. Bellman-Ford and Negative Cycle Detection
   Problems: Bellman-Ford Shortest Path, Detect Negative Cycle
4. Bridge Finding and Strong Connectivity
   Problems: Bridges in Graph (Tarjan's Algorithm), Strongly Connected Components (Kosaraju's Algorithm)
5. Greedy Algorithms
   Problems: Activity Selection, Job Sequencing Problem, Sort K-Sorted Array, Minimum Platforms
6. Dynamic Programming (DP) – Bottom-Up vs Top-Down
   Problems: Fibonacci (DP), Climbing Stairs (1D DP)
7. 1D DP Patterns – Subset-based Optimization
   Problems: Coin Change, Maximum Product Subarray
8. 2D DP and Knapsack Problems
   Problems: Longest Common Subsequence (LCS), Unique Paths, Subset Sum, 0/1 Knapsack, Target Sum

Module 2 - Database Management Systems (DBMS) (Weeks 9–16)

1. SQL Basics and Filtering
   Problems: Top N Salaries, Group-wise Maximums, Students above Average
2. SQL Joins
   Problems: Employee-Department Join, Customers without Orders
3. Aggregate Functions with GROUP BY and Conditions
   Problems: Department with Highest Salary, Average Salary Excluding Extremes
4. Subqueries and Operators
   Problems: Second Highest Salary, Employees Earning More than Manager, Duplicate Emails
5. Correlated Subqueries and Set Operations
   Problems: Compare Average Salaries, Conditional Grouping, Non-matching Records
6. Views and Indexes
   Problems: Create and Query Simple Views, Indexed Lookups on Large Tables.
7. Functional Dependencies and Normalization
   Problems: Identify Normal Forms, Remove Redundancy, Find Candidate Keys
8. SQL Interview Simulation
   Problems: Join Multiple Tables, Identify Inconsistencies, Rank Without Window Functions.

References:

1. https://leetcode.com
2. https://www.geeksforgeeks.org/
3. https://www.codechef.com
4. https://www.hackerearth.com
5. https://www.interviewbit.com
6. https://www.hackerrank.com

23ATX-- AUDIT COURSE

0 0 0 0

Course Outcomes

At the end of the Audit Course the students will be able to

1. Interpret the meaning of values and select their goals by self- Investigation based on personal values.
2. Interpret the major events and issues related to a period in Indian history.
3. Assess the benefits and limitations of science and its application in technological developments towards human welfare
4. Check the awareness regarding basic human rights and to uphold the dignity of every individual.
5. Assess the individual and group behaviour and understand the implications of organizational behaviour on the process of management.
6. Determine the appropriateness of various leadership styles and conflict management strategies used in organizations.

COs –POs Mapping

CO's	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	-	-	-	-	-	2	2	3	1	2	-	2	-	-
CO2	-	-	-	-	-	3	2	1	1	1	-	1	-	-
CO3	-	-	-	-	-	2	3	1	1	1	-	2	-	-
CO4	-	-	-	-	-	3	2	3	1	2	-	2	-	-
CO5	-	-	-	-	-	2	2	1	3	2	-	1	-	-
CO6	-	-	-	-	-	2	2	3	1	2	-	2	-	-

3–Strongly linked | 2–Moderately linked| 1–Weakly linked

Audit Courses: Audit courses are among the compulsory courses and do not carry any credits. All the students shall register for one Audit courses in the beginning of 3rdsemester. List of the courses will be notified at the beginning of the third semester for all students and the student has to choose one audit course under self-study mode at the beginning of third semester. All the students (regular and lateral entry students) shall complete the audit course similar to other regular courses and the results will be indicated with "Satisfactory" or "Not Satisfactory" performance.

7th Semester
Career Path III (ML Ops)
23MLC13 Conversational AI (Elective V)

3 0 0 3

Course Outcomes

1. Understand the fundamentals of Natural Language Processing and basic vectorization approaches.
2. Understand the types of AI assistants, platforms and able to build their own Conversational AI.
3. Understand Dialog systems and Speech Recognition.
4. Understand the Dialog Management and Modeling.
5. Apply different rule based dialog management.
6. Make use of End-to-End Neural Dialogue Systems.

CO–PO Mapping

COs	PO1	PO2	PO4	PO12	PSO1
1	3	2	1	1	1
2	2	3	1	3	1
3	3	2	2	3	2
4	2	3	1	3	3
5	1	3	2	3	3
6	1	2	3	1	1

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I **12 Hours**

Introduction to NLP:

Definition, History, Building blocks of language, Approaches to NLP, NLP Pipeline, Deep learning in NLP and Transformers in NLP.

Basic Vectorization approaches- One-Hot Encoding, Bag of Words, Bag of N-Gram, TF-IDF; Neural language models, N-gram language model Sequence labeling for POS and Named Entities: POS tagging, Named Entities tagging.

Evaluation of Named Entity Recognition, Markov chains.

Unit II **12 Hours**

Introduction to Conversational AI: Introduction to AI assistants and their platforms: Types of AI assistants and their platforms, Primary use cases for AI assistant technology: self-service assistant, agent assist, and classification and routing.

Building your first conversational AI: Building a conversational AI for Fictitious Inc, What's the user's intent and Responding to the user.

The applications of Conversational AI and Objective of Chatbots.

Unit-III **12 Hours**

Dialog Management and Modelling

Introducing Dialogue Systems: Dialogue System, History, Present-day Dialogue System, Modelling conversation in dialogue systems.

Rule-Based Dialogue Systems: A typical dialogue systems architecture and Tools for developing dialogue systems and Evaluating Dialogue Systems

Challenges of the dialogue systems, Conversational Agents and Multimodal dialogue systems.

Unit IV **12 Hours**

End-to-End Neural Dialogue Systems

Neural Network Approaches to Dialogue Modelling, A Neural Conversational Model, Introduction to the Technology of Neural Dialogue and Open-Domain Neural Dialogue Systems.

Case Study: Chatbots in healthcare and mental health support, Voice-enabled devices and smart home applications.

Objective of Conversational AI with Rasa and Design Principles of the Chatbot.

Total: 48 Hours

Textbook (s)

1. Andrew R. Freed. Conversational AI: Chatbots that work, Manning; 1st edition (12 October 2021)
2. Yoav Goldberg. Neural Network Methods for Natural Language Processing, Morgan and Claypool Life Sciences, 30 April 2017

3. Michael McTear . Conversational AI: Dialogue Systems, Conversational Agents, and Chatbots (Synthesis Lectures on Human Language Technologies), Morgan & Claypool Publishers, 30 October 2020
4. Dan Jurafsky and James H. Martin. Speech and Language Processing (3rd ed. draft), Pearson, 22 Nov 2023.

Reference (s)

1. Xiaoquan Kong , Guan Wang . Conversational AI with Rasa by Packt., 1st edition, Oct 2021.
2. Stephan Bisser . Microsoft Conversational AI Platform for Developers End-to-End Chatbot Development from Planning to Deployment.1st edition, 17 Feb 2021.
3. Lee Boonstra . The Definitive Guide to Conversational AI with Dialog flow and Google Cloud build advanced enterprise chatbots, voice., 1st edition, 24 June 2021.
4. https://hao-cheng.github.io/ee596_spr2019/

Internal Assessment Pattern

Cognitive Level	Int. Test 1(%)	Int. Test 2 (%)	Open Book Exam (%)
Remember	20	20	--
Understand	50	40	--
Apply	30	40	50
Analyze	--	--	50
Evaluate	--	--	--
Create	--	--	--
Total (%)	**100**	**100**	**100**

SAMPLE QUESTION (S)

Remember

1. What is Natural Language Processing.
2. Define purpose of different building blocks of NLP
3. List any two challenges in NLP.

Understand

1. Explain the Spoken Language Understanding (SLU)
2. Explain the role of various Intent Classification.

Apply

1. Demonstrate the Frame-based SLU.
2. Make Use of Speech Technology in Dialogue System.

Analyze

1. Distinguish of modeling conversation in Dialogue systems.
2. Compare the Rule-Based Dialogue Systems and Evaluating Dialogue Systems.

Evaluate

1. Evaluate the End-to-End Neural Dialogue Systems.
2. Design an automated speech recognition system

Open book Question:

1. Design a model to analyse a given sentence is negative, positive or neutral for a given dataset and build end to end system.
2. Design and develop a model to Chatbots in healthcare and mental health support.

Career Path II (Full Stack Developer)
23CSC23 Web Application Databases

3 0 0 3

Course Outcomes:

1. Describe the architecture and programming concepts of web databases.
2. Implement state management and database connectivity mechanisms
3. Apply structured database management techniques.
4. Apply the principles of unstructured databases and NoSQL technologies.
5. Develop proficiency in database connections and communication protocols.
6. Analyze the security mechanisms for web services

CO-PO Mapping:

COs	PO1	PO3	PO5	PSO1
1	3	1	2	3
2	2	3	3	3
3	3	3	1	3
4	3	2	2	2
5	2	3	3	3
6	2	2	1	1

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit-I **11 Hours**

Overview of Web databases

Web application Overview: Basic concepts of web, Web database architectures, Web database gateways, Web-database programming: Client-side and server-side web database programming, State management: Session management, cookies, request and response header.

Database gateways, CGI, Browser extensions

Unit-II **13 Hours**

Structured Database – PostgreSQL

Structured Databases: Structured data, Data models, Relational database management systems, Overview of PostgreSQL – Introduction, Environment Setup, PostgreSQL Database: Create, Select and Drop, CRUD operations, Advanced PostgreSQL: Constraints, Views, Transactions, Auto Increment, Privileges, Functions, Python interface, Concept of migrate database, Challenges of structured data- MySQL, SQLite

PostgreSQL Interfaces, Triggers, MySQL Connection, SQLite Connection

Unit-III **13 Hours**

Unstructured Database – MongoDB

Unstructured Databases: Unstructured data and its usages, NoSQL databases, Overview of MongoDB – Introduction, No SQL Database, Advantage over RDBMS, MongoDB Data Types, Install MongoDB, Data Modeling, MongoDB Operators, Database Commands, MongoDB Architecture - Create Database, Drop Database, Collection - Create Collection, Drop Collection, CRUD: Documents, MongoDB Shell, MongoDB Cloud, Challenges of unstructured data- Cassandra, GraphQL

Architecture of Cassandra, Cassandra Data model, GraphQL Architecture and Application Components

Unit-IV

Database connections, Security & Communication in web services **11 Hours**

Connecting to database: Native database APIs, Database-independent APIs, Template-driven database access packages, Third-party class libraries.

Web requests: Structure of web requests, JavaScript Object Notation (JSON), Request methods – GET, POST, PUT, DELETE, response status codes

Security: User authentication, User authorisation, Proxy servers, Digital signatures, Digital certificates, SSL and HTTPS.

JDBC, ODBC, template parsing, JSON, User authentication and authorisation

Total: 48 Hours

Textbook (s)

1. Web Database Applications with PHP and MySQL, Hugh E. Williams, David Lane, 2nd Edition, O'Reilly, 2004.
2. Mastering PostgreSQL 13, Hans-Jürgen Schönig, 4th Edition, Kindle Edition.
3. NoSQL Distilled, PramodSadalage, Martin Fowler, 1st Edition, 2012

Reference (s)

1. Getting Started with NoSQL, GauravVaish, Paperback, Packt, 2013.
2. Fundamentals of Database Systems, ElmasriNavathe Pearson Education.
3. An Introduction to Database systems, C.J. Date, A.Kannan, S.SwamiNadhan, Pearson, 8th Edition.
4. Web Application Security, Andrew Hoffman, O'Reilly, 2020.

Internal Assessment Pattern

Cognitive Level	Int. Test 1 (%)	Int. Test 2 (%)	Open Book Exam (%)
Remember	20	20	--
Understand	40	40	--
Apply	20	20	40
Analyze	20	20	40
Evaluate	--	--	20
Create	--	--	--
Total (%)	100	100	100

Sample Questions

Remember:

1. What are the basic components of a web application?
2. What is a web database architecture?
3. What is a relational database management system (RDBMS)?
4. List different types of NoSQL databases.
5. Define database-independent APIs.

Understand:

1. Explain the process of session management in a web application.
2. Explain how PostgreSQL handles transactions.
3. Explain the architecture of MongoDB and how it differs from RDBMS.
4. How does JSON differ from XML in web communication?
5. Explain the difference between DELETE and TRUNCATE in PostgreSQL.

Apply:

1. Create a client-side script that validates form inputs.
2. Perform a database migration using PostgreSQL.
3. Implement CRUD operations in MongoDB using Python or Node.js.
4. Implement JWT-based authentication in a web application.
5. Write a query to filter data using comparison operators.

Analyze:

1. Compare two-tier and three-tier web database architectures.
2. Compare PostgreSQL, MySQL, and SQLite in terms of performance and scalability.
3. How does MongoDB handle ACID properties compared to traditional databases?
4. Compare session-based authentication vs. token-based authentication.
5. Analyze the impact of response status codes on API performance.

Career Path III (Cloud Computing)

23MLC33 Cloud Security Essentials (Elective V)

3 0 0 3

Course Outcomes

1. Understand the fundamental concepts and architecture of cloud computing from a security perspective.
2. Describe the characteristics, benefits, and challenges of various cloud service and deployment models in relation to security.
3. Identify and evaluate common threats, vulnerabilities, risks, and privacy concerns in cloud-based environments.
4. Understand security challenges, threats and risks involved in the cloud application.
5. Analyze the Security Requirements for the Architecture in cloud services.
6. Understand Data Security and best practices involved in securing the cloud.

CO–PO Mapping

COs	PO_1	PO_2	PO_3	PO_4	PO_5	PSO_1
1	3	2	2	2	2	2
2	3	2	2	2	2	2
3	2	2	2	2	2	2
4	2	2	2	2	2	2
5	3	2	3	2	3	2
6	3	2	2	2	2	2

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I **12 Hours**

Cloud Computing Software Security Fundamentals- Cloud Information Security Objectives, Cloud Security Services, Relevant Cloud Security Design Principles,

Secure Cloud Software Requirements, Security Concerns, Risk Tolerance, Legal and Regulatory Issues

Unit II **12 Hours**

Cloud Computing Risk Issues and Security Challenges:

The CIA Triad,Privacy and Compliance Risks,Threats to Infrastructure, Data, and Access Control, Cloud Service Provider Risks, Disaster Recovery-Disaster Recovery Planning, Disasters in cloud, Cloud Disaster Management.

Security Challenges- Security Policy Implementation, Virtualization Security Management, VM Security Recommendations, VM-Specific Security Techniques.

Unit III **12 Hours**

Securing the cloud: Architecture and Data Security

Security Requirements for the Architecture, Security Patterns and Architectural Elements, Cloud Security Architecture, Planning Key Strategies for Secure Operation.

Unit IV **12 Hours**

Overview of Data Security in Cloud Computing, Data Encryption: Applications and Limits, Cloud Data Security: Sensitive Data Categorization, Cloud Data Storage, Cloud Lock-in, Key strategies to secure the cloud, Best practices for cloud computing, security monitoring

Total: 48 Hours

Textbook (s)

1. Russell Dean Vines and Ronald L. Krutz, Cloud Security: A Comprehensive Guide To Secure Cloud Computing, Wiley India Pvt Ltd, 2010
2. Vic Winkler (J.R.), Securing The Cloud: Cloud Computing Security Techniques and Tactics, Syngress/Elsevier ,2011

Reference (s)

1. Thomas Erl, Cloud Computing Design Patterns, Prentice Hall, 2015
2. Barrie Sosinsky, Cloud Computing Bible, Wiley India, 2011
3. Rajkumar Buyya, James Broberg, Andrzej M. Goscinski, Cloud Computing: Principles and Paradigms, Wiley, 2011

Internal Assessment Pattern

Cognitive Level	Int. Test 1 (%)	Int. Test 2 (%)	Open Book Exam
Remember	50	40	-
Understand	30	40	-
Apply	20	20	50
Analyze	--	--	50
Evaluate	--	--	-
Create	--	--	-
Total (%)	100	100	100

Remember

1. Define Cloud Computing.
2. List Cloud Service Models
3. List Cloud Information Security Objectives
4. What is cloud foundry?

Understand

1. Identify Cloud Computing Risk Issues and Security Challenges
2. Explain the Data Security
3. Explain all session management techniques
4. Explain different cloud service models

Apply

1. Demonstrate the process for war file deployment in the cloud.
2. How does Sensitive Data Categorization happens in Cloud?

Analyze

1. Justify your answer which technique is used for Session management.
2. Deploy Security Challenges

23IT017 Social Network Analysis (Elective V)

3 0 0 3

Course Outcomes

At the end of the course, students will be able to

1. Acquire Knowledge to analyze Social Networks
2. Model, Aggregate and Represent Knowledge for Semantic Web
3. Use Extraction and Mining tools for Social Networks
4. Apply Reality Mining to Predict Human Behaviors for Social Communities
5. Apply various Algorithms for Evolution and Opinion Mining in Social Networks
6. Write algorithms and systems for expert location in social networks

CO – PO Mapping

COs	PO2	PO4	PO5	PO12
1	2	2	3	2
2	2	2	3	2
3	1	1	3	3
4	1	1	2	2
5	2	2	3	2
6	2	2	2	2

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Syllabus

Unit I **12 Hours**

Introduction

Introduction to Web, Limitations of current Web, Development of Semantic Web, Emergence of the Social Web, Statistical Properties of Social Networks, Network analysis, Development of Social Network Analysis-Key concepts and measures in network analysis, Discussion networks-Blogs and online communities-Web-based networks

Case Studies of Social Network sites like Facebook-Twitter-Linkedin etc.

Unit II **12 Hours**

Evolution

Evolution in Social Networks– Framework, Tracing Smoothly Evolving Communities, Models and Algorithms for Social Influence Analysis, Influence Related Statistics, Social Similarity and Influence, Influence Maximization in Viral Marketing, Link Prediction in Social Networks, Feature based Link Prediction

Neurons &Social Actor-Advanced techniques in Link Prediction

Unit III **12 Hours**

Mining Communities and Opinion Mining

Applications of Community Mining Algorithms, Node Classification in Social Networks

Opinion Extraction–Sentiment Classification and Clustering, Temporal Sentiment Analysis-Irony Detection in Opinion Mining-Wish Analysis–Product Review Mining–Review Classification

Tracking Sentiments towards Topics over Time

Unit IV **12 Hours**

Modelling and Visualization

Visualizing Online Social Networks, A Taxonomy of Visualizations, Graph Representation-Centrality-Clustering-Node-Edge Diagrams-Visualizing Social Networks with Matrix-Based Representations-Node-Link Diagrams, Hybrid Representations, Modelling and Aggregating Social Network Data, Random Walks and their Applications, Ontological representation of Social Individuals and Relationships.

Use of Hadoop and MapReduce-Gephi

Total: 48 Hours

Textbook (s)

5. Charu C. Aggarwal, Social Network Data Analytics, Springer, 2011
6. Peter Mika, Social Networks and the Semantic Web, Springer, 1st Edition, 2007.

Reference (s)

1. BorkoFurht, Handbook of Social Network Technologies and Applications, Springer, 1st Edition, 2010.
2. GuandongXu,Yanchun Zhang and Lin Li, Web Mining and Social Networking, Techniques and applications, Springer, 1st Edition, 2011.
3. Giles, Mark Smith, John Yen, Advances in Social Network Mining and Analysis, Springer, 2010.
4. Ajith Abraham, Aboul Ella Hassanien, VáclavSnášel, Computational Social Network Analysis: Trends, Tools and Research Advances, Springer, 2009.

SAMPLE QUESTION (S)

Internal Assessment Pattern

Cognitive Level	Int. Test 1 (%)	Int. Test 2 (%)	Comprehensive Test1 (%)
Remember	35	20	-
Understand	35	30	-
Apply	20	40	70
Analyze	10	10	20
Evaluate	--	-	10
Create	--	-	-
Total (%)	100	100	100

Sample Question (S)

Remember

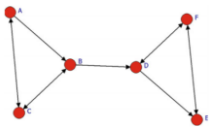
1. What are the limitations of current web?
2. List the statistical properties of social networks
3. Define social similarity.
4. What are the two different measures in influence related statistics

Understand

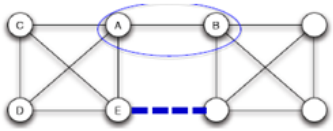
1. Explain about development of semantic web in research.
2. Demonstrate web based networks.
3. Illustrate the ways of visualizing social networks with matrix based representation and node – link diagram.
4. Explain core methods of community detection and mining.
5. Explain algorithms and systems for expert location in social networks.
6. Outline the models and algorithms for social influence analysis.

Apply

1. Make use of the following graph to solve Degree Centrality, Between-ness Centrality and Cliques.

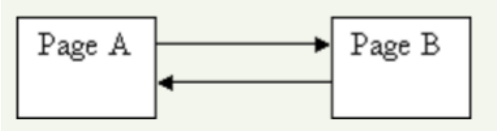


2. Solve Path, Length and Distance between nodes (i.e., Density measures) using SNA Data Processing Tools.
3. Make use of the above graph to solve network density and cut points.
4. Develop random graphs using Erdos Reini Model
5. Solve Weak Ties,Traidic closure for the following graph.

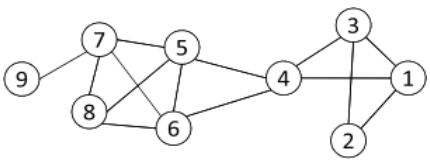


Analyse

1. Compute PageRank for the following figure.



2. Discover Spectral Clustering steps for the following graph.



3. Examine the ways to visualize community's hubs and node centrality measures using sna package.
4. Analyze positive, moderate and negative feeds using sentiment analysis on twitter data.

Open Book Exam Questions

1. Visualization of team data in Gephi-What should be my considerations?
2. How do we calculate data processing time using cloud analyst?

23ML001 Human Computer Interaction (Elective V)

3 0 0 3

Course Outcomes

1. Demonstrate the capabilities of both humans and computers from the viewpoint of human information processing.
2. Interpret typical human–computer interaction (HCI) models, styles, and various historic HCI paradigms.
3. Choose an interactive design process and universal design principles to designing HCI systems
4. Make use of HCI design principles, standards and guidelines.
5. Examine and identify user models, user support, socio-organizational issues, and stakeholder requirements of HCI systems
6. Categorize and discuss HCI issues in groupware, ubiquitous computing, virtual reality, multimedia, and Word Wide Web-related environments.

COs–POs Mapping

COs	PO1	PO2	PO3	PO12	PSO1
1	1	1	1	1	2
2	1	1	1	1	2
3	2	2	2	1	2
4	2	1	2	1	2
5	2	2	1	1	2
6	2	2	2	1	2

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I **12 Hours**

Introduction

Importance of user Interface-definition, importance of good design, Benefits of good design, A brief history of Screen design, The graphical user interface-popularity of graphics, the concept of direct manipulation, graphical system, Characteristics, Web user-Interface popularity, characteristics-Principles of user interface.

Unit II **12 Hours**

Design process

Human interaction with computers, importance of human characteristics human consideration, Human interaction speeds, understanding business junctions, Screen Designing:-Design goals-Screen planning and purpose, organizing screen elements, ordering of screen data and content-screen navigation and flow-Visually pleasing composition amount of information-focus and emphasis-presentation information simply and meaningfully-information retrieval on web-statistical graphics-Technological consideration in interface design.

Unit III **12 Hours**

Windows

New and Navigation schemes selection of window, selection of devices based and screen based controls, Components-text and messages, Icons and increases-Multimedia, uses problems, choosing colors.

Unit IV **12 Hours**

Software tools &Interaction Devices

Specification methods, interface-Building Tools, Keyboard and function keys, pointing devices-speech recognition digitization and generation-image and video displays-drivers.

Total: 48 Hours

Textbook (s)

1. Wilbert O Galitz, Wiley Dream Tech, The essential guide to user interface design, 3rd Edition, Wiley Computer Publishing, 2007
2. Ben Shneidermann, Designing the user interface, 3rd Edition, Pearson Education Asia, 2008

Reference (s)

1. Alan Dix, Janet Fincay, GreGoryd, Abowd, Russell Bealg, Human Computer Interaction., 4th Edition, Pearson Education, 2018

Internal Assessment Pattern

Cognitive Level	Int. Test 1 (%)	Int. Test 2 (%)	Open Book Examination(%)
Remember	40	30	--
Understand	40	40	--
Apply	20	30	50
Analyze	--	--	50

Evaluate	--	--	--
Create	--	--	--
Total (%)	100	100	100

SAMPLE QUESTION (S)

Remember
1. Define Human Computer Interaction.
2. What is the basic goal of Human Computer Interaction?
3. State the long term goal of HCI
4. List the factors HCI designers must consider for User Interface Design

Understand
1. Demonstrate the HCI importance.
2. Extend the Trouble faced by human with Computers.
3. Illustrate Psychological Responses to Poor Design
4. Interpret Human-factor variables used in Comparison of pointing devices.

Apply
1. Identify the amount of Information need to Present in a page.
2. Organize the different Software tool specification methods.
3. Model out the six types interaction tasks properties of Pointing devices

Analyze
1. Distinguish the Focus and Emphasis of visual elements in web page design.
2. Categorize the Formatting of Menus.
3. Analyze organizational structure of Keyboard layouts

23CS012 Wireless Adhoc Network (Elective-V)

3 0 0 3

Course Outcomes

1. Explain the various challenging issues of wireless Adhoc networks.
2. Apply the MAC protocols using reservation and scheduling mechanisms.
3. Apply the routing protocols and transport layer protocol using various TCS mechanisms in wireless Adhoc networks.
4. Explain the issues of MAC and network layer solution to provide quality of service.
5. Explain the significances of various battery management schemes for wireless Adhoc networks.
6. Analyze the security requirements and secure routing in wireless Adhoc networks.

COs – POs Mapping

CO's	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	1	-	-	2	1	-	-	2	-	-	-	-
CO2	3	3	2	-	-	1	2	-	-	1	-	-	-	-
CO3	3	3	3	-	-	2	3	-	-	1	-	-	-	-
CO4	3	3	2	-	-	2	1	-	-	2	-	-	-	-
CO5	3	3	2	-	-	1	1	-	-	2	-	-	-	-
CO6	3	2	3	-	-	3	3	-	-	2	-	-	-	-

3–Stronglylinked|2–Moderatelylinked|1–Weaklylinked

UNIT I **12 Hours**

Wireless AdHoc Networks: Introduction, Properties, applications, limitations, Issues in AdHoc Wireless Networks, Ad Hoc Wireless Internet. **MAC Protocols:** Introduction, Issues in Designing a MAC protocol for AdHoc Wireless Networks, Design goals of a MAC Protocol for Ad Hoc Wireless Networks, Classifications of MACProtocols, Contention-Based Protocols, Contention-Based Protocols with reservation Mechanisms.
Contention–Based MAC Protocols with Scheduling Mechanisms

UNIT II **12 Hours**

RoutingProtocols: Introduction,IssuesinDesigningaRoutingProtocolforAdHocWirelessNetworks,Classification of Routing Protocols, Proactive/ Table–Driven Routing Protocols, Reactive/ On–Demand Routing Protocols, Hybrid Routing Protocols, Hierarchical Routing Protocols, Power – Aware Routing Protocols. **TransportLayer:** Introduction, Issues in Designing a Transport Layer Protocol for Ad Hoc Wireless Networks, Design Goals of a Transport Layer Protocol for Ad Hoc Wireless Networks, Classification of Transport Layer Solutions, TCPOverAdHoc Wireless Networks.
OtherTransportLayerProtocolforAd HocWirelessNetworks.

UNIT III **12 Hours**

QualityofService: Introduction,IssuesandChallengesinProvidingQoSinAdHocWirelessNetworks,Classification of QoS Solutions, MAC Layer Solutions, Network Layer Solutions, QoS Frameworks for Ad HocWirelessNetworks.**EnergyManagement:**Introduction,NeedforEnergyManagementinAdHocWirelessNetworks,ClassificationofAdHocWirelessNetworks,BatteryManagementSchemes,TransmissionPowerManagementSchemes.
System Power Management Schemes.

UNIT IV **12 Hours**

Security Protocols: Network Security Requirements, Issues and Challenges in Security Provisioning,Network Security Attacks, Key Management, Secure Routingin AdHoc Wireless Networks.

WirelessSensorNetworks: Introduction, Sensor Network Architecture, Data Dissemination, Data Gathering, Location Discovery.
Quality of a Sensor Network, Evolving Standards, Other Issues

Total: 48 Hours

Textbook(s)

1. AdHoc Wireless Networks: Architectures and Protocols-C.Siva Ram Murthy and B.S.Manoj,2004,PHI.
2. WirelessAd-hocandSensorNetworks:Protocols,PerformanceandControl-JagannathanSarangapani,CRCPress

Reference(s)

1. WirelessSensorNetworks-C.S.Raghavendra,KrishnaM.Sivalingam,2004,Springer.
2. Ad-HocMobileWirelessNetworks:Protocols&Systems,C.K.Toh,Firsted.PearsonEducation

SAMPLEQUESTION(S)

Internal Assessment Pattern

Cognitive Level	Int.Test1 (%)	Int.Test2 (%)	OBE(%)
Remember	40	50	20
Understand	35	30	20
Apply	25	20	60
Analyze	--	--	--
Evaluate	--	--	--
Create	--	--	--
Total(%)	100	100	100

Remember

1. Find out the major significance of cellular wireless networks and Adhoc wireless networks.
2. List few latest application systems of Adhoc wireless networks.
3. Identify the challenging issues in Adhoc network configuration and maintenance.
4. Show the proactive steps to solve the hidden terminal problem.
5. List the transmission impediments of wireless channel.

Understand

1. Summarize the issues that affect the deployment and performance of adhoc wireless system.
2. Outline the benefits when deployment of a commercial adhoc wireless networks compared to wired network.
3. Compare Ad-hoc Networks and Wireless LAN.
4. Compare the efficiency of the packet queuing mechanism adopted in MACA and MACAW.
5. Relate contention-based protocols with reservation mechanisms and contention-based protocols with scheduling mechanisms by means of packet delivery, reception, and acknowledgment ratio.

Apply

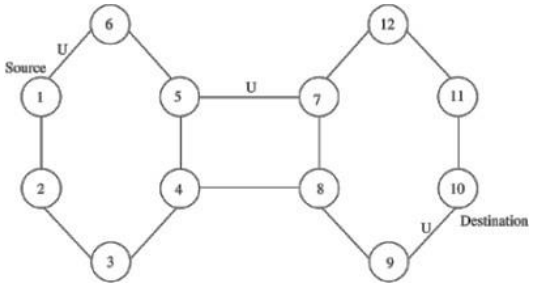
1. Calculate the maximum achievable data rate over a 9 KHz channel whose signal-to-noise ratio is 20 dB.
2. Think of four scenarios where wireless networks can replace wired networks in order to improve the efficiency of people at their workplace. Briefly describe how in each case a wireless network will fit the role better than a wired network.

Analyze

1. Discuss the deployment scenarios for various HIPERLAN standards in the ETSI BRAN system.
2. Compare the EY-NPMA and the CSMA/CA mechanisms.

Evaluate

1.Consider the topology given in below to simulate DSR,SSA,andABR protocols for path establishment from node 1 to node 10, find the paths found and the ratio of the number of *Route Request*packetssentinthenetwork.(Linkslabeled"U"refertounstableones.)



2. Justify whether the LCC algorithm (when run consistently with node degrees or node IDs) gives a deterministic result? If so, prove the above fact. Otherwise, give a counter-example.
3. Discuss how security provisioning in ad hoc wireless networks differs from that in infrastructure-based networks.?

23ML002 Large Language Models (Elective VI)

3 0 0 3

Course Outcomes

1. Understand the fundamentals & building blocks of LLMs
2. Exemplify basic data processing techniques.
3. Understand prompt engineering mechanism.
4. Understand the properties of LLMs.
5. Evaluate the performance of various LLMs.
6. Elaborate various case studies of LLMs.

CO–PO Mapping

COs	PO1	PO2	PO4	PO12	PSO1
1	3	2	1	1	1
2	2	3	1	3	1
3	3	2	2	3	2
4	2	3	1	3	3
5	1	3	2	3	3
6	1	2	3	1	1

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I **12 Hours**

Introduction to Language Models, History of Language models, Inspiration to LLM, Understanding LLMs : Human Thought Versus LLM Processing, Transformer architecture: Understanding the Architecture , Key Components of the Transformer Model and Practical Applications of Transformers.

Unit II **12 Hours**

Dataset preparation - probabilistic tokenization, Training - RLHF, instruction tuning, MoE, context window, Multimodality of LLMs. Prompt Engineering : Definition of prompt and prompt engineering, Need of Prompt Engineering, Elements of a Prompt, Designing Prompts for Different Tasks, Prompt Engineering Techniques, Risks.

Unit III **12 Hours**

Properties of LLM - Scaling laws, Emergent abilities, Interpretation of LLMs, Evaluation of performance - Perplexity, BPW, BCT & BPT, Task specific benchmarks

Unit IV **12 Hours**

Case studies of popular LLMs - GPT-1/2/3/3.5, BERT/PaLM/Gemini, OPT/LlaMA

Total : 48 Hours

Articles / Reading Materials

1. "Better Language Models and Their Implications". OpenAI. 2019-02-14. Archived from the original on 2020-12-19. Retrieved 2019-08-25.
2. Vaswani, Ashish; Shazeer, Noam; Parmar, Niki; Uszkoreit, Jakob; Jones, Llion; Gomez, Aidan N; Kaiser, Łukasz; Polosukhin, Illia (2017). "Attention is All you Need" (PDF). Advances in Neural Information Processing Systems. Curran Associates, Inc. 30.
3. Merritt, Rick (2022-03-25). "What Is a Transformer Model?". NVIDIA Blog. Retrieved 2023-07-25.
4. Rogers, Anna; Kovaleva, Olga; Rumshisky, Anna (2020). "A Primer in BERTology: What We Know About How BERT Works". Transactions of the Association for Computational Linguistics. 8: 842–866. arXiv:2002.12327. doi:10.1162/tacl_a_00349. S2CID 211532403.
5. Huyen, Chip (18 October 2019). "Evaluation Metrics for Language Modeling". The Gradient.
6. Wiggers, Kyle (28 April 2022). "The emerging types of language models and why they matter". TechCrunch.

Internal Assessment Pattern

Cognitive Level	Int. Test 1 (%)	Int. Test 2 (%)	Open Book Exam (%)
Remember	40	30	--
Understand	40	40	--
Apply	20	30	50
Analyze	--	--	50
Evaluate	--	--	--
Create	--	--	--
Total (%)	100	100	100

23CS015 Software Project Management (Elective VI)

3 0 0 3

Course Outcomes

1. Explain the concepts and issues of Software Project Management
2. Illustrate various Software Architectures
3. Classify Software Risks and Risk Management Strategies
4. Design effective software development model to meet organizational needs
5. Experiment with appropriate methodologies to develop a project schedule
6. Experiment with appropriate techniques to assess ongoing project performance

COs – POs Mapping

CO's	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	2	-	-	-	-	-	-	-	2	2	-	2
CO2	3	2	2	-	-	-	-	-	-	-	2	2	-	3
CO3	3	2	2	-	-	-	-	-	-	-	2	2	-	3
CO4	3	2	3	-	-	-	-	-	-	-	2	2	-	3
CO5	3	2	3	-	-	-	-	-	-	-	3	2	-	3
CO6	3	2	3	-	-	-	-	-	-	-	3	2	-	3

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I **12 Hours**

Conventional Software Management: The Waterfall Model, Conventional Software Management Performance. **Transition:** The principles of conventional software Engineering, Principles of Modern software management, transitioning to an iterative process.

Evolution of Software Economics: Software Economics, Software Cost Estimation. **Improving Software Economics:** Reducing Software Product Size, Improving Software Processes, Improving Team Effectiveness, Improving Automation, Achieving Required Quality, Peer Inspections.

Software Development Phases, Procedural vs Object Oriented Paradigm.

Unit II **12 Hours**

Life Cycle Phases: Engineering and Production Stages, Inception, Elaboration, Construction, Transition Phases.

Artifacts of Software Process: The Artifact Sets, Types of Artifacts.

Software Architectures: A Management Perspective and Technical Perspective,

Workflows of the process: Software Process Workflows, Iteration Workflows.

Off-the-Shelf Components, Requirement Specification Document, Methods of Specification.

Unit III **12 Hours**

Checkpoints of the Process: Major Milestones, Minor Milestones, Periodic Status Assessments.

Iterative Process Planning: Work breakdown Structures, Planning Guidelines, Cost and Schedule Estimating process, Iteration Planning Process.

Project Organizations and Responsibilities: Line-of-Business Organizations, Project Organizations, Evolution of Organizations.

Process Automation: Automation Building Blocks, The Project Environment.

CASE Tools for Requirements Specification, Planning and Estimation, Pragmatic Planning.

Unit IV **12 Hours**

Project control and Process Instrumentation: The Seven Core Metrics, Management Indicators, Quality Indicators, Life Cycle Expectations, Pragmatic Software Metrics, Metrics Automation, Tailoring the Process.

Future Software Project Management: Modern Project Profiles, Next Generation Software Economics,The COCOMO Cost Estimation Model: Basic COCOMO

Bugs of testing, Bug tracking tools

Total: 48 Hours

Textbook (s)

1. Walker Royce, "Software Project Management – A UnifiedFramework", 1stEdition, Pearson Education, 2005.
2. PankajJalote, "Software Project Management in Practice", 1stEdition, Pearson Education, 2002.

Reference (s)

1. Bob Hughes, "Mike Cotterell, Rajib Mall, Software ProjectManagement", 5thEdition, McGraw-Hill Higher Education, 2011.
2. Joel Henry, "Software Project Management", 1st Edition, Pearson Education, 2004.

3. Norman E. Fenton,Shari Lawrence Pfleeger, "Software Metrics: A Rigorous and Practical Approach ", 1st Edition, PWS Publishing Company, 1997.

Internal Assessment Pattern

Cognitive Level	Int. Test 1 (%)	Int. Test 2 (%)	Open Book Exam (%)
Remember	40	30	--
Understand	40	40	--
Apply	20	30	50
Analyze	--	--	50
Evaluate	--	--	--
Create	--	--	--
Total (%)	100	100	100

SAMPLE QUESTION (S)

Remember

1. Define Software Process
2. List the phases of Software Development.
3. What is an Artifact?
4. What is Process Automation?

Understand

1. Explain various formal methods of specification.
2. Explain COCOMO II Model.
3. Explain Critical Path Methods to schedule a software project.

Apply

1. Identify the amount of Information need to present in a page.
2. Organize the different Software tool specification methods.
3. Assume that the size of an organic type software product has been estimated to be 32,000 lines of source code. Assume that the average salary of software engineers be Rest. 15,000/- per month. Determine the effort required to develop the software product and the nominal development time.

Open Book Questions

1. Calculate the effort and development time for the model related to your mini-project which was developed in previous semester using COCOMO basic cost estimation model. (Take your mini-project size in KLOC)[Apply]
2. Analyze your mini-project with respect to changes using Type 0, Type 1, Type 2, Type 3 & Type 4 categories. [Apply]

23ML002 Reinforcement Learning (Elective VI)

3 0 0 3

Course Outcomes

The student will be able to

1. Understand the key principles and terminology in reinforcement learning
2. Understand how real-world problems can be structured as Markov Decision Processes
3. Describe the use of dynamic programming in reinforcement learning
4. Apply model-free reinforcement learning methods in reinforcement learning tasks.
5. Explain the concepts behind policy gradient methods and deep learning-based approaches in reinforcement learning.
6. Identify various advanced reinforcement learning methods and summarize their applications in different fields.

CO–PO Mapping

COs	PO1	PO2	PO3	PSO1	PSO2
1	2	1	1	1	1
2	2	2	2	1	1
3	1	1	2	1	1
4	1	3	2	2	1
5	1	3	2	2	1
6	1	3	2	2	1

Unit 1: Introduction to Reinforcement Learning and Markov Decision Processes **12 Hours**

Introduction to RL terminology: Agent, Environment, Action, State, Reward, Episode -Taxonomy of RL methods - Reinforcement Learning Framework: Agent-environment interface, Goals and rewards. Markov Decision Processes (MDPs): Understanding states, actions, and rewards, Episodes in RL, Markov property - Dynamics and Returns: Transition probabilities and dynamics of MDPs, Calculation of returns.

Real world applications of reinforcement learning.

Unit 2: Dynamic Programming and Monte Carlo Methods **12 Hours**

Policy and Value Functions - Bellman equations for value functions -Dynamic Programming in RL: Policy iteration and value iteration, Limitations of dynamic programming - Monte Carlo Methods in RL: Monte Carlo Policy Evaluation, Monte Carlo Control, Control without Exploring Starts, Importance Sampling in Policy Prediction, Incremental Implementation, Policy Monte Carlo Control.

Eligibility Traces

Unit 3: Model-Free Control Methods **12 Hours**

Temporal Difference Learning: TD learning Algorithms: TD(0) and TD(λ) algorithms. Model-Free Control Algorithms: SARSA, Q-learning, Double Q-learning, n-step Bootstrapping methods - Exploration vs. exploitation strategies: Exploration Strategies in RL, Exploration-exploitation trade-off.

Function approximation, hierarchical reinforcement learning

Unit 4: Advanced Algorithms and Strategies **12 Hours**

Deep Reinforcement Learning: Introduction to Deep Q-Networks (DQN), Experience Replay, Target Networks - Policy Gradient Methods: Basics of policy gradients, REINFORCE algorithm - Actor-Critic Methods: Combining policy gradients with value functions, Actor-critic algorithms, Function Approximation in Actor-Critic - Proximal Policy Optimization (PPO).

Policy Approximation, Multi-Agent Reinforcement Learning

Total: 48 Hours

Textbooks:

1. Richard S. Sutton and Andrew G. Barto, "Reinforcement Learning: An Introduction," Second Edition, The MIT Press, 2019.
2. Marco Wiering and Martijn Van Otterlo, "Reinforcement Learning: State-of-the-Art," Adaptation, Learning, and Optimization, Vol. 12, Springer, 2012.
3. Maxim Lapan, "Deep Reinforcement Learning Hands-On," Second Edition, Packt Publishing, 2018.

Reference Books:

1. Csaba Szepesvári, "Algorithms for Reinforcement Learning," Synthesis Lectures on Artificial Intelligence and Machine Learning, Morgan & Claypool Publishers, 2009.
2. Ian Goodfellow, Yoshua Bengio, and Aaron Courville, "Deep Learning," MIT Press, 2016.
3. Yuxi Li, "Reinforcement Learning Applications", arXiv preprint arXiv:1908.06973v1, 2019.

Internal Assessment Pattern

Cognitive Level	Int. Test 1 (%)	Int. Test 2 (%)	Assignment Test (%)
Remember	40	30	--
Understand	40	40	20
Apply	20	30	50
Analyze	--	--	30
Evaluate	--	--	--
Create	--	--	--
Total (%)	100	100	100

23SIX02 Summer Internship II

0 0 0 1.5

Course Outcomes

1. Demonstrate communication skills to meet the requirement of industry
2. Develop logical thinking and analytical skills to thrive in competitive examinations
3. Use mathematical concepts to solve technical quizzes
4. Develop technical skills to work out real time problems
5. Develop algorithms for different applications
6. Solve industry defined problems using appropriate programming skills

CO–PO Mapping

CO's	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	2	1	2	1	2	3	1	3	3	3	1	3	3	1
CO2	3	3	3	3	2	2	1	3	3	1	1	3	3	2
CO3	3	2	3	2	2	1	1	3	2	3	1	2	3	1
CO4	3	3	3	3	3	3	2	3	3	3	1	3	3	2
CO5	3	3	3	3	3	3	3	3	3	3	2	3	3	2
CO6	3	3	3	3	3	3	3	2	3	3	3	3	3	2

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

23PWX01Project Work

0 0 16 8

Course Outcomes

1. Identify a contemporary engineering application to serve the society at large
2. Use engineering concepts and computational tools to get the desired solution
3. Justify the assembled/fabricated/developed products intended.
4. Organize documents and present the project report articulating the applications of the concepts and ideas coherently
5. Demonstrate ethical and professional attributes during the project implementation.
6. Execute the project in a collaborative environment.

CO–PO Mapping

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**
CO1	3	3	2	3	2	3	3	2	3	3	3	3	3	3
CO2	3	3	3	3	3	3	2	3	3	3	3	3	3	3
CO3	3	3	3	3	2	3	2	3	2	3	3	2	3	3
CO4	3	3	3	3	2	2	3	3	2	3	3	2	3	3
CO5	3	3	3	3	3	3	3	3	3	3	3	3	3	3
CO6	3	3	3	3	3	3	3	3	3	3	3	3	3	3

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

8th Semester

23CS018 Information Retrieval Systems (Elective VIII)

0 0 0 3

Course Outcomes

1. Explain the Objectives of Information Retrieval Systems and its relation to Digital Libraries and Data warehouse
2. Illustrate various System Search Browse and Miscellaneous Capabilities
3. Explain Automatic Indexing, Information Extraction methods
4. Demonstrate Inverted file structures, N-gram data structure, and PAT data structure
5. Interpret Classical techniques of Information Retrieval, and additional techniques employed by Web search engines
6. Develop a small information retrieval system by using user search techniques and text search Algorithms

COs – POs Mapping

CO's	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	2	2	-	-	-	-	-	3	-	-	-	3
CO2	3	2	2	2	-	-	-	-	-	2	-	-	-	3
CO3	2	3	1	1	-	-	-	-	-	2	-	-	-	3
CO4	2	3	1	1	-	-	-	-	-	2	-	-	-	3
CO5	2	3	1	1	-	-	-	-	-	2	-	-	-	3
CO6	3	3	2	2	-	-	-	-	-	2	-	-	-	3

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I **11 Hours**

Introduction: Definition, Objectives, Functional Overview, Relationship to DBMS, Digital libraries and Data Warehouses.

Information Retrieval System Capabilities: Search, Browse, Miscellaneous.

Miscellaneous capabilities.

Unit II **13 Hours**

Cataloging and Indexing: Objectives, Indexing Process, Automatic Indexing, Information Extraction.

Data Structures: Introduction, Stemming Algorithms, Inverted file structures, N-gram data structure, PAT data structure, Signature file structure, Hypertext data structure.

Signature file structure, Hypertext data structure.

Unit III **12 Hours**

Automatic Indexing: Classes of automatic indexing, Statistical indexing, Natural language, Conceptindexing, Hypertext linkages

Document and Term Clustering: Introduction, Thesaurus generation, Item clustering, Hierarchy of clusters.

Measurement Example-TREC Results.

Unit IV **12 Hours**

User Search Techniques: Search statements and binding, Similarity measures and ranking, Relevance feedback, Selective dissemination of information search, Searching the Internet and hypertext, Information Visualization

Text Search Algorithms: Introduction, Software text search algorithms, Hardware text search systems.

Hardware text search systems

Total: 48 Hours

Textbook (s)

1. Kowalski, Gerald, Mark T Maybury: Information Retrieval Systems: Theory and Implementation, Kluwer Academic Press, 1997.
2. http://nlp.stanford.edu/IR-book/html/htmledition/evaluation-in-information-retrieval-1.html

Reference (s)

1. Frakes, W.B. and RicardoBaeza Yates, Information Retrieval Data Structures and Algorithms, 1st Edition Prentice Hall, 1992.
2. Robert Korfhage, Information Storage & Retrieval, 1st Edition, John Wiley & Sons, 2005

Internal Assessment Pattern

Cognitive Level	Int. Test 1 (%)	Int. Test 2 (%)	Open Book Exam (%)
Remember	30	30	--
Understand	40	40	--
Apply	10	20	80
Analyze	20	10	20

Evaluate	--	--	--
Create	--	--	--
Total (%)	100	100	100

SAMPLE QUESTION (S)

Remember

1. Define measures for IRS
2. Define PAT
3. State Browse capabilities
4. List automatic indexing
5. Recall term clustering algorithms

Understand

1. Explain Functional Overview
2. Explain Stemming algorithms
3. Summarize Brute force algorithms
4. Illustrate S/w text search Algorithms
5. Interpret standard test collection
6. Classify S/w & H/w text search system

Apply

1. Demonstrate Precision
2. Design a hybrid Indexing Data Structure
3. Use stemming Algorithms for IRS

Analyze

1. Analyze s/w and h/w indexing methods
2. Analyze search and Browse capabilities
3. Compare Pat and Stemming algorithms
4. Analyze Term clustering Algorithms

Evaluate

1. Determine Which of the indexing algorithm is efficient
2. Check all automatic indexing methods for its efficiency
3. Compare S/w text search and H/w text search algorithms

23CS019 Fundamentals of DevOps (Elective VIII)

0 0 0 3

Course Outcomes:

1. Illustrate the need of Improvement and value of DevOps
2. Outline the Value Stream using DevOps
3. Describe the Organizational Change and transformation
4. Illustrate the Concept and Goal of Accelerate Flow
5. Outline Feedback Loops and its usage
6. Demonstrate the Concept and Goal of Learning

COs – POs Mapping

CO's	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	3	-	-	-	-	-	-	-	3	-	-	3
CO2	3	3	2	-	-	-	-	-	-	-	2	-	-	3
CO3	3	2	2	-	-	-	-	-	-	-	2	-	-	3
CO4	3	2	3	-	-	-	-	-	-	-	2	-	-	3
CO5	3	3	3	-	-	-	-	-	-	-	2	-	-	3
CO6	3	2	3	-	-	-	-	-	-	-	2	-	-	3

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

UNIT I **12 Hours**

Introduction to DevOps: Understanding Improvement, The Convergence, History, and Value of DevOps; **Understanding the Value Stream using DevOps:**Analyzing the technology Value Stream; The ways of DevOps: Flow, Feedback Loops, Culture of Continual Experimentation and Learning;

Infrastructure as Code (IaC), Versioning and Reusability

UNIT II **12 Hours**

Value Stream in DevOps: Concept and Goals of Streams, Picking a Value Stream, Understanding Organizational Change, Enabling Transformation;

Canary Deployments, Service Mesh

UNIT III **13 Hours**

Accelerate Flow: Concept and Goal of Accelerate Flow, Continuous Delivery Patterns and Practices, the Deployment Pipeline, Creating Consistency in the Pipeline, Automated Testing, Deployment Strategies;
Amplifying the Feedback Loops: Concept and Goals of Feedback Loops, Creating a Service Reliability Culture, Fast Feedback, Understanding Monitoring, Understanding Complexity

Incident Response and Management, Resilience Engineering

UNIT IV **11 Hours**

Learning Acceleration with DevOps: Concept and Goal of Learning, Learning Organizations, Communication, Blameless Culture.

Psychological Safety, Systems Thinking.

Total: 48 Hours

Text Books:

1. Kim, G., Behr, K., and Spafford, G. (2013). The Phoenix Project: A Novel About IT, DevOps, and Helping Your Business Win. IT Revolution Press
2. Kim, G., Humble, J., Debois, P., and Willis, J. (2016). The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations. IT Revolution Press

Reference Books:

1. JulianFish ,ThePracticalGuide toEnterpriseDevOps andContinuousDelivery

Internal Marks Assessment:

CognitiveLevel	Int. Test 1(%)	Int. Test 2(%)	OBE Test (%)
Remember	25	25	
Understand	40	30	40
Apply	20	20	30
Analyze	15	20	30
Evaluate	--	--	--
Create	--	--	--
Total (%)	100	100	100

Sample Questions

Remember

1) What is DevOps
2) Define Value Stream
3) What is **Accelerate Flow**

Understand

1) Explain about Ways of DevOps
2) Explain Picking Value Stream min DevOps
3) Describe Feedback Loops

Apply

1) Give Some Real Time Applications of DevOps
2) Design a deployment pipeline using automated testing and deployment strategies.
3) Show how blameless culture and psychological safety improve incident response.

23DS003 Cyber Security (Elective VIII)

0 0 0 3

Course Outcomes

1. Explain the fundamental concepts of Cyber security
2. Demonstrate the web security and different attacks
3. Identify different network scanning and security measures
4. List out different types of Intrusion detection.
5. Model different types of Intrusion prevention systems
6. Outline different cyber-crimes, IT laws and acts.

CO–PO Mapping

COs	PO1	PO3	PO4	PO5	PSO1
1	3	3	1	1	1
2	3	3	1	3	1
3	1	2	3	3	2
4	3	1	1	3	3
5	1	3	3	3	3
6	1	3	2	1	1

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I

Introduction to Cyber Security **12 Hours**

What is Cyber Security, its need, cyber-threats, Cyber Warfare-Cyber Crime-Cyber terrorism-Cyber Espionage), Career Growth, Statistics, Inferences, Need for a Comprehensive Cyber Security Policy, Classification of Cyber Crimes, kinds of cybercrimes, Reasons for Cyber Crimes, Cyber Security Tools : Nmap, Metasploit, wireshark, tcpdump, snort.
Cyber security awareness, social engineering, cyber stalking

Unit II

Web Security **12 Hours**

Same origin Policy, Cross Origin Resource Sharing, DDOS, SQL Injection, XSS, Homograph, Generating and storing session tokens.

Networking Scanning & Security Measures:

Packet Sniffing and spoofing, Network scanning types, port scanning & its tools, and Network Architecture
Security Measures :IPtables(firewalls) , Webservers (Nmap & Metasploit for securing webservers),
Cyber Threats and Attacks (Malware, DOS, MITM, Social engineering attacks, Spoofing, Phishing)
Cross-Site Request Forgery (XSRF/CSRF), spear phishing.

Unit III

Intrusion Detection System **12 Hours**

Intruders, Intrusion Detection, Analysis Approaches, Network-Based IDS, Host-Based IDS, signature based IDS, anomaly based IDS, advantages and disadvantages of NIDS and HIDS
Intrusion Detection Tools, snort architecture, snort rules, case studies of intrusion detection systems,Intrusion detection exchange format.
Honeypots, different types of honeypots, benefits and dangers of honeypots
Firewall vs IDS, Physical IDS, honeynet

Unit IV

Cyber Laws and Digital Forensics **12 Hours**

Digital Forensics: Introduction to Digital Forensics, historical background of digital forensics, Forensic Software, and Hardware, need for computer forensics science, special tools and techniques digital forensic life cycle, challenges in digital forensic.
Law Perspective: Introduction to the Legal Perspectives of Cybercrimes and Cyber security, Cybercrime and the Legal Landscape around the World, Why Do We Need Cyber laws, The Indian IT Act, Cybercrime Scenario in India, Digital Signatures and the Indian IT Act.
Cybercrime and Punishment

Total: 48 Hours

Textbook (s)

1. Wenliang Du, Computer & Internet Security: A Hands-on Approach, (2020)
2. William Stallings, Lawrie Brown, Computer Security Principle sand Practice Third Edition,2015
3. Sunit Belapure and Nina Godbole, Cyber Security: Understanding Cyber Crimes, Computer Forensics and Legal Perspectives, Wiley India Pvt. Ltd, 2011.
4. Nelson Phillips and Enfinger Steuart, "Computer Forensics and Investigations", Cengage Learning, New Delhi, 6th edition 2018.

Reference (s)

1. Pande, Jeetendra. "Introduction to Cyber Security.", (2017)
2. Pavan Duggal, Cyber frauds, cybercrimes & law in India. 21 July 2013.
3. Ali A. Ghorbani, Network intrusion detection and prevention concepts and techniques, Springer, 2010
4. Roberto Di Pietro, Luigi V. Mancini (2008), Intrusion Detection System, Springer
5. DafyddStuttard and Marcus Pinto. The Web Application Hacker's Handbook: Finding and Exploiting Security Flaws, Wiley Publication, 2nd edition, 31 Aug 2011.

Internal Assessment Pattern

Cognitive Level	Int. Test 1 (%)	Int. Test 2 (%)	OBE (%)
Remember	50	40	-
Understand	30	40	-
Apply	20	20	50
Analyze	--	--	50
Evaluate	--	--	-
Create	--	--	-
Total (%)	100	100	-

SAMPLE QUESTION (S)

Remember

1. What is cyber security?
2. Define security policy.
3. List classification in cyber security.
4. Define network scanning.
5. What is digital forensics?

Understand

1. Explain security policy for password protection.
2. Explain reasons for cybercrime.
3. Outline SQL injection attack.
4. Classify different approaches for packet filtering using firewall.
5. Explain IP Spoofing.

Apply

1. Apply IPS methods to prevent introduter.
2. Apply snort rules to detect intrusion
3. Identify open ports in a network using namp.
4. Plan to filter unauthorized packets using iptables
5. Build host based IPS.

Analyze

1. Compare and contrast iptables and snort.
2. Compare and contrast different tools to address DOS.
3. Examine different post scanning methods.
4. Distinguish nmap and metasploit.
5. Compare different tools related to digital forensics

23FIX01 Full Semester Internship (FSI)

0 0 0 9

Course Outcomes

1) Use the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems
2) Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences
3) Select appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations
4) Use ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice
5) Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings
6) Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions

COs – POs Mapping

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**
CO1	3	3	3	2	3	2	2	2	2	2	2	2	3	3
CO2	3	3	3	3	3	2	2	2	2	2	2	2	3	3
CO3	3	3	3	2	3	3	3	2	2	2	3	2	3	3
CO4	2	2	2	3	3	3	3	2	2	2	3	2	2	3
CO5	2	2	3	3	3	3	3	2	2	3	3	3	3	3
CO6	3	3	3	3	3	3	3	2	2	3	3	3	3	3

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Department of CSE – Artificial Intelligence & Machine Learning

AR23 - Career Path Courses

Sem	Course Code	Course	POs &PSOs	L	T	P	Credits
Career Path-1: Machine Learning Operations (ML Ops)							
5	23MLC11	Computer Vision & Pattern Recognition	1,2,3,12,PSO1	3	0	0	3
6	23MLC12	Machine Learning for Business Intelligence	1,2,3,4,5,PSO1,PSO2	3	0	2	4
7	23MLC13	Conversational AI	1,2,4,12,PSO1	3	0	0	3
Career Path-2: Full Stack Developer							
5	23CSC21	Backend Programming Languages	1,2,3,5,PSO1,PSO2	3	0	0	3
6	23CSC22	Web Application Frameworks	1,2,3,5,PSO1, PSO2	3	0	2	4
7	23CSC23	Web Application Databases	1,3,5,PSO1	3	0	0	3
Career Path-3: Cloud Computing							
5	23MLC31	Fundamentals of Cloud Computing	1,2,3,4,5,PSO1,PSO2	3	0	0	3
6	23MLC32	Cloud Services using AWS	1,2,3,4,5,PSO1	3	0	2	4
7	23MLC33	Cloud Security Essentials	1,2,3,4,5,PSO1	3	0	0	3

Career Path 1 (Machine Learning Operations (ML Ops))
23MLC11 Computer Vision & Pattern Recognition (ML-Ops Career path)

3 0 0 3

Course Outcomes

1. Explain fundamental computer vision concepts, including applications and challenges.
2. Understand image processing techniques and their application in computer vision.
3. Apply feature detection and matching techniques to analyze image data.
4. Understand pattern recognition and machine learning concepts and their application in image analysis.
5. Describe and implement segmentation techniques to partition images into meaningful regions.
6. Understand and apply object detection and recognition techniques for identifying and analyzing objects in images or sequences.

CO–PO Mapping

CO	PO1	PO2	PO3	PO12	PSO1
1	3	2	1	1	1
2	3	2	2	1	1
3	3	3	3	1	3
4	3	3	3	2	3
5	2	3	3	2	3
6	2	3	3	2	3

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I **12 Hours**

Introduction to Computer Vision and Image Processing

Computer Vision: Introduction to computer vision - applications of computer vision - Image formation, Geometric primitives, 2D transformations, and photometric image formation - Sampling and aliasing in images: Image processing Techniques: Point operators: Thresholding, Image Enhancement - Linear filters - Non-Linear filters - Multi-resolution representations : Gaussian pyramid, Laplacian Pyramid - wavelets: 1D Haar wavelet transform , 2D Haar wavelet transform.

Neighborhood operators (morphological operations) and content-based image retrieval.

Unit II **12 Hours**

Feature Detection and Matching

Feature detection : Image features - Edge Detection - Overview of edge detection methods, Prewitt, Sobel, and Canny Edge detectors - Line detection: Hough Transform for line detection - Feature descriptors: Histogram of Oriented Gradients (HOG), Local Binary Pattern (LBP), SIFT and SURF- Feature Matching techniques - Performance Analysis.

Harris corner detection, and Piecewise Linear Approximation

Unit III **12 Hours**

Pattern Recognition and Machine Learning

Introduction to pattern recognition: Pattern, Features, RST invariant features - Pattern recognition systems - Different approaches to pattern recognition - Statistical Pattern Recognition: Bayesian decision theory and discriminant functions. Supervised learning: Parametric methods (linear regression), Nonparametric methods (k-nearest neighbors, decision trees) - Unsupervised learning: Clustering algorithms (k-means, mean shift clustering). Neural pattern recognition: Structure and types of neural networks, Learning algorithms for neural networks.

Dimensionality reduction methods- PCA (Principal Component Analysis) and LDA (Linear Discriminant Analysis)

Unit IV **12 Hours**

Image Analysis and Object Recognition

Segmentation techniques: Region-based segmentation: Split and merge, region growing, Contour-based segmentation: Active contours, Clustering-based segmentation: K-means, mean shift. Object detection: Techniques for object detection - Face recognition: Methods and challenges in face recognition - Instance recognition Techniques – Image Classification – Semantic segmentation - Motion estimation: Optical flow, activity recognition, motion estimation.

Medical image segmentation and Deep learning object detection

Total: 48 Hours

Textbook (s)

1. Richard Szeliski, Computer Vision: Algorithms and Applications , Springer, 2022, ISBN:978-1848829343
2. Rober.J. Shelkoff, John Wiley &Sons,Pattern Recognition- Statistical, Structural and Neural Approaches,Wiley, 2007, ISBN: 978-8126513703.

Reference (s)

1. R.O.Duda, P.E.Hart and D.G.Stork, Pattern Classification, Wiley India, 2006, ISBN: 978-8126511167.
2. Simon J. D. Prince, Computer Vision: Models, Learning, and Inference, Cambridge University Press, 2012,78-1107011793
3. Milan Sonka, Vaclav Hlavac, Roger Boyle, Image Processing, Analysis, and Machine Vision, Brooks/Cole, 2007, ISBN:978-0495082521.

Internal Assessment Pattern

Cognitive Level	Int. Test 1 (%)	Int. Test 2 (%)	Open Book Exam
Remember	30	30	-
Understand	50	50	-
Apply	20	20	50
Analyze	--	--	50
Evaluate	--	--	-
Create	--	--	-
Total (%)	**100**	**100**	**100**

Remember

1. What is computer vision and what is its role in image processing?
2. Define Hough transform.
3. List the roles of linear filtering in image processing.
4. What is Instance recognition?

Understand

1. How does sampling affect the quality of an image?
2. Explain the working principles of K-nearest neighbors (KNN) and decision trees in pattern recognition.
3. Explain the concept of wavelets and their use in multi-resolution representations.
4. Describe the process of feature detection in computer vision.
5. Describe the process of optical flow estimation and its applications in motion tracking.

Apply

1. Discuss the application of mean shift and mode finding in image segmentation.
2. Explain the concept of multi-resolution representations in image analysis. How are wavelets used in this context?

Analyze

1. Analyze the impact of aliasing in images and discuss methods to mitigate its effects.
2. Compare and contrast parametric and nonparametric methods in supervised learning for pattern recognition.
3. Analyze the challenges and potential solutions for object tracking in crowded or occluded scenes.

23MLC12 Machine Learning for Business Intelligence

3 0 2 4

Course Outcomes

1. Understand the basic concepts of business analytics.
2. Identify the application of business analytics and use tools to analyze business data.
3. Apply data mining techniques to find useful patterns in large data and help make better business decisions.
4. Become familiar with various metrics, measures used in business analytics.
5. Illustrate various descriptive, predictive, and prescriptive methods and techniques.
6. Model the business data using various business analytical methods and techniques.

CO–PO Mapping

CO	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2
1	3	1	2	1	1	2	2
2	3	1	3	1	3	3	3
3	3	3	1	2	3	2	2
4	1	3	1	3	3	2	2
5	1	1	3	3	3	3	3
6	1	2	2	1	3	3	3

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

UNIT I **11+ 9 Hours**

Introduction to Business Analytics: Introduction to Business Analytics, need and science of data driven (DD) decision making, Descriptive, predictive, prescriptive analytics and techniques, big data analytics, Web and Social media analytics, Machine Learning algorithms, framework for decision making, challenges in DD decision making and future.

Recommender Systems, Automated Decision-Making, Reinforcement Learning for Decision-Making

Practical Components

1. Descriptive, Predictive analysis on numeric data
2. Implementation of central tendency models on an appropriate dataset
3. Sample case study on Types of Social Media Analytics Report
4. Sample case study on Decision making

UNIT II **12+ 9 Hours**

Data mining, Text mining, Web mining, Spatial mining, Process mining, Data warehouse and datamarts. Data mining process KDD, CRISP-DM, SEMMA and Domain-Specific, Classification and Prediction performance measures -RSME, MAD, MAP, MAPE, Confusion matrix, Receiver Operating Characteristic curve & AUC; Validation Techniques - hold-out, k-fold cross-validation, LOOCV, random subsampling, and bootstrapping.

Nested Cross-Validation, Stratified Cross-Validation, Time Series Cross-Validation

Practical Components

1. Measure Prediction performance with confusion matrix on sample dataset
2. Implement ROC curve and AUC
3. Implementation of K-fold cross validation for sample dataset
4. Implementation of bootstrapping with suitable datasets

UNIT III **11+ 6 Hours**

Forecasting Techniques: Introduction, time-series data and components, forecasting accuracy, moving average method, single exponential smoothing, Holt's method, Holt-Winter model, Croston's forecasting

method, regression model for forecasting, Auto regression models, autoregressive moving process, ARIMA, Theil's coefficient.
Coefficient of Variation, Shannon Diversity Index, Simpson's Diversity Index, Gini Coefficient

Practical Components

1. Implementation of various smoothing techniques on sample datasets
2. Implement Moving Average forecasting technique
3. Implement Holt-Winter forecasting technique
4. Implementation of regression for forecasting
5. Measure the Theil's coefficient for suitable data

UNIT IV **11+ 6 Hours**

Six Sigma: Introduction, origin, 3-Sigma Vs Six-Sigma process, cost of poor quality, sigma score, industry applications, six sigma measures, DPMO, yield, sigma score, DMAIC methodology, Six Sigma toolbox
Statistical Process Control, Design of Experiments, Theory of Constraints (TOC

Practical Components

1. Illustrate difference between 3-sigma vs six-sigma process
2. Implementation of DMAIC method with suitable scenario
3. Case study on Enhancing Employee Management using Lean Six Sigma toolbox

Total 48+32 Hours

Textbooks:

1. Jaiwei Ham and Micheline Kamber, Data Mining concepts and techniques, Kauffmann Publishers 2006
2. Efraim Turban, Ramesh Sharda, Jay E. Aronson and David King, Business Intelligence, Prentice Hall, 2008.
3. W.H.Inmon, Building the Data Warehouse, fourth edition Wiley India pvt. Ltd. 2005
4. Marc J. Schniederjans, Dara G. Schniederjans, Christopher M. Starkey, "Business Analytics Principles, Concepts, and Applications with SAS", Associate Publishers,2015.

References:

1. S. Christian Albright, Wayne L. Winston, "Business Analytics - Data Analysis andDecision Making", 5th Edition, Cengage, 2015.
2. U Dinesh Kumar, "Data Analytics", Wiley Publications, 1st Edition, 2017.
3. Ralph Kimball and Richard Merz, The data warehouse toolkit, John Wiley, 3rd edition,2013.
4. Michel Berry and Gordon Linoff, Mastering Data mining, John Wiley and Sons Inc,2nd Edition, 2011
5. https://machinelearningmastery.com/time-series-forecasting-methods-in-python-cheat-sheet/
6. https://leansixsigmatoolbox.com/2023/06/08/ongoing-education-lean-six-sigma-employee-engagement/

Internal Assessment Pattern

Cognitive Level	Int. Test 1 (%)	Int. Test 2 (%)	Open Book Exam
Remember	30	30	-
Understand	50	50	-
Apply	20	20	50
Analyze	--	--	50
Evaluate	--	--	-
Create	--	--	-
Total (%)	**100**	**100**	**100**

Remember

1. What is business intelligence?
2. Define predictive analysis

Understand

1. Explain the k-fold cross validation with a suitable machine learning algorithm
2. Explain forecasting techniques for time-series data

Apply

1. Illustrate difference between 3-sigma vs six-sigma process
2. Implementation of DMAIC method with suitable scenario

23MLC13 Conversational AI (Elective V)

3 0 0 3

Course Outcomes

1. Understand the fundamentals of Natural Language Processing and basic vectorization approaches.
2. Understand the types of AI assistants, platforms and able to build their own Conversational AI.
3. Understand Dialog systems and Speech Recognition.
4. Understand the Dialog Management and Modeling.
5. Apply different rule based dialog management.
6. Make use of End-to-End Neural Dialogue Systems.

CO–PO Mapping

COs	PO1	PO2	PO4	PO12	PSO1
1	3	2	1	1	1
2	2	3	1	3	1
3	3	2	2	3	2
4	2	3	1	3	3
5	1	3	2	3	3
6	1	2	3	1	1

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I **12 Hours**

Introduction to NLP:

Definition, History, Building blocks of language, Approaches to NLP, NLP Pipeline, Deep learning in NLP and Transformers in NLP.

Basic Vectorization approaches- One-Hot Encoding, Bag of Words, Bag of N-Gram, TF-IDF; Neural language models, N-gram language model Sequence labeling for POS and Named Entities: POS tagging, Named Entities tagging.

Evaluation of Named Entity Recognition, Markov chains.

Unit II **12 Hours**

Introduction to Conversational AI: Introduction to AI assistants and their platforms: Types of AI assistants and their platforms, Primary use cases for AI assistant technology: self-service assistant, agent assist, and classification and routing.

Building your first conversational AI: Building a conversational AI for Fictitious Inc, What's the user's intent and Responding to the user.

The applications of Conversational AI and Objective of Chatbots.

Unit-III **12 Hours**

Dialog Management and Modelling

Introducing Dialogue Systems: Dialogue System, History, Present-day Dialogue System, Modelling conversation in dialogue systems.

Rule-Based Dialogue Systems: A typical dialogue systems architecture and Tools for developing dialogue systems and Evaluating Dialogue Systems

Challenges of the dialogue systems, Conversational Agents and Multimodal dialogue systems.

Unit IV **12 Hours**

End-to-End Neural Dialogue Systems

Neural Network Approaches to Dialogue Modelling, A Neural Conversational Model, Introduction to the Technology of Neural Dialogue and Open-Domain Neural Dialogue Systems.

Case Study: Chatbots in healthcare and mental health support, Voice-enabled devices and smart home applications.

Objective of Conversational AI with Rasa and Design Principles of the Chatbot.

Total: 48 Hours

Textbook (s)

1. Andrew R. Freed. Conversational AI: Chatbots that work, Manning; 1st edition (12 October 2021)
2. Yoav Goldberg. Neural Network Methods for Natural Language Processing, Morgan and Claypool Life Sciences, 30 April 2017
3. Michael McTear . Conversational AI: Dialogue Systems, Conversational Agents, and Chatbots (Synthesis Lectures on Human Language Technologies), Morgan & Claypool Publishers, 30 October 2020
4. Dan Jurafsky and James H. Martin. Speech and Language Processing (3rd ed. draft), Pearson, 22 Nov 2023.

Reference (s)

1. Xiaoquan Kong , Guan Wang . Conversational AI with Rasa by Packt., 1st edition, Oct 2021.
2. Stephan Bisser . Microsoft Conversational AI Platform for Developers End-to-End Chatbot Development from Planning to Deployment.1st edition, 17 Feb 2021.
3. Lee Boonstra . The Definitive Guide to Conversational AI with Dialog flow and Google Cloud build advanced enterprise chatbots, voice., 1st edition, 24 June 2021.
4. https://hao-cheng.github.io/ee596_spr2019/

Internal Assessment Pattern

Cognitive Level	Int. Test 1(%)	Int. Test 2 (%)	Open Book Exam (%)
Remember	20	20	--
Understand	50	40	--
Apply	30	40	50
Analyze	--	--	50
Evaluate	--	--	--
Create	--	--	--
Total (%)	**100**	**100**	**100**

SAMPLE QUESTION (S)

Remember

1. What is Natural Language Processing.
2. Define purpose of different building blocks of NLP
3. List any two challenges in NLP.

Understand

1. Explain the Spoken Language Understanding (SLU)
2. Explain the role of various Intent Classification.

Apply

1. Demonstrate the Frame-based SLU.

2. Make Use of Speech Technology in Dialogue System.

Analyze

1. Distinguish of modeling conversation in Dialogue systems.
2. Compare the Rule-Based Dialogue Systems and Evaluating Dialogue Systems.

Evaluate

1. Evaluate the End-to-End Neural Dialogue Systems.
2. Design an automated speech recognition system

Open book Question:

1. Design a model to analyse a given sentence is negative, positive or neutral for a given dataset and build end to end system.
2. Design and develop a model to Chatbots in healthcare and mental health support.

Career Path 2 - Full Stack Developer

23CSC21 Backend Programming Languages

3 0 0 3

Course Outcomes:

1. Demonstrate a comprehensive understanding of web fundamentals.
2. Apply RESTful API for modern web applications.
3. Develop and deploy basic web applications using Node.js and Express.js.
4. Demonstrate the ability to understand C# concepts and build web applications using ASP.NET MVC.
5. Create efficient and dynamic RESTful APIs with CRUD functionality.
6. Design and develop web applications using Python.

CO-PO Mapping:

COs	PO1	PO2	PO3	PO5	PSO1	PSO2
1	3	2	1	1	2	1
2	3	3	3	2	3	3
3	3	3	3	3	3	3
4	3	3	3	3	3	3
5	3	3	3	3	3	3
6	3	3	3	3	3	3

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit-I **12 Hours**

Overview of the web: Basic concepts of web, Anatomy of HTTP transaction, Request and response structures, HTTP status codes, REST APIs, Browser storage - Local storage & Session storage

RESTful programming: Introduction to Restful API, REST Principles, Resource Identification, Representation of Resources: JSON, XML and Plain Text/HTML, HTTP Methods: GET, POST, PUT and DELETE, Stateless Communication, Versioning, Error Handling, Pagination.

Basic HTML Tags, Basic CSS concepts, Authentication and Authorization in RESTful

Unit-II **12 Hours**

Programming with JavaScript

Introduction to Node.js, Features of Node.js, npm, understanding package. json, Modules: fs, http, path and os, Express.js: Introduction to Express.js, Setting up an Express server, Handling GET and POST requests, Middleware in Express, Template Engine: Embedded JavaScript Templates, rendering dynamic HTML, building a REST API: Creating routes for CRUD operations, understanding authentication vs. authorization, Implementing basic JWT.

Basics of JavaScript and ES6 JavaScript, Node Packages, async, await

Unit-III **14Hours**

Programming with C#

Introduction to .NET: Introduction, .NET Framework vs .NET Core vs .NET, components of .NET, CLR, FCL, Managed code and Unmanaged code, C# Concepts: Introduction, Features, Datatypes, Functions, Arrays, Classes and Objects, Properties, Inheritance, Polymorphism, Interfaces, Namespaces, Collections, Delegates, Events, Exception Handling, Introduction to ASP.NET: MVC, Routing, Razor pages vs MVC, Create RESTful endpoints, Model Binding

C# variables, LINQ, async/await, Multithreading, ORM, ASP.NET life cycle, web forms, ADO.NET introduction

Unit-IV **10 Hours**

Programming with Python

Introduction to Python, Classes and objects, Constructors, Encapsulation, Inheritance, Polymorphism, Decorators and Closures, Abstract Base Classes, Generators and Iterators, Metaclasses, Functional Programming: map(), filter(), reduce(), Python's HTTP package, building a REST API: creating routes for CRUD operations.

Basic concepts of python, Python packages

Total: 48 Hours

Textbook (s)

1. Restful Web APIs, Leonard Richardson, Mike Amundsen, Sam Ruby, O'Reilly Media.
2. Web Development with Node and Express, Ethan Brown, O'Reilly Media.
3. C# Programming, Wikibooks Contributors, Platypus Global Media.
4. Python Programming: A Modern Approach, Vamsi Kurama, Pearson.

Reference (s)

1. Pro ASP.NET Core 6, Adam Freeman, 9th edition, Apress.
2. Automate the Boring Stuff with Python, Al Sweigart, No Starch Press, 2nd Edition.

Internal Assessment Pattern

Cognitive Level	Int. Test 1 (%)	Int. Test 2 (%)	Open Book Exam (%)
Remember	30	30	--
Understand	40	40	--
Apply	30	30	50
Analyze	--	--	50
Evaluate	--	--	--
Create	--	--	--
Total (%)	100	100	100

SAMPLE QUESTION (S)

Remember:

1. What are the advantages of Restful API?
2. What is the difference between Node.js and Express.js?
3. Define C# Function?
4. Write a short note about Abstract Base Classes.
5. Write a short note on Browser Storage.

Understand:

1. Explain about request and response structure.
2. What is Restful API? Explain about various Rest principles.
3. Describe the procedure to use middleware in Express.js.
4. Explain in detail about CLR.
5. Discuss about map(), filter(), reduce() in Python.

Apply:

1. Implement a basic Node.js program that reads and writes a file using the fs module.
2. Create a simple HTTP server using the http module that responds with "Welcome to Node.js!".
3. Write a simple C# program to handle an Exception.
4. Write a simple C# code that calculates and displays the sum of two numbers entered by the user.
5. Implement a simple HTTP server in Python that handles GET and POST requests.

Analyze:

1. How does the client-server architecture impact the way web applications handle user requests?
2. Compare the http module in Node.js with Express.js in terms of request handling and ease of use.
3. Compare how Razor pages are different from MVC.
4. Analyze the difference between .NET Framework and .NET Core?
5. Analyze the difference between synchronous and asynchronous HTTP requests in Python.

23CSC22 Web Application Frameworks

3 2 0 4

Course Outcomes:

1. Describe the fundamentals of web framework.
2. Design and implement Angular-based applications.
3. Design and implement React-based applications.
4. Build robust and secure web applications using Spring Boot.
5. Create efficient and dynamic REST APIs with CRUD functionality
6. Design and develop dynamic web applications using MVT architecture

CO-PO Mapping:

COs	PO1	PO2	PO3	PO5	PSO1	PSO2
1	3	3	1	1	2	1
2	3	3	3	3	2	3
3	3	3	3	3	2	3
4	3	3	3	3	3	2
5	2	3	2	3	2	3
6	3	3	3	2	2	3

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit-I **13+10 Hours**

Fundamentals of Web Framework

Introduction to frameworks-Framework applications, General-purpose website frameworks-Server-side, Client-side features, Types of framework architecture.

Angular.js

Introduction – Angular MVC, Ajax, Data binding, Angular concepts - Directives, Components & Modules, Templates & Data Binding, Routing & Navigation, Modules, Expressions, HTTP and API integration.

JavaScript ES6 concepts, MVC, Three-tier organisation, Ajax, Scopes, Controllers

Practical Components

1. Implementation of angular directives
2. Implementation of angular components
3. Implementation of custom/user-defined directive
4. Create a TODO Application using Angular JS (Single page application)
5. Implement routing and navigation in Angular JS

Unit-II **13+12 Hours**

React.js

Introduction to React: What is Full-Stack Web Development?, Introduction to React, React App Overview, Core Components: Introduction to JSX, React Components, State and Props, Lifecycle Methods: class and function components, use of fetch and axios, Hooks, React Router and Single Page Applications: Presentational and Container Components, React Virtual DOM, React Router: Parameters, React Forms, Flow Architecture and Introduction to Redux: Controlled Forms, Uncontrolled Components, The Flux Architecture, Introduction to Redux, React Redux Forms

JavaScript ES6 concepts, React Events, State Management, Introduction to React Native

Practical Components

1. React Components
2. React Router and Single page applications
3. Controlled Form Validation, Uncontrolled Forms
4. Create a TODO Application using React JS

5. Creating a movie list Application in React JS
6. Creating a Progressive web application with Reusable React Components

Unit-III **10+4 Hours**

Spring Boot

Introduction to Spring Boot, Application Structure, Framework Theory, Annotations, Bean Management, Dependency Injection, Spring Data: JDBC, ORM, JPA, Building Rest APIs, Starter Dependencies, database Integration, Creating Model Class.

Spring MVC, Error Handling, Spring Boot Security, Asynchronous Programming

Practical Components

1. Configuration in Spring Boot and Create a Simple Spring Boot Application
2. Apply Validation to given input data with REST endpoint and test with valid or invalid data using any API tester.

Unit-IV

Django **12+6 Hours**

Introduction, virtual environment, install Django, Create Django App, App Life Cycle, Project Structure, URL mapping, Django MVT - model and CRUD operations on model, Django view and Django template and configuration, Static File Handling, Django Admin- create user, Response and Redirect, Django Forms

Basic concepts of python, Python packages, Django Cookies and sessions handling

Practical Components

1. Creating models for database queries
2. Writing Django template for rendering data
3. Implementation of REST API using Django

Total: 48+32 Hours

Textbook (s)

1. Angular: Up and Running, ShyamSeshadri, 1st Edition, O'Reilly, 2018
2. Struts the Complete Reference, James Holmes, 2nd Edition, Mc. Graw Hill Professional, 2006.
3. Professional Java Development with the Spring Framework, Alef Arendsen, Thomas Risberg, Rod Johnson, Wiley publications, 2007.
4. Programming with Django, Wiley Publishing

Reference (s)

1. Angular 6 for Enterprise-Ready Web Applications, DoguhanUluca, 1st edition, 2018
2. The Definitive Guide to Django, Adrian Holovaty, Jacob Kaplan-Moss, Apress, 2009.
3. Struts 2 In Action, Donald Brown, Chad Michael Davis, Scott Stanlick, Dreamtech press, 2008.

Internal Assessment Pattern

Cognitive Level	Int. Test 1 (%)	Int. Test 2 (%)
Remember	50	40
Understand	30	40
Apply	20	20
Analyze	--	--
Evaluate	--	--
Create	--	--
Total (%)	100	100

SAMPLE QUESTION (S)

Remember:

1. What is a web framework?
2. Define Full-Stack Web Development.
3. What is Spring Boot?
4. What is Django?
5. What command is used to create a Django project?

Understand:

1. Explain the concept of Model-View-Controller (MVC) in a framework.
2. Explain the difference between state and props in React.
3. Describe the role of JSX in React.
4. Explain the purpose of Spring Boot Annotations with examples.
5. What is the difference between Django MVT and traditional MVC architecture?

Apply:

1. Create a simple HTTP request using a client-side framework.
2. Write a simple React component using JSX.
3. Write a simple Spring Boot application that prints "Hello, World!".
4. Implement a simple CRUD application using Spring Data JPA.
5. Develop a Django model for a product catalog and apply CRUD operations.

Analyze:

1. Compare and contrast server-side and client-side frameworks.
2. Compare useEffect and lifecycle methods in class components.
3. How does Spring Boot handle Bean Management and Dependency Injection?
4. How does Django's URL dispatcher work?
5. Compare Django ORM vs. raw SQL queries.

23CSC23 Web Application Databases

3 0 0 3

Course Outcomes:

1. Describe the architecture and programming concepts of web databases.
2. Implement state management and database connectivity mechanisms
3. Apply structured database management techniques.
4. Apply the principles of unstructured databases and NoSQL technologies.
5. Develop proficiency in database connections and communication protocols.
6. Analyze the security mechanisms for web services

CO-PO Mapping:

COs	PO1	PO3	PO5	PSO1
1	3	1	2	3
2	2	3	3	3
3	3	3	1	3
4	3	2	2	2
5	2	3	3	3
6	2	2	1	1

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit-I **11 Hours**

Overview of Web databases

Web application Overview: Basic concepts of web, Web database architectures, Web database gateways, Web-database programming: Client-side and server-side web database programming, State management: Session management, cookies, request and response header.

Database gateways, CGI, Browser extensions

Unit-II **13 Hours**

Structured Database – PostgreSQL

Structured Databases: Structured data, Data models, Relational database management systems, Overview of PostgreSQL – Introduction, Environment Setup, PostgreSQL Database: Create, Select and Drop, CRUD operations, Advanced PostgreSQL: Constraints, Views, Transactions, Auto Increment, Privileges, Functions, Python interface, Concept of migrate database, Challenges of structured data- MySQL, SQLite

PostgreSQL Interfaces, Triggers, MySQL Connection, SQLite Connection

Unit-III **13 Hours**

Unstructured Database – MongoDB

Unstructured Databases: Unstructured data and its usages, NoSQL databases, Overview of MongoDB – Introduction, No SQL Database, Advantage over RDBMS, MongoDB Data Types, Install MongoDB, Data Modeling, MongoDB Operators, Database Commands, MongoDB Architecture - Create Database, Drop Database, Collection - Create Collection, Drop Collection, CRUD: Documents, MongoDB Shell, MongoDB Cloud, Challenges of unstructured data- Cassandra, GraphQL

Architecture of Cassandra, Cassandra Data model, GraphQL Architecture and Application Components

Unit-IV

Database connections, Security & Communication in web services **11 Hours**

Connecting to database: Native database APIs, Database-independent APIs, Template-driven database access packages, Third-party class libraries.

Web requests: Structure of web requests, JavaScript Object Notation (JSON), Request methods – GET, POST, PUT, DELETE, response status codes

Security: User authentication, User authorisation, Proxy servers, Digital signatures, Digital certificates, SSL and HTTPS.
JDBC, ODBC, template parsing, JSON, User authentication and authorisation

Total: 48 Hours

Textbook (s)

1. Web Database Applications with PHP and MySQL, Hugh E. Williams, David Lane, 2nd Edition, O'Reilly, 2004.
2. Mastering PostgreSQL 13, Hans-Jürgen Schönig, 4th Edition, Kindle Edition.
3. NoSQL Distilled, PramodSadalage, Martin Fowler, 1st Edition, 2012

Reference (s)

1. Getting Started with NoSQL, GauravVaish, Paperback, Packt, 2013.
2. Fundamentals of Database Systems, ElmasriNavathe Pearson Education.
3. An Introduction to Database systems, C.J. Date, A.Kannan, S.SwamiNadhan, Pearson, 8th Edition.
4. Web Application Security, Andrew Hoffman, O'Reilly, 2020.

Internal Assessment Pattern

Cognitive Level	Int. Test 1 (%)	Int. Test 2 (%)	Open Book Exam (%)
Remember	20	20	--
Understand	40	40	--
Apply	20	20	40
Analyze	20	20	40
Evaluate	--	--	20
Create	--	--	--
Total (%)	100	100	100

Sample Questions

Remember:

1. What are the basic components of a web application?
2. What is a web database architecture?
3. What is a relational database management system (RDBMS)?
4. List different types of NoSQL databases.
5. Define database-independent APIs.

Understand:

1. Explain the process of session management in a web application.
2. Explain how PostgreSQL handles transactions.
3. Explain the architecture of MongoDB and how it differs from RDBMS.
4. How does JSON differ from XML in web communication?
5. Explain the difference between DELETE and TRUNCATE in PostgreSQL.

Apply:

1. Create a client-side script that validates form inputs.
2. Perform a database migration using PostgreSQL.
3. Implement CRUD operations in MongoDB using Python or Node.js.
4. Implement JWT-based authentication in a web application.
5. Write a query to filter data using comparison operators.

Analyze:

1. Compare two-tier and three-tier web database architectures.
2. Compare PostgreSQL, MySQL, and SQLite in terms of performance and scalability.
3. How does MongoDB handle ACID properties compared to traditional databases?
4. Compare session-based authentication vs. token-based authentication.
5. Analyze the impact of response status codes on API performance.

Career Path-3: Cloud Computing

23MLC31 Fundamentals of Cloud Computing

3 0 0 3

Course Outcomes

1. Interpret the architecture and infrastructure models of cloud computing, strengths, and limitations of cloud computing.
2. Understand the techniques of virtualization, including virtual machines and virtualized datacenters.
3. Apply design principles and architectural patterns to develop scalable, resilient, and cloud-native applications.
4. Compare and contrast various cloud service provider implementations and deployment models based on performance, cost, and features.
5. Understand the strategies for cloud scaling, load balancing, fault tolerance, and disaster recovery in cloud environments.
6. Identify and assess security, privacy, and risk management issues associated with cloud computing platforms and services.

COs–POs Mapping

COs	PO_1	PO_2	PO_3	PO_4	PO_5	PSO_1	PSO_2
1	3	2	2	1	2	2	1
2	3	3	2	2	3	2	2
3	2	2	3	2	3	3	2
4	3	2	3	2	2	2	2
5	3	3	3	2	2	2	2
6	2	2	2	2	2	2	2

3–Stronglylinked|2–Moderatelylinked|1–Weakly linked

Unit I **12 Hours**

Cloud Computing

Cloud computing: Introduction, SOA, Cloud computing architectures, Value of cloud computing, Cloud Infrastructure models including SaaS, PaaS, IaaS, Storage-as-a-Service, Database-as-a-Service, Information-as-a-Service,Process-as-a-Service,Application-as-a-Service, Integration-as-a-Service, and Security-as-a-Service. Public cloud, private cloud and hybrid cloud, Cloud Services.

Before moving into the cloud: Know Your Software Licenses, The Shift to a Cloud Cost Model. History of Cloud Computing, Advantages of Cloud Computing, and Disadvantages of Cloud Computing.

Companies in the Cloud Today, Amazon Web Services, Google services, IBM Cloud, Windows Azure, Tata Cloud, Salesforce.com

Unit II **13 Hours**

Virtualization & Design

Virtualization,Virtualmachine, Implementation Levels of Virtualization,Virtualization Structures/Tools and Mechanisms, Virtualization of CPU, Memory, and I/O Devices, Virtual Clusters and Resource Management, Datacentre, Virtualization for Data-Centre Automation. Service Levels for Cloud Applications Ready for the cloud: Web Application Design, Machine Image Design, Privacy Design, Database Management.

*Various hypervisors like VMware, KVM,*oracleVM,

Unit III **12 Hours**

Cloud centres, Scaling a Cloud Infrastructure and Disaster Recovery:

Cloud centres in detail: Comparing approaches, Xen, Eucalyptus, CloudStack, and OpenStack. Cloud Scaling:Capacity Planning, Cloud Scale, Types of cloud scaling.

Disaster Recovery:Disasters in cloud,Disaster Recovery Planning, Cloud Disaster Management.

Requirements for modern data centres-high availability and Service Orientated-Infrastructures(SOI).Modern data centre use case studies.

Unit IV **11 Hours**

Cloud Computing Software Security Fundamentals

Data Security, Network Security, Host Security, Compromise Response.

Cloud information Security Objectives, Cloud Security Services, Relevant Cloud Security Design Principles, Secure Cloud Software Requirements, Approaches to Cloud Software Requirements Engineering, Cloud Security Policy Implementation.

Cloud Computing Risk Issues: The CIA Triad, Privacy and Compliance Risks, Threats to Infrastructure Data and Access Control, Cloud Access Control Issues, Cloud Service Provider Risks.

MicrosoftAzure, IBMBluemix

Total: 48 Hours

Textbook(s)

1. GeorgeReese,Cloud ApplicationArchitectures,1st Edition O'ReillyMedia,2009
2. RonaldL.KrutzandRussell DeanVines,CloudSecurity,1st Edition,WileyPublishing,2010

Reference(s)

1. MichaelMiller,CloudComputing-WebBasedApplicationsthatchangethewayyouworkandcollaborateonline, 1st Edition,PearsonEducation,Publishing, 2011
2. KaiHwang,GeoffreyCFoxandJackJ.Dongarra,Distributed&CloudComputingfromParallelProcessingto theInternet ofThings,1st Edition, MK Publishing,2010
3. DavidSLinthicum,CloudComputingandSOAConvergenceinYourEnterprise:AStep-by-StepGuide,1st Edition, Addison-Wesley, 2009

SAMPLEQUESTION(S)

Internal Assessment Pattern

Cognitive Level	Int.Test1 (%)	Int.Test2 (%)	AssignmentTest3(%)
Remember	40	45	20
Understand	40	45	60
Apply	20	10	
Analyze	--	--	20
Evaluate	--	--	--
Create	--	--	--
Total(%)	100	100	100

Remember

1. Define Cloud Computing.
2. List types of virtualization.
3. Define proactive scaling.
4. What is CIAT RAID?

Understand

1. Explain different cloud Infrastructure models.
2. Explain different levels of virtualization.
3. Explain about Recovery Point Objective.
4. With neat diagram explain the functioning of Xen Architecture.
5. List and explain various cloud service providers risks.

Analyze

1. Compare cloud center and service infrastructure.
2. Analyze different cloud services provided by Amazon

23MLC32 Cloud Services using AWS

3 0 2 4

Course Outcomes

1. Describe various services offered by Amazon Web Services (AWS) and their applications in cloud computing.
2. Understand the concepts and implementation of Identity and Access Management (IAM) for secure AWS resource access.
3. Demonstrate the usage of AWS Directory Services and AWS Artifact for identity integration and compliance.
4. Deploy and manage scalable computing resources using Amazon Elastic Compute Cloud (EC2).
5. Illustrate the features and functionalities of Amazon Elastic File System (EFS) for scalable storage.
6. Compare AWS storage solutions such as S3, EBS, and Glacier based on use cases and performance needs.

COs–POs Mapping

COs	PO_1	PO_2	PO_3	PO_4	PO_5	PSO_1
1	2	2	2	2	2	2
2	2	2	2	2	2	2
3	2	2	3	2	2	2
4	3	3	3	3	3	2
5	2	2	2	2	2	2
6	2	2	2	2	2	2

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit-I **12+8 Hours**

Introduction to AWS:

Classic Data Center , Virtualization, Service Comparison: AWS, Azure, and GCP, Amazon Web Services (AWS) and its Benefits, AWS Global Infrastructure, Signup an AWS Free Tier Account.

Different Amazon Web Services, Ways to access AWS: CLI, Console, and SDKs, Explore Management Console and Configure AWS CLI

AWS CloudShell

Practical Components

1. Signup an AWS Free Tier Account
2. Explore Management Console and configure AWS CLI
3. Access AWS: CLI, Console, and SDKs
4. Working with AWS services in AWS CloudShell

Unit-II **12+8 Hours**

Identity and Access Management in AWS:

Identity and Access Management (IAM), Managing Users with IAM, Managing Permissions with Groups, IAM Policy and its Elements, IAM Roles ,Password Policy, Best Practices for IAM

AWS Security Token Services, AWS Single Sign-on, AWS Resources Access Manager, Active directory, Microsoft Active directory, AWS Directory Services, AWS Artifact, AWS Audit Manager, Accessing Billing and Alerts.

AWS Certificate Manager

Practical Components

1. Creating New Users, Roles, and Policies on AWS Console
2. Creating an user group for a collection of IAM users
3. Login to AWS Console via MFA
4. AWS Artifact

Unit-III **12+8 Hours**

Amazon Elastic Compute Cloud (EC2):

Amazon Elastic Compute Cloud (EC2) and Its Benefits, Amazon Machine Image (AMI), Security Groups in AWS, Authentication through Key-pair, Hardware Tenancy – Shared vs. Dedicated, Networking Layer in EC2: VPC, Elastic Network Interface (ENI) and Its Attributes

Public IP vs. Elastic IP, Instance Store

Elastic Block Store (EBS), Its Features and Volume Types

Solid State Drive: General Purpose SSD and Provisioned IOPS

Hard Disk Drive: Throughput Optimized HDD and Cold HDD

Snapshots, Elastic File System (EFS) and Its Features, Amazon Batch, Windows File Server

Amazon FSx for Windows File Server

Practical Components

1. Creating an EC2 instance and Custom AMI
2. Host the Website Inside EC2 instance
3. Create an Elastic IP and attaching an EBS Volume Externally
4. Creating a Snapshot

Unit-IV **12+8 Hours**

AWS Storage Options:

Traditional Storage, Need to Move to Cloud Storage, Traditional vs. Cloud Storage Cost, Cloud Storage, Different Storage Options Available on AWS, Simple Storage Service (S3) and Its Components, Working of S3, Bucket Policy, Access Control List (ACL)

Cross-Region Replication (CRR) and Its Use Case, Amazon S3 Transfer Acceleration, Choice of Storage Classes on S3, Lifecycle Policy of S3 Bucket, AWS Backup.

CDN: CloudFront, Storage Gateway

Practical Components

1. Hosting a Static Website on Amazon S3
2. Replicating Data Across Regions
3. Transfer and Retrieve Data from Glacier
4. Accessing Website Using CloudFront

Total:48+32 Hours

Textbook(s):

1. Amazon Web Services (AWS) 2022 Everything You Need To Know About The Amazon Web Service From Start To End, 2022, Maxwell Harris, Maxwell Harris.
2. Implementing AWS: Design, Build, and Manage your Infrastructure, Yohan Wadia, Rowan Udell, Lucas Chan, Udita Gupta, packt publishers.

Internal Assessment Pattern

Cognitive Level	Internal Test 1 %	Internal Test 2 %	Lab Examination %
Remember	25	25	---
Understand	50	25	---
Apply	25	50	40
Analyze	---	---	20
Evaluate	---	---	40
Create			
Total (%)	**100**	**100**	**100**

Sample Question(s)

Remember

1. List any two cloud services provided by Amazon AWS.
2. What is AWS Artifact?
3. Define EC2.
4. What is Access Control List?

Understand

1. What is Virtualization and explain about that.
2. Explain about Identity and Access Management (IAM).
3. Write and explain about AWS Directory Services.
4. Explain about Access Control List (ACL).

Apply

1. How do we sign up for an AWS Free Tier Account?
2. Illustrate the process of Move to Cloud Storage from normal storage.

How Cross-Region Replication (CRR) works?

23MLC33 Cloud Security Essentials (Elective V)

3 0 0 3

Course Outcomes

1. Understand the fundamental concepts and architecture of cloud computing from a security perspective.
2. Describe the characteristics, benefits, and challenges of various cloud service and deployment models in relation to security.
3. Identify and evaluate common threats, vulnerabilities, risks, and privacy concerns in cloud-based environments.
4. Understand security challenges, threats and risks involved in the cloud application.
5. Analyze the Security Requirements for the Architecture in cloud services.
6. Understand Data Security and best practices involved in securing the cloud.

CO–PO Mapping

COs	PO_1	PO_2	PO_3	PO_4	PO_5	PSO_1
1	3	2	2	2	2	2
2	3	2	2	2	2	2
3	2	2	2	2	2	2
4	2	2	2	2	2	2
5	3	2	3	2	3	2
6	3	2	2	2	2	2

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I **12 Hours**

Cloud Computing Software Security Fundamentals- Cloud Information Security Objectives, Cloud Security Services, Relevant Cloud Security Design Principles,

Secure Cloud Software Requirements, Security Concerns, Risk Tolerance, Legal and Regulatory Issues

Unit II **12 Hours**

Cloud Computing Risk Issues and Security Challenges:

The CIA Triad,Privacy and Compliance Risks,Threats to Infrastructure, Data, and Access Control, Cloud Service Provider Risks, Disaster Recovery-Disaster Recovery Planning, Disasters in cloud, Cloud Disaster Management.

Security Challenges- Security Policy Implementation, Virtualization Security Management, VM Security Recommendations, VM-Specific Security Techniques.

Unit III **12 Hours**

Securing the cloud: Architecture and Data Security

Security Requirements for the Architecture, Security Patterns and Architectural Elements, Cloud Security Architecture, Planning Key Strategies for Secure Operation.

Unit IV **12 Hours**

Overview of Data Security in Cloud Computing, Data Encryption: Applications and Limits, Cloud Data Security: Sensitive Data Categorization, Cloud Data Storage, Cloud Lock-in, Key strategies to secure the cloud, Best practices for cloud computing, security monitoring

Total: 48 Hours

Textbook (s)

1. Russell Dean Vines and Ronald L. Krutz, Cloud Security: A Comprehensive Guide To Secure Cloud Computing, Wiley India Pvt Ltd, 2010
2. Vic Winkler (J.R.), Securing The Cloud: Cloud Computing Security Techniques and Tactics, Syngress/Elsevier ,2011

Reference (s)

1. Thomas Erl, Cloud Computing Design Patterns, Prentice Hall, 2015
2. Barrie Sosinsky, Cloud Computing Bible, Wiley India, 2011
3. Rajkumar Buyya, James Broberg, Andrzej M. Goscinski, Cloud Computing: Principles and Paradigms, Wiley, 2011

Internal Assessment Pattern

Cognitive Level	Int. Test 1 (%)	Int. Test 2 (%)	Open Book Exam
Remember	50	40	-
Understand	30	40	-
Apply	20	20	50
Analyze	--	--	50
Evaluate	--	--	-
Create	--	--	-
Total (%)	100	100	100

Remember

1. Define Cloud Computing.
2. List Cloud Service Models
3. List Cloud Information Security Objectives
4. What is cloud foundry?

Understand

1. Identify Cloud Computing Risk Issues and Security Challenges
2. Explain the Data Security
3. Explain all session management techniques
4. Explain different cloud service models

Apply

1. Demonstrate the process for war file deployment in the cloud.
2. How does Sensitive Data Categorization happens in Cloud?

Analyze

1. Justify your answer which technique is used for Session management.
2. Deploy Security Challenges