Anil Neerukonda Institute of Technology & Sciences (Autonomous)

(Permanent Affiliation by Andhra University & Approved by AICTE
Accredited by NBA (ECE, EEE, CSE, IT, Mech. Civil & Chemical) & NAAC)
Sangivalasa-531 162, Bheemunipatnam Mandal, Visakhapatnam District
Phone: 08933-225083/84/87 Fax: 226395
Website: www.anits.edu.in email: principal@anits.edu.in

2022-23

Academic Regulations (R20-CSD)

Curriculum & Syllabi (III Year I&II Semesters)

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING WITH DATA SCIENCE

ANITS-B Tech CSE (Data Science) CURRICULUM – REGULATIONS –R20

I Year Course structure – CSE(Data Science)

Semester - I

Course Code	Title of the course	Category	Periods						Sessional Marks	Semester end Exam marks	Total Marks	Credits
			L	T	P	E	O	Total				
CSD111	EngineeringMathematics – I	BS	3	0	0	1	6	10	40	60	100	3
CSD112	Communicative English	HS	3	0	0	0	3	6	40	60	100	3
CSD113	BASIC ELECTRONICS	ES	3	0	0	1	3	7	40	60	100	3
CSD114	PROBLEM SOLVING WITH C	ES	3	0	0	0	3	6	40	60	100	3
CSD115	Digital Logic Design	ES	3	0	0	1	3	7	40	60	100	3
CSD116	English Language Lab	HS	0	0	3	0	3	6	50	50	100	1.5
CSD117	Problem solving with C – lab.	ES	0	0	3	0	3	6	50	50	100	1.5
CSD118	Environmental Science (Mandatory non-credit course)	BS	3	0	0	0	1	4	50	-	50	-
Total			**18**	**0**	**6**	**3**	**25**	**52**	**350**	**400**	**750**	**18**

II Year Course structure – CSE(Data Science)

Semester - I

Course Code	Title of the course	Category	Periods						Sessional Marks	Semester end Exam marks	Total Marks	Credits
			L	T	P	E	O	Total				
CSD121	Engineering Mathematics – II	BS	3	0	0	1	6	10	40	60	100	3
CSD122	Engineering Physics	BS	3	0	0	1	4	8	40	60	100	3
CSD123	Engineering Chemistry	BS	3	0	0	1	4	8	40	60	100	3
CSD124	ELEMENTS OF ELECTRICAL ENGINEERING	ES	3	0	0	1	4	8	40	60	100	3
CSD125	Engineering Drawing	ES	2	0	3	1	3	9	40	60	100	3.5
CSD126	Engineering Physics Lab.	BS	0	0	3	0	1	4	50	50	100	1.5
CSD127	Engineering Chemistry Lab.	BS	0	0	3	0	1	4	50	50	100	1.5
CSD128	Engineering Workshop	ES	0	0	3	0	1	4	50	50	100	1.5
CSD129	Human Values and Professional Ethics(Mandatory non-credit course)	HS	3	0	0	0	1	4	50	-	50	-
Total			**17**	**0**	**12**	**5**	**25**	**59**	**400**	**450**	**850**	**20**

II Year Course structure – CSE(Data Science)

Semester - I

CODE	SUBJECT NAME	Category	Periods						Sessional Marks	Semester end Exam marks	Total Marks	Credits
			L	T	P	E	O	Total				
CSD 211	DATA STRUCTURES&ALGORITHMS	PC	2	1	0	1	4	8	40	60	100	3
CSD 212	COMPUTER ORGANIZATION AND MICROPROCESSORS	PC	3	0	0	1	4	8	40	60	100	3
CSD 213	JAVA PROGRAMMING	PC	3	0	0	1	4	8	40	60	100	3
CSD 214	DATA COMMUNICATION AN D COMPUTER NETWORKS	PC	3	0	0	1	4	8	40	60	100	3
CSD 215	DISCRETE MATHEMATICAL STRUCTURES	BS	3	0	0	1	4	8	40	60	100	3
CSD 216	DESIGN THINKING & PRODUCT INNOVATION	ES	2	0	2	1	3	8	40	60	100	3
CSD 217	JAVA PROGRAMMING LAB	PC	0	0	3	0	2	5	50	50	100	1.5
CSD 218	DATA STRUCTURES LAB USING C	PC	0	0	3	0	2	5	50	50	100	1.5
CSD 219	Constitution of Indian & - Intellectual Property Rights	HS	2	0	0	0	1	3	50	-	50	-
Total			**18**	**1**	**8**	**6**	**28**	**61**	**390**	**460**	**850**	**21**

Semester - II

II Year Course structure – CSE(Data Science)

CODE	SUBJECT NAME	Category	Periods						Sessional Marks	Semester end Exam marks	Total Marks	Credits
			L	T	P	E	O	Total				
CSD 221	PROBABILITY , STATISTICS AND QUEUING THEORY	BS	3	0	0	1	6	10	40	60	100	3
CSD 222	BASICS OF DATA SCIENCE	PC	2	1	0	2	4	9	40	60	100	3
CSD 223	OPERATING SYSTEMS	PC	3	0	0	1	4	8	40	60	100	3
CSD 224	PYTHON PROGRAMMING	PC	3	0	0	1	4	8	40	60	100	3
CSD 225	THEORY OF COMPUTATION AND COMPILERS	PC	2	1	0	1	4	8	40	60	100	3
CSD 226	DESIGN ANALYSIS OF ALGORITHMS	PC	2	1	0	1	4	8	40	60	100	3
CSD 227	***PYTHON PROGRAMMING LAB***	PC	0	0	***3***	***0***	***1***	4	***50***	***50***	100	1.5
CSD 228	***CO & MICRO PROCESSOR INTERFACING LAB***	***PC***	0	0	***3***	***0***	***1***	4	***50***	***50***	***100***	1.5
CSD 229	***OPERATING SYSTEM LAB***	***PC***	0	0	***3***	***0***	***1***	4	***50***	***50***	***100***	1.5
Total			**15**	**3**	**9**	**7**	**29**	**63**	**390**	**510**	**900**	**22.5**

III Year Course structure – CSE(Data Science)

Semester - I

CODE	SUBJECT NAME	Category	Periods						Sessional Marks	Semester end Exam marks	Total Marks	Credits
			L	T	P	E	O	Total				
CSD 311	OPEN ELECTIVE -I* (Block Chain as EC)	OE	3	0	0	1	2	6	40	60	100	3
CSD 312	PROFESSIONAL ELECTIVE -I	PE	3	0	0	1	2	6	40	60	100	3
CSD 313	DATA ANALYSIS & VISUALIZATION	PC	3	0	0	1	2	6	40	60	100	3
CSD 314	COMPETITIVE PROGRAMMING	SOC	2	1	0	1	5	9	40	60	100	3
CSD 315	DATA BASE MANAGEMENT SYSTEMS	PC	3	0	0	1	4	8	40	60	100	3
CSD 316	***DATA SCIENCE WITH PYTHON LAB***	***PC***	0	0	***3***	***0***	1	4	***50***	***50***	100	1.5
CSD 317	***DATA BASE MANAGEMENT SYSTEMS LAB***	***PC***	0	0	***3***	***0***	1	4	***50***	***50***	100	1.5
CSD 318	***COMPETITIVE PROGRAMMING LAB***	***SOC***	0	0	3	0	1	4	***50***	***50***	100	1.5
CSD 319	**QA-I &SOFT SKILLS**	HS	0	0	3	0	1	4	***100***	***0***	100	1.5
CSD 31A	***SUMMER INTERNSHIP-INDUSTRY-1***	***PR***	0	0	***0***	***0***	***1***	1	***100***	***0***	***100***	2
Total			**14**	**1**	**9**	**5**	**18**	**47**	**300**	**400**	**700**	**23**

Semester - II

III Year Course structure – CSE(Data Science)

CODE	SUBJECT NAME	Category	Periods						Sessionals Marks	Semester end Exam marks	Total Marks	Credits
			L	T	P	E	O	Total				
CSD 321	OPEN ELECTIVE -II*	OE	3	0	0	1	2	6	40	60	100	3
CSD 322	PROFESSIONAL ELECTIVE -II	PE	3	0	0	1	2	6	40	60	100	3
CSD 323	PROFESSIONAL ELECTIVE -III	PE	3	0	0	1	4	8	40	60	100	3
CSD 324	OBJECT ORIENTED SOFTWARE ENGINEERING	PC	3	0	0	1	4	8	40	60	100	3
CSD 325	WEB TECHNOLOGIES	PC	2	1	0	1	4	8	40	60	100	3
CSD 326	MACHINE LEARNING	PC	3	0	0	1	4	8	40	60	100	3
CSD 327	***WEB TECHNOLOGIES LAB***	***PC***	0	0	***3***	***0***	***1***	4	***50***	***50***	100	1.5
CSD 328	***MACHINE LEARNING LAB***	***PC***	0	0	***3***	***0***	***1***	4	***50***	***50***	100	1.5
CSD 329	**QA-II&VERBAL ABILITY**	HS	0	0	3	0	1	4	***100***	***0***	100	1.5
Total			**17**	**1**	**9**	**6**	**23**	**56**	**340**	**460**	**800**	**22.5**

IV Year Course structure – CSE(Data Science)												
Semester - I (Tentative)												
CODE	**SUBJECT NAME**	**Category**	**Periods**						**Sessionals Marks**	**Semester end Exam marks**	**Total Marks**	**Credits**
			L	**T**	**P**	**E**	**O**	**Total**				
CSD 411	OPEN ELECTIVE -III*	OE	3	0	0	1	2	6	40	60	100	3
CSD 412	PROFESSIONAL ELECTIVE -IV	PE	3	0	0	1	2	6	40	60	100	3
CSD 413	PROFESSIONAL ELECTIVE -V	PE	3	0	0	1	3	7	40	60	100	3
CSD 414	BUSINESS INTELLIIGENCE	HS	3	0	0	0	2	5	40	60	100	3
CSD 415	DATA ANALYTICS	SOC	2	1	0	1	4	8	40	60	100	3
CSD 416	***OOSE LAB***	***PC***	0	0	***3***	***0***	***1***	4	***50***	***50***	***100***	1.5
CSD 417	***DATA ANALYTICS LAB***	***SOC***	0	0	***3***	***0***	***1***	4	***50***	***50***	***100***	1.5
CSD 418	***PROJECT -PHASE 1***	***PR***	0	0	***3***	***0***	***1***	4	***100***	***0***	***100***	2
CSD 419	***SUMMER INTERNSHIP-INDUSTRY-2***	***PR***	0	0	***0***	***0***	***1***	1	***100***	***0***	***100***	2
Total			**14**	**1**	**9**	**4**	**17**	**45**	**500**	**400**	**900**	**22**

IV Year Course structure – CSE(Data Science)												
Semester - II (Tentative)												
CODE	**SUBJECT NAME**	**Category**	**Periods**						**Sessionals Marks**	**Semester end Exam marks**	**Total Marks**	**Credits**
			L	**T**	**P**	**E**	**O**	**Total**				
CSD 421	OPEN ELECTIVE -IV*	OE	3	0	0	1	3	7	40	60	100	3
CSD 422	***PROJECT PHASE 2/INTERNSHIP IN INDUSTRY***	***PR***	0	0	9	0	2	11	***100***	***100***	***200***	8
Total			**3**	**0**	**9**	**1**	**5**	**18**	**140**	**160**	**300**	**11**

Total Credits **160**

*Open Elective can be Inter Department Disciplinary Course,Emerging Courses or MOOC. Final decision will be taken by the department.

PROFESSIONAL ELECTIVES	
PE1	•CSD 312(A)Smart Systems Design & Programming •CSD312(B)Advanced Data Structures •CSD312(C) No SQL Data Bases •CSD 312(D)Artificial Intelligence
PE2	•CSD322(A) Deep Learning •CSD 322(B) Mobile Computing • CSD322(C) Digital ImageProcessing • CSD322(D)Data warehousing and Data mining
PE3	• CSD323(A)Distributed Operating Systems • CSD323(B)Embedded Systems • CSD323(C)Human Computer Interaction • CSD323(D)Pattern Recognition
PE4	• CSD412(A)Computer vision • CSD412(B)Bioinformatics • CSD 412(C)High Performance Computing • CSD415(D)Principles Of Programming Languages
PE5	• CSD413(A)IOT • CSD413(B)Fuzzy Computing • CSD413(C)Social Network Analysis • CSD413(D) Cloud Computing

OPEN ELECTIVES	
OE 1 3rd Yr-Sem -1	• Block Chain (as an emerging course)
OE2 3rd Yr-Sem -2	• Introduction to Machine Learning
OE3 4th Yr-Sem -1	Will be decided next year
OE4 4th Yr-Sem -2	Will be decided next year

BLOCKCHAIN	
CSD 311(OE as Emerging Subject)	**Credits: 3**
Instruction : 3 Periods / Week	Sessional Marks: 40
End Exam : 3 Hours	End Exam Marks: 60

Prerequisites:

- Knowledge of Data structures.
- Students must have knowledge of some programming languages (such as C, C++, and Java).

Course Objectives:

- Understand how blockchain systems (mainly Bitcoin and Ethereum) work.
- Design, build, and deploy smart contracts and distributed applications.
- Integrate ideas from blockchain technology into their own projects.

Course Outcomes:

By the end of the course, the student will be able to:	
1.	Explain the basic concepts and technology used for blockchain
2.	Describe the primitives of the distributed computing and cryptography related to blockchain.
3.	Illustrate the concepts of Bitcoin and their usage
4.	Analyze the working of Ethereum and Smart Contracts
5.	Design and build Smart Contracts.

Mapping of course outcomes with program outcomes:

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

SYLLABUS

UNIT-I: **8 Periods**

Introduction of Blockchain: Back-story of Blockchain, What is Block chain, Centralized Vs Decentralized Systems, Layers of Blockchain, Why Blockchain is Important? Blockchain uses and use cases, Public Vs Private Blockchains.

Learning Outcomes: At the end of this unit, Students are able to

1. Explain the need of Blockchain
2. Describe the benefits of Blockchain

UNIT II: **14 Periods**

How Blockchain Works: Laying the Blockchain foundation, Cryptography, Symmetric Key Cryptography, MAC and HMAC, Asymmetric Key Cryptography, Game Theory: Nash Equilibrium, Prisoner's Dilemma, Byzantine Generals Problem, Blockchain data structure, Merkle Trees, Properties of Blockchain Solutions, Distributed Consensus Mechanisms, Blockchain Applications, and Scaling Blockchain.

Learning Outcomes: At the end of this unit, Students are able to

1. Describe the Blockchain working.
2. Explain the different Cryptographic Concepts used in Blockchain

UNIT-III: **12 Periods**

How Bitcoin Works: The History of Money, Dawn of Bitcoin, What Is Bitcoin?, Working with Bitcoins, The Bitcoin Blockchain, Block Structure, The Genesis Block, The Bitcoin Network, Network Discovery for a New Node, Bitcoin Transactions, Consensus and Block Mining, Block Propagation, Bitcoin Scripts, Full Nodes Vs SPVs, Bitcoin Wallets.

Learning Outcomes: At the end of this unit, Students are able to

1. Illustrate the concept of Bitcoin
2. Explain different primitive concepts related to Bitcoin

UNIT-IV: **14 Periods**

How Ethereum Works: From Bitcoin to Ethereum: Ethereum as a Next-Gen Blockchain, Design Philosophy of Ethereum, **Enter the Ethereum Blockchain**: Ethereum Blockchain, Ethereum Accounts, Trie Usage, Merkle Patricia Tree, RLP Encoding, Ethereum Transaction and Message Structure, Ethereum State Transaction Function, Gas and Transaction Cost.
Ethereum Smart Contracts, Ethereum Virtual Machine and Code Execution, Ethereum Ecosystem.

Learning Outcomes: At the end of this unit, Students are able to

1. Describe How Ethereum works
2. Analyze how Smart Contracts are used in Ethereum

UNIT-V: **12 Periods**

Blockchain Application Development:
Interacting with the Bitcoin Blockchain, Setup and Initialize the bitcoinjs Library in a node.js Application, Interacting Programmatically with Ethereum—Sending Transactions, Interacting Programmatically with Ethereum—Creating a Smart Contract, Interacting Programmatically with Ethereum—Executing Smart Contract Functions.

Learning Outcomes: At the end of this unit, Students are able to

1. Design programs to interact with Blockchain
2. Build Smart Contracts

Text Books:

1. Bikramaditya Singhal, Gautam Dhameja and Priyansu Sekhar Panda, "Beginning Blockchain:A Beginner's Guide to Building Blockchain Solutions" 2018, Apress
2. Antonopoulos and G. Wood, "Mastering Ethereum" 1st Edition,2018, O'Reilly Publications

Reference Books:

1. Antonopoulos, "Mastering Bitcoin" 1st Edition, 2014, O'Reilly Publications

Web Resources:

1. **https://solidity-by-example.org/**
2. https://www.coursera.org/learn/blockchain-basics
3. https://cs251.stanford.edu/syllabus.html

Prepared By
Mr. S. Ratan Kumar, Associate Professor, Dept of CSE
Dr. Sivaranjani Reddy, HoD, Dept of CSE

SMART SYSTEM DESIGN & PROGRAMMING	
CSD 312(A)	**CREDITS: 3**
Instruction: 3 Theory / Week	**Sessional Marks: 40**
End Exam: 3 Hours	**End Exam Marks: 60**

Pre Requisites:

- Basic knowledge of Microprocessor & Interfacing, Computer Organization, Digital logic circuits
- Students must have knowledge of the C programming language.

Course Objectives:

- To learn the design and programming of microcontrollers.
- To learn the basics of ARM processors.
- To learn to program using ARM assembly language.
- To familiarize the students with Arduino kit and Raspberry Pi to implement small scale embedded system applications.

Course Outcomes:

By the end of the course, student will be able to	
CO-1:	Describe the Embedded system fundamentals, design and memory management.
CO-2:	Write programs in ARM based assembly level language.
CO-3:	Design Embedded system applications.
CO-4:	Test and debug embedded system applications.
CO-5:	Develop applications on Arduino and Raspberry Pi kits.

Mapping of Course Outcomes with Program Outcomes:

Mapping		**PO**												**PSO**	
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**1**	**2**
CO	**1**	2	1	1	1	1				1	2		1		
	2	3	2	3	2	2				1	2		1	3	
	3	3	3	3	3	2	1	1		3	2	2		3	
	4	2	3	3	3	2	2	2		3	2			3	
	5	3	2	3	2	2	1	2		2	2	2		2	

SYLLABUS

UNIT-I 10-12 Periods

Introduction to Embedded Systems - Application domain of embedded systems, Desirable features and general features, Figures of merit, classification of MCUs.
Hardware Point of View - Microcontroller Unit, Memory for embedded systems.
Examples – Mobile phone, Automotive electronics, RFID, WISENET, Robotics, Biomedical applications, Brain machine interface.
Learning Outcomes – At the end of this unit, students are able to

1. Know about the Embedded systems and their classifications.
2. Identify the different micro controller units used in verity of hardware units.

UNIT-II 10-12 Periods

Hardware Software Co-design and Embedded Product Development Lifecycle Management – Hardware Software Co-design, Modeling of systems, Embedded product development lifecycle management, Lifecycle models.
Embedded Design: A Systems Perspective – A typical example, Product design, The design process, Testing, Bulk manufacturing.
Learning Outcomes – At the end of this unit, students are able to

1. Extract information about the embedded product development lifecycle through different models.
2. Describe the design process of embedded products and manufacturing.

UNIT –III 15-20 Periods

ARM Architecture and Assembly Language Programming – History, Architecture, Interrupt vector table, Programming, ARM Assembly language, ARM instruction set, Conditional execution, Arithmetic, logical & compare instructions, Multiplication, Division, Starting ALP, General structure of an Assembly Language Line, Writing ALP, Branch instructions, Loading Constants, Load and Store instructions.
Learning Outcomes – At the end of this unit, students are able to

1. Develop the simple ARM assembly language programs using instruction set.
2. Analyze the programs developed by using branching and comparing.

UNIT-IV 10-12 Periods

Introduction to Arduino : What Is Physical Computing?. The Arduino Way, The Arduino Platform, Really Getting Started with Arduino. Advanced Input and Output. Troubleshooting. **Case study:** Automatic Garden-Irrigation System. .
Learning Outcomes – At the end of this unit, students are able to

1. Describes how to compute with Arduino platform and troubleshooting.
2. Demonstrate the Adriano systems with some real time applications.

UNIT-V **15-20 Periods**

Introducing the Raspberry Pi: The History of Raspberry Pi, Exploring the Pi Board, Hardware Requirements of the Pi, The Pi Operating System, Connecting the Peripherals, Configuring the Pi, Getting Started with Python, Accessing the GPIO Pins, Using the GPIO Library in python, Connecting the Temperature/Humidity Sensor, Setting Up the Motion Sensor. **Case Study**: Weather Station.

Learning Outcomes – At the end of this unit, students are able to

1. Recognize and explores Raspberry Pi boards and its functionality when connecting with peripherals.
2. Summarize accessing the GPIO pins and it library in python and analyze different sensors.

Text Books:

1. Das, Lyla B, *Embedded Systems: An Integrated Approach*,Pearson Education India, 2013.
2. Donat, Wolfram, *Learn Raspberry Pi Programming with Python*, Apress, 2014.
3. Banzi, Massimo, and Michael Shiloh, *Getting Started with Arduino: The Open Source Electronics Prototyping Platform*, Maker Media, Inc., 2014.

Reference Books:

1. Hohl, William, and Christopher Hinds. *ARM Assembly Language: Fundamentals and Techniques*, Crc Press, 2016.
2. Monk, Simon, *Raspberry Pi cookbook: Software and hardware problems and solutions*, O'Reilly Media, Inc., 2016.
3. Simon Monk, *30 Arduino™ Projects for the Evil Genius*, 2013, The McGraw-Hill Companies.

Web Resources:

1. http://nptel.ac.in/syllabus/117106111/
2. Muhammad Ali Mazidi, *ARM Assembly Language Programming & Architecture*, Kindle edition

---

Prepared By

Mr. S. Ratan Kumar, Assoc Professor,CSE Dept
Ms Keerthi Lingam, Asst Professor,CSE Dept
Ms G. Gowri Pushpa, Asst Professor,CSE Dept
Ms G. Santoshi, Asst Professor,CSE Dept

ADVANCED DATA STRUCTURES	
CSD 312(B)	**Credits: 3**
Instruction : 3 Periods / Week	Sessional Marks : 40
End Exam : 3 Hours	End Exam Marks : 60

Prerequisites:

- Knowledge of Data structures.
- Students must have the knowledge of some programming languages (such as C, C++, Java).

Course Objectives:

- Understand the variety of advanced data structures (skip lists, hash tables, priority queues, balanced search trees, graphs).
- Give the advantages and dis-advantages of each of the advanced data structure.
- Learn how to apply algorithm design techniques and data structures to solve problems.
- Learn different external sorting techniques and analyze their efficiency.

Course Outcomes:

By the end of the course, the student will be able to:	
1.	Implement the ADTs like linear list, skip list and hash tables and their operations
2.	Design and apply binary heaps, leftist heaps, Binomial queues and sorting for solving the real-world scenarios
3.	Describe the methods to balance a binary search tree using Rotation methods, Color changing methods, splitting and merging of nodes.
4.	Apply Algorithms for solving problems by using string matching techniques
5.	Solve problems using graph algorithms such as unweighted shortest path, graphs with negative edge cost.

Mapping of course outcomes with program outcomes:

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

SYLLABUS

UNIT-I: **12 Periods**

Skip lists and Hashing:
Sets, Map**,** Dictionaries, representation of dictionary as ADT, Linear list, skip list, hash table representation, collision resolution techniques, an application-text compression using dictionary.

Learning Outcomes: At the end of this unit, the students will be able to
1. Explain ADT, sets, maps and dictionaries.
2. Implement lists and hashing techniques.

UNIT-II: **12 Periods**

Priority Queues:
Binary heap, applications of binary heap, Applications of priority queues, leftist heaps, Binomial queues.
Sorting: Shell sort, Indirect sorting, bucket sort, External sorting.

Learning Outcomes: At the end of this unit, the students will be able to
1. Explain the operations of the priority queue.
2. Apply different sorting algorithms

UNIT-III: **12 Periods**

Balanced Search Trees:
Red-black trees, Representation of Red-black tree, Insertion, Deletion and searching of nodes in Red-black tree. Splay trees, B-Trees, Indexed Sequential Access Method (ISAM), B-Trees of order m, Representation of B-Tree, Insertion, deletion and searching a node in B-Tree, B+ trees of order m, Representation of B+ -Tree, Insertion, deletion and searching a node in B+-Tree

Learning Outcomes: At the end of this unit, the students will be able to
1. Explore balanced search trees.
2. Implement basic operations on the balanced search trees.

UNIT-IV: **12 Periods**

Digital Search Structures:
Digital Search trees, Binary Tries and Patricia, Multiway Tries.
String Matching:
Exact String Matching- Straight forward Algorithms, The Knuth-Morris-Pratt Algorithm, The Boyer-Moore Algorithm**.**

Learning Outcomes: At the end of this unit, the students will be able to
1. Explore digital search structures
2. Apply string matching algorithms to solve real time problems

UNIT-V: **12 Periods**

Graphs:
Graph algorithms-Topological sorting, shortest-path algorithms- unweighted shortest path, graphs with negative edge cost (Bellman–Ford), acyclic graphs, Network flow problems, Applications of BFS, DFS,

Learning Outcomes: At the end of this unit, the students will be able to

1. Construct the various types of graphs.
2. Apply the graph concepts to solve the real time problems.

Text Books:

1. SartajSahni, "Data Structures, Algorithms and Applications in C++", Second Edition, University Press
2. Adam Drozdek, Data Structures and Algorithms in C++, 3rd Edition, Cengage Learning 2008

Reference Books:

1. Richard F. Gilberg, Behrouz A. Forouzan, "Data Structures: A Pseudocode Approach with C", Second Edition, Cengage Learning.
2. NB Venkateswarulu and EV Prasad, "C and Data structures: A Snap Shot Oriented Treatise with Live Examples from Science and Engineering", S Chand, 2010.
3. Mark Allen Weiss, "Data Structures and Algorithm Analysis in C++", Third Edition, Pearson Education.

Web Resources:

1. http://nptel.ac.in/courses/106102064
2. http://nptel.ac.in/courses/106103069

Prepared By
Dr. Sangeetha Viswanadham, Prof, Dept of CSE
Mr. K. Chandrasekhar, Assistant Prof, Dept of CSE
Mrs. N. Lokeswari, Assistant Prof, Dept of CSE
Mrs. K. Amaravathi, Assistant Prof, Dept of CSE

NOSQL DATABASES	
CSD 312(C)	**Credits:3**
Instruction : 3 Periods /Week	Sessional Marks : 40
End Exam : 3 Hours	Ena Exam Marks : 60

Prerequisites:

- Knowledge on Relational Database management systems.

Course Objectives:

- Distinguish and describing how NoSQL databases differ from relational databases from theoretical perspective.
- Explore the origins of NoSQL databases and the characteristics .
- Demonstrate competency in selecting a particular NoSQL database for specific use cases.
- Demonstrate Document databases with MongoDB.

Course Outcomes:

By the end of the course, the student will be able to:	
1.	Compare and contrast the uses of relational RDBMSs and NoSQL systems for different types of data and applications.
2.	Differentiate various data models.
3.	Recognize Key value Databases and document databases.
4.	Create a sample database using NoSql.
5.	Apply the Query concepts in MongoDB database

Mapping of course outcomes with program outcomes:

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

SYLLABUS

UNIT-I : **12 Periods**

Why NoSQL?

The value of relational databases – Impedance mismatch – Application and integration databases – Attack of the cluster.

Learning Outcomes: The student will be able to

1. Recall Relational databases and security aspects in Realtime
2. Identify working with multiple databases.

UNIT-II : **12 Periods**

Aggregate Data Models :

Aggregates - Example of Relations and Aggregates – Consequences of Aggregate Orientation - Key-Value and Document Data Models - Column-Family Stores

More Details on Data Models :

Relationships - Graph Databases – Schema less Databases - Materialized Views - Modeling for Data Access.

Learning Outcomes: The student will be able to

1. Explain internal operations on Database.
2. Analyze how to view data from database in different ways.

UNIT –III : **12 Periods**

Distribution Models :

Single Server – Shading - Master-Slave Replication - Peer-to-Peer Replication – Combining Shading and Replication

Consistency:

Update Consistency - Read Consistency - Relaxing Consistency - The CAP Theorem - Relaxing Durability.

Learning Outcomes: The student will be able to

1. Analyze how multiple clients can interact with database server.
2. Apply updating values dynamically in database.

UNIT-IV : **12 Periods**

Key-Value Databases:

What Is a Key-Value Store - Key-Value Store Features – Consistency – Transactions - Query Features - Structure of Data – Scaling - Suitable Use Cases - Storing Session Information - User Profiles, Preferences - Shopping Cart Data - When Not to Use - Relationships among Data - Multioperation Transactions - Query by Data - Operations by Sets.

Learning Outcomes: The student will be able to

1. Use the way how Operations used in Real-time applications.
2. Solve working with the database transactions.

UNIT-V : **12 Periods**

Document Databases:

What Is a Document Database? – Features – Consistency – Transactions – Availability - Query Features – Scaling - Suitable Use Cases - Event Logging - Content Management Systems - Blogging Platforms - Web Analytics or Real-Time Analytics - E-Commerce Applications - When Not to Use - Complex Transactions Spanning Different Operations - Queries against Varying Aggregate Structure.

Introduction to MongoDB:

Introduction to MongoDB - The Data Model - Working with Data – GridFS.

Learning Outcomes: The student will be able to

1. Integrate database server like MongoDB to cloud apps.
2. Test Cloud data storage.

Text Book:

1. Pramod J.Sadalag and Martin Fowler," *NoSQL Distilled, A Brief Guide to the Emerging World of Polyglot Persistence*" ,1st Edition, Addison Wesley
2. David Hows, Eelco Plugge, Peter Membrey , and Tim Hawkins, "*The definitive guide to MongoDB"*, "*A complete guide to dealing with big data using MongoDB".* 1st Edition, Apress

Reference Books:

1. Shashank Tiwari, Professional NoSQL, Wrox Press, Wiley, 2011, ISBN: 978-0-470-94224-6
2. Gaurav Vaish, Getting Started with NoSQL, Packt Publishing, 2013.

Web Resources:

1. http://allvideolectures.com/courses/course/96uv57kBOZ.
2. https://university.mongodb.com/

Prepared By
Professor KVSVN Raju, Dept of CSE
Mrs.A.Deepthi, Assistant Prof, Dept of CSE
Mrs.A.N.R.Latha, Assistant Prof, Dept of CSE
Mr. S. Bosu babu, Assistant Prof, Dept of CSE

ARTIFICIAL INTELLIGENCE	
CSD 312(D)	**Credits: 3**
Instruction : 3 Periods / Week	Sessional Marks : 40
End Exam : 3 Hours	End Exam Marks : 60

Prerequisites:

- Probability and Discrete Mathematics, Data Structures and Algorithms , Formal languages and Automata Theory.

Course Objectives:

- To discuss about the Basic principles, techniques, and applications of artificial intelligence.
- To analyze and apply the insights into knowledge representation, problem-solving, and learning methods in Science and Engineering domains.

Course Outcomes:

By the end of the course, the student will be able to:	
1.	Illustrate foundations of Artificial Intelligence (AI) and its applications, Problem Types, Characteristics and Search Space Representations.
2.	Apply Search Techniques (Brute-Force, Heuristic and Game Playing) of Artificial Intelligence and solving AI problems by applying suitable searching methods.
3.	Discuss Knowledge Representation, Approaches and Types and apply to process through different reasoning approaches such as Propositional and Predicate Calculus.
4.	Explain different types of Learning in Artificial Intelligence and apply planning in real-time world examples
5.	Discuss about Uncertainty and its importance and classify the various approaches of the Expert systems using case studies.

Mapping Of Course Outcomes With Program Outcomes:

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

SYLLABUS

UNIT I: **10 Hours**

FOUNDATIONS OF ARTIFICIAL INTELLIGENCE:
AI problems, foundation of AI and history of AI intelligent agents: Agents and Environments, the concept of rationality, the nature of environments, structure of agents, problem solving agents, problem formulation, AI Techniques, Problem Types and Characteristics, State Space Search, Production Systems and its characteristics, Applications of Artificial Intelligence.

Learning Outcomes: At the end of this unit, Student will be able to

1. Discuss foundations of AI and its agents, environment and rationality and also about problem solving agents.
2. Explain different types of AI problems as Search space representation, characteristics and techniques.

UNIT II: **12 Hours**

SEARCHING:
Searching- Searching for solutions, uniformed search strategies – Breadth first search, depth first Search, Bi-Directional Search and Uniform-Cost Search.
Informed Search Algorithms (Heuristic search): Introduction, Heuristic evaluation function, Generate-and-Test, Best-First Search, A* Algorithm, Problem Reduction Algorithm, AO* Algorithms, Hill climbing, Simulated Annealing, Constraint Satisfaction Algorithm (CSP).
Game Playing - Adversarial search, Games, mini-max algorithm, optimal decisions in multiplayer games, Problem in Game playing, Alpha-Beta pruning, Evaluation functions.

Learning Outcomes: At the end of this unit, Student will be able to

1. Apply Uninformed search algorithms on AI problems
2. Apply Heuristic and Game Playing techniques on selective AI Problems

UNIT III: **10 Hours**

KNOWLEDGE REPRESENTATION AND REASONING:
Knowledge and Reasoning: Knowledge Representation Issues: Representations and Mappings, Approaches to Knowledge Representation, Issues in Knowledge Representation; Using Predicate Logic: Representing Simple Facts in Logic, Representing Instance and ISA Relationships, Computable Functions and Predicates, Resolution, Natural Deduction;
Representing Knowledge Using Rules: Procedural versus Declarative Knowledge, Logic Programming, Forward versus Backward Reasoning, Matching, Control Knowledge.

Learning Outcomes: At the end of this unit, Student will be able to

1. Discuss representation of Knowledge and reasoning, issues, approaches and applications.
2. Apply Propositional and Predicate logic through Knowledge Representation and case studies.

UNIT 4: **9 Hours**

LEARNING AND PLANNING:

Learning: Introduction, Types of Learning, Supervised Learning, Unsupervised Learning, Reinforcement Learning, Applications of Learning, Case-Based Reasoning

Planning: Overview, An Example Domain: The Blocks World, Components of a Planning System, Goal Stack Planning, Hierarchical Planning;

Learning Outcomes: At the end of this unit, Student will be able to

1. Discuss different Learning approaches of AI inorder to solve real world problems.
2. Discuss and Apply Planning on real-world problems such as Block-World, Wearing a shoe,etc.

UNIT 5: **9 Hours**

Reasoning in Uncertain Environments and Expert Systems:

Probabilistic Reasoning: Statistical Reasoning: Probability and Baye's Theorem, Bayesian Networks, Dempster-Shafer Theory

Expert systems:- Introduction, basic concepts, structure of expert systems, the human element in expert systems how expert systems works, problem areas addressed by expert systems, expert systems success factors, types of expert systems and Applications.

Learning Outcomes: At the end of this unit, Student will be able to

1. Discuss about Uncertainty using Probabilistic and Statistical reasoning approaches
2. Illustrate foundations of Expert Systems and how to apply in real-time Scenario.

Text Books:

1. Elaine Rich, Kevin Knight, Shivashankar B. Nair, "*Artificial Intelligence"*, Third Edition,2009
2. George F Luger: " *Artificial Intelligence, Structures, Strategies for Complex Problem Solving"*, 3rd Edition
3. Stuart Russell and Peter Norvig, *Artificial Intelligence: A Modern Approach*,3rd Edition,2016

Reference Books:

1. David L Poole, Alan K. Mackworth, *Artificial Intelligence: Foundations of ComputationalAgents, Cambridge University Press*,2010
2. Dan W. Patterson, *Introduction to Artificial Intelligence and Expert Systems*, PHI Cherry Bhargava, Pardeep Kumar Sharma - *Artificial Intelligence. Fundamentals and Applications-CRC Press (2022)*
3. Amit Konar - *Artificial intelligence and soft computing: behavioral and cognitive modeling of the human brain-CRC Press (2000)*

Web References:

1. https://nptel.ac.in/courses/106/105/106105077/
2. https://www.udemy.com/course/natural-language-processing-with-python-and-nltk/

Prepared By
Dr. P.E.S.N. Krishna Prasad , Prof, Dept of CSE
Mrs. D. Aswini, Assistant Prof, Dept of CSE
Mrs. A. Yasaswi, Assistant Prof, Dept of CSE

DATA ANALYSIS AND VISUALIZATION	
CSD 313	**Credits: 3**
Instruction: 3 Periods / Week	Sessional Marks: 40
End Exam: 3 Hours	End Exam Marks: 60

The theme of this course is to understand the fundamental representation and analysis of data and the ways to visualize in quantitative and qualitative approaches by applying data analysis and visualization techniques.

Prerequisites:

- Number Theory and Linear Algebra, Probability and Statistics, Data Structures, Programming languages.

Course Objectives:

- To understand and analyze the data in different variants with quantitative and qualitative approaches.
- To apply different visualization techniques that projects effective and efficient representations using various Visualization tools

Co**urse Outcomes:**

By the end of the course, the student will be able to:	
1.	Discuss the basic foundations of visual data analysis and its influence factors.
2.	Discuss the basic concepts of graphics and its presentations, issues.
3.	Apply different data visualization techniques through graph representations.
4.	Apply multivariate data visualization and visualization of temporal, geo, and spatial data.
5.	Discuss dimensionality reduction and automatic analysis.

Mapping of course outcomes with program outcomes:

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

SYLLABUS

Unit 1: Introduction to Data Visualization: 12 Periods

A brief History of Data Visualization (Book 2), Basic Considerations – Visualization, Interaction and Computation, Five Ws of Interactive Visual Data Analysis, Enhancing the Data Analysis, Influencingfactors, Process models (Book 1).

Learning Outcomes:

1. To discuss foundations of data visualization and interactive aspects.
2. To discuss influence factors and process models of data analysis

Unit 2: Graphics Representations: (Book 2) 12 Periods

Good Graphics: Content, Context and Construction, Presentation Graphics and Exploratory Graphics, Presentation (What to Whom, How and Why), Scientific Design Choices in Data Visualization, Higher- dimensional Displays and Special Structures.
Static Graphics: Complete Plots, Customization, Extensibility, and other Issues,

Learning Outcomes:

1. To discuss presentation and explorative graphics in data visualization.
2. To discuss and apply static graphics models for data visualization

Unit 3: Data Visualization through their Graph Representations: 12 Periods

(Book 2) Introduction, Data and Graphs, Graph layout Techniques, Graph Drawing.

Learning Outcomes:

1. To discuss the importance of data and graphs.
2. To apply various graph layouts in different kinds of data for visualization

Unit 4: Visualization Methods and Techniques: (Book 1) 12 Periods

Visual Encoding and Presentation, Multivariate Data Visualization, Visualization of Temporal Data,Visualization of Geo-spatial Data, Graph Visualization.
Interacting with Visualizations: Requirements for Efficient Interaction, Basic Operations for Interaction,Interactive Selection and attenuation, Navigating Zoomable Visualizations, Comparisons.

Learning Outcomes:

1. To discuss different types of data visualizations.
2. To discuss interactive and navigation environments of Visualizations and comparisons.

Unit 5: Automatic Analysis: (Book 1) 12 Periods

Visual Representations, Focusing on Relevant Data, Abstracting Data, Grouping Similar Data Elements, Dimensionality Reduction. Visualization in Multi-Display Environments, Guiding the user and ProgressiveVisual Data Analysis.

Learning Outcomes:

1. To apply focus on data and abstracting, grouping and reduction methods.
2. To discuss visualization in multi-environments.

Text Books:

1. Christian Tominski, Heidrun Schumann, "Interactive Visual Data Analysis", CRC Press, 2020
2. Chun-houh Chen, Wolfgang Hardle, Antony Unwin, "Handbook of Data Visualization", Springer2008.

Reference Books:

1. Nurul Haszeli Ahmad, Tableau for Beginner Data Analysis and Visualization 101 (2020)
2. Hwang, Jaejin, Yoon, Youngjin, Data Analytics and Visualization in Quality Analysis using Tableau,CRC Press, 2022
3. Stephen Few, Show me the Numbers – Designing the Tables and Graphs to Enlighten, SecondEdition, Analytics Press, 2012
4. Tony Fischetti, R Data Analysis and Visualization-Packt (2016)
5. Jay Jacobs, Bob Rudis - Data-Driven Security Analysis, Visualization and Dashboards-Wiley (2014)
6. Alex Campbell, Data Visualization Guide Clear Introduction to Data Mining, Analysis, andVisualization, 2021

Prepared By

Mr. P. Krishnanjaneyulu, Assistant Prof, Dept of CSE
Dr. P.E.S.N. Krishna Prasad, Prof, Dept of CSE
Mr. R. Siva Kiran, Assistant Professor, Dept of CSE

COMPETITIVE PROGRAMMING	
CSD 314	**CREDITS:3**
INSTRUCTION: 2 Periods & 1 Tutorial/Week	**SESSIONAL MARKS: 40**
FINAL EXAM: 3Hrs	**FINAL EXAM MARKS: 60**

Pre requisites:

- C and Data Structures

Course Objectives:
The focus of the course is to

- Development and implementation of advanced algorithms
- Develop the skills required for programming competitions.
- Learn to select appropriate algorithms for a given problem, integrate multiple algorithms
- Solving a complex problem, designing new algorithms, and implementing them.
- Solving problems in teams and working under time pressure.

Course Outcomes:
By the end of the course Students will be able to

1.	Identify type of the problem and Apply sorting, searching techniques to solve problems.
2.	Perform bit manipulation and string manipulation operations.
3.	Compute the given complex problem using mathematical theorem and graph algorithms.
4.	Solve given problems using dynamic programming and Greedy Approach.
5.	Analyze and develop backtracking algorithms and Geometrical Algorithms.

CO-PO Mapping

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

SYLLABUS

UNIT I **12 Periods**

Problem Solving Paradigms:
Overview and Motivation, Complete Search, Divide and Conquer, Greedy, Dynamic Programming, Brute force, **Sorting & Search**: Sorting Theory, Counting Sort, Radix Sort, Heap Sort, Bucket Sort, Ternary Search.
Case Studies: The 3n + 1 Problem (UVa IDs: 100), Vito's Family (UVa IDs: 110401/10041)
Learning Outcomes:

1. Identify the type of the problem statement.
2. Apply sorting and searching techniques to solve the given problem.

UNIT II **12 Periods**

Bit Manipulation: Bit representation, bit operations, representing sets
Strings: String terminology, Trie structures, String hashing, Z-algorithm
Case Studies: BITMASK (UVAID: 10718), WERTYU(UVAIDs:110301/10082)
Learning Outcomes:

1. Apply the bit operators to solve given problem.
2. Implement various string operations and patter matching algorithms.

UNIT III **12 Periods**

Number theory: Primes and factors, Modular arithmetic **Graphs:** Flows and Cuts: Ford-Fulkerson algorithm, Path Covers **Paths and circuits**: Eulerian paths.
Case Studies: Mine sweeper (UVA ID: 10189), Tourist Guide (UVa IDs: 111006/10199)
Learning Outcomes:

1. Solve the complex problem using mathematical theorem.
2. Analyzing Graph algorithms to solve suitable problems.

UNIT IV: Dynamic programming & Greedy Algorithms **12 Periods**

Dynamic Programming: Coin problem, Longest increasing subsequence, Paths in a grid, Edit distance, Counting Tilings
Greedy Algorithms: coin problem, tasks and deadlines, Minimizing sums, Data compression
Case Studies: Is Bigger Smarter? (UVa Ids:10131), Distinct Subsequences (UVa Ids:10069)
Learning Outcomes:

1. Solve complex problem using dynamic programming approach.
2. Solve complex problem using Greedy approach.

UNIT V: Computational Geometry & Backtracking **12 Periods**

Computational Geometry :Points and lines, Polygon (Convex, Concave), Distance functions. **Backtracking:** Constructing All Subsets, Constructing All Permutations.
Case Studies: 15-Puzzle Problem (UVAID:10181), The Closest Pair (UVa IDs: 10245)
Learning Outcomes:

1. Implement Geometrical algorithms.
2. Analyze and develop backtracking algorithms.

Text Books:

1. Steven Halim, Felix Halim, “Competitve programming 3”, Handbook for ACMICPC and IOI contestants.
2. Antti Laaksonen . “Competitive Programmer’s Handbook”, August 19, 2019

Reference Books:

1. Steven S. Skiena Miguel A. Revilla “Programming challenges”, The Programming Contest Training Manual, Springer.
2. Antti Laaksonen, “Guide to Competitive Programming: Learning and Improving Algorithms Through Contests”,
3. Ahmed Shamsul Arefin, “Art of Programming Contest” , special online edition for UVA online judge users

Online Resources:

1. https://jadi.net/wp-content/uploads/2017/07/competetive-programmers-handbook.pdf
2. http://acm.cs.buap.mx/downloads/Programming_Challenges.pdf
3. https://www.comp.nus.edu.sg/~stevenha/myteaching/competitive_programming/cp1.pdf
4. https://www.comp.nus.edu.sg/~stevenha/database/Art_of_Programming_Contest_SE_for_uva.pdf

Problem reading & Practice:

https://uva.onlinejudge.org/index.php

Prepared By
Mrs. B. Siva Jyothi, Assistant Prof, Dept of CSE
Mr. S Ratan Kumar, Associate Prof, Dept of CSE
Dr K Prasad, Assistant Prof, Dept of CSE
Mrs. Marline Joys, Assistant Prof, Dept of CSE

DATA BASE MANAGEMENT SYSTEMS	
CSD 315	**Credits : 3**
Instruction **:** 3 Periods /week	Sessional Marks : 40
End Exam : 3 Hours	End Exam Marks : 60

Prerequisites:

- Elementary knowledge about computers including some experience using UNIX or Windows. Knowledge about data structures and algorithms, corresponding to the basic course on Data Structures and Algorithms.

Course Objectives:

- Understand basic database concepts, including the structure and operation of the relational data model.
- Understand logical database design principles, including E-R diagrams and database normalization.
- To learn the basics of SQL and construct queries using SQL.
- Understand the concept of database transaction and concurrency control, backup and recovery, data object locking and protocols.

Course Outcomes:

By the end of the course, the student will be able to:	
1.	Describe basic concepts of database systems and principles of transaction processing, concurrency techniques and recovery of database.
2.	Apply Conceptual and logical database design principles, including E-R diagrams.
3.	Compose SQL queries to perform operations on database. (Create, Retrieve, Update, Delete)
4.	Construct relational algebra expressions for queries
5.	Analyze and apply schema Refinement ,database normalization principles.

Mapping of Course Outcomes with Program Outcomes:

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

SYLLABUS

UNIT-I: 12 periods

Introduction to DBMS:

Overview of DBMS, File system versus a DBMS , Advantages of a DBMS, Three Schema architecture of DBMS, Data Models, Database Languages, Transaction Management , Structure of a DBMS ,Client/Server Architecture ,Database Administrator and Users.

Entity-Relationship Model:

Design Issues, ER Modeling concepts , Cardinality constraints, Weak-entity types, Subclasses and inheritance, Specialization and Generalization, Conceptual Database Design With the ER Model.

Learning Outcomes: At the end of this unit the Students will be able to

1. To design and build a simple database system and demonstrate competence with the fundamental tasks involved with modeling, designing, and implementing a DBMS.
2. Describe the fundamental elements of relational database management systems and design ER-models to represent simple database application scenarios

UNIT-II: 10 periods

Relational Model:

Structure of Relational Databases, Basics of Relational Model ,Integrity Constraints, Logical Database Design, Introduction to Views, Destroying/ Altering Tables and Views, Relational Algebra, Relational Calculus.

Learning Outcomes: At the end of this unit the Students will be able to

1. Design entity relationship and convert entity relationship diagrams into RDBMS and formulate SQL queries on the respect data into RDBMS and formulate SQL queries on the data.
2. Recall Relational Algebra concepts, and use it to translate queries to Relational Algebra statements and vice versa.

UNIT-III: 12 periods

SQL:

Concept of DDL, DML, DCL, Set operations, Nested queries, Aggregate Functions, Null Values, Referential Integrity Constraints, assertions, views, Embedded SQL ,Cursors Stored procedures and triggers, ODBC and JDBC, Triggers and Active Database, designing active databases.

Learning Outcomes: At the end of this unit the Students will be able to

1. Perform PL/SQL programming using concept of Cursor Management, Error Handling, Package and Triggers.
2. Use the basics of SQL and construct queries using SQL in database creation and interaction

UNIT-IV: **12 periods**

Database Design:

Schema Refinement, Functional Dependencies, Reasoning about Functional Dependencies, Normalization using functional dependencies, Decomposition, Boyce-Codd Normal Form, 3NF, Normalization using multi-valued dependencies, 4NF, 5NF

Security:

Access Control, Discretionary Access Control - Grant and Revoke on Views and Integrity Constraints, Mandatory Access Control.

Learning Outcomes: At the end of this unit the Students will be able to

1. Apply various Normalization techniques , functional Dependency and Functional Decomposition to improve the database design.
2. Implement typical security techniques in real time applications.

UNIT-V: **15 periods**

Transaction Management:

The ACID Properties, Transactions & Schedules, Concurrent Execution of Transactions, Lock-Based Concurrency Control.

Concurrency Control: 2PL, Serializability and Recoverability, Introduction to Lock Management, Lock Conversions, Dealing with Deadlocks, Specialized Locking Techniques, Concurrency Control without Locking.

Crash Recovery: Introduction to ARIES, The Log, Other Recovery-Related Structures, The Write-Ahead Log Protocol, Check pointing, recovering from a System Crash, Media Recovery.

Learning Outcomes: At the end of this unit the Students will be able to

1. Execute various SQL queries related to Transaction Processing & Locking using the concept of Concurrency control.
2. Analyze the crash recovery techniques of database systems and apply transaction processing mechanisms in relational databases.

Text Books:

1. Raghu Ramakrishnan, Johannes *Gehrke "Database Management Systems*", 4th Edition, McGraw- Hill 2003

References Books:

1. A.Silberschatz.H.Korth, *"Database System Concepts"* , 6th Edition,2010 McGraw-Hill

Web Resources:

1. https://www.oreilly.com/library/view/web-database-applications/0596005431/ch01.html
2. http://nptel.ac.in/courses/IIT-MADRAS/Intro_to_Database_Systems_Design/pdf/1_Introduction.pdf
3. https://www.edx.org/learn/databases
4. https://www.youtube.com/watch?v=1057YmExS-I

Prepared By
Prof. K.V.S.V.N. Raju, Prof, Dept of CSE
Mr. V. Srinivasa Raju, Assistant Prof, Dept of CSE

DATA SCIENCE WITH PYTHON LAB	
Code: CSD 316	**Credits : 1.5**
Instruction :3 Periods & 1 Tut/Week	**Sessional Marks : 50**
End Exam : 3 Hours	**End Exam Marks : 50**

Prerequisites:

- Basic knowledge on Probability and statistics.
- Basics of Python Programming.

Course Objectives:

- To train the students to apply the principles of data analytics to analyze and effectively visualize the data.
- Train the students to gain the knowledge of computational statistical approaches and their application to a variety of datasets.
- Practical way of Understanding the nature of analysis of variance and Multivariate statistical models.

Course Outcomes:

By the end of the course, the student will be able to:	
1.	Work with different IDE and Python notebooks for mathematical, scientific applications and for data analysis.
2.	Explore Pandas, matplotlib and seaborn packages. Write example programs to visualize the data.
3.	Implement data pre processing, validation methods on given data sets and interpret the results.
4.	Work with Linear Regression and Multiple Regression models in Python.

S.No	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO 11	PO12	PSO1	PSO2
CO 1	-	-	-	-	3	-	-	-	-	-	-	-	-	-
CO 2	1	-	-	-	1	-	-	-	-	-	-	-	-	-
CO 3	1	1	1	-	1	-	-	-	-	-	-	-	-	-
CO 4	1	1	1	-	1	-	-	-	-	-	-	-	-	-
CO5	1	1	1	-	1	-	-	-	-	-	-	-	-	-

Week 1.

1. Getting familiarity with Python IDE and Notebooks.
2. Write a Program in Python to Read and write different types of Files (csv, json, txt etc).

Week 2.

1. Getting familiarity with Numpy.
2. Write a Program in Python to Manipulate, Aggregate and Analyze data using Numpy.

Week 3.

1. Getting familiarity with pandas.
2. Write a Program in Python to Handle and Analyze data using Pandas.

Week 4.

1. Write a Program in Python to perform statistical analysis on given Data set.
2. Apply basic statistical methods on Sample Datasets (Data sets from https://www.kaggle.com/datasets, https://archive.ics.uci.edu/ml/datasets.php etc.,)

Week 5.

1. Working with vectors and matrices in Python.
2. Working with matplotlib and seaborn packages in Python.

Week 6.

1. Write a Python program using Scikit-learn to print the keys, number of rows-columns, feature names and the description of the Iris data.
2. Write a Python program to get the number of observations, missing values and nan values from the given data set.
3. Write a Python program to get observations of each species (setosa, versicolor, virginica) from iris data and plot it using seaborn or matplotlib packages.

Week 7.

1. Write a Python program to create a plot to get a general Statistics of Iris data.
2. Write a Python program to create a plot to get a general Statistics of Iris data.
3. Write a Python program to create a Pie plot to get the frequency of the three species of the Iris data.
4. Write a Python program to create a graph to find relationship between the sepal length and width.

Week 8.

1. Write a Python program to create a graph to see how the length and width of SepalLength, SepalWidth, PetalLength, PetalWidth are distributed.
2. Write a Python program to create a joinplot to describe individual distributions on the same plot between Sepal length and Sepal width.
3. Write a Python program using seaborn to Create a kde (Kernel Density Estimate) plot of sepal_length versus sepal width for setosa species of flower.
4. Write a Python program using seaborn to Create a kde (Kernel Density Estimate) plot of petal_length versus petal width for setosa species of flower.
5. Write a Python program using seaborn to Create a kde (Kernel Density Estimate) plot of petal_length versus petal width for setosa species of flower.

Week 9.

1. Write a Python program to split the iris dataset into its attributes (X) and labels (y). The X variable contains the first four column.
2. Write a Python program using Scikit-learn to split the iris dataset into 70% train data and 30% test data. Out of total 150 records, the training set will contain 120 records and the test set contains 30 of those records. Print both datasets.
3. Write a Python program using Scikit-learn to convert Species columns in a numerical column of the iris dataframe. To encode this data map convert each value to a number. e.g. Iris-setosa:0, Iris-versicolor:1, and Iris-virginica:2. Now print the iris dataset into 80% train data and 20% test data. Out of total 150 records, the training set will contain 120 records and the test set contains 30 of those records. Print both datasets.
4. Write a Python program using Scikit-learn to split the iris dataset into 70% train data and 30% test data. Out of total 150 records, the training set will contain 105 records and the test set contains 45 of those records. Predict the response for test dataset (SepalLengthCm, SepalWidthCm, PetalLengthCm, PetalWidthCm) using the K Nearest Neighbor Algorithm. Use 5 as number of neighbors.

Week 10.

1. Write a Program in Python to add an indeed field, changing misleading data fields, Re-expressing categorical data as numerical data, standardizing numerical fields and identifying outliers for data preparation phase. for bank marketing data set. (https://www.kaggle.com/datasets/janiobachmann/bank-marketing-dataset)
2. Write a Program in Python to perform Simple Linear Regression.
3. Write a Program in Python to predict House rent using linear regression.

Week 11.

1. Write a Python program using Scikit-learn to split the iris dataset into 80% train data and 20% test data. Out of total 150 records, the training set will contain 120 records and the test set contains 30 of those records. Train or fit the data into the model and calculate the accuracy of the model using the K Nearest Neighbor Algorithm.
2. Write a Python program using Scikit-learn to split the iris dataset into 80% train data and 20% test data. Out of total 150 records, the training set will contain 120 records and the test set contains 30 of those records. Train or fit the data into the model and using the K Nearest Neighbor Algorithm calculate the performance for different values of k.
3. Write a Python program using Scikit-learn to split the iris dataset into 80% train data and 20% test data. Out of total 150 records, the training set will contain 120 records and the test

set contains 30 of those records. Train or fit the data into the model and using the K Nearest Neighbor Algorithm and create a plot to present the performance for different values of k.

4. Write a Python program using Scikit-learn to split the iris dataset into 80% train data and 20% test data. Out of total 150 records, the training set will contain 120 records and the test set contains 30 of those records. Train or fit the data into the model and using the ID3 Algorithm and plot the accuracy.

Week 12.

1. Write a Python program to search the Street address, name from a given location information using Nominatim API and GeoPy package.
2. Write a Python program to search the country name from given state name using Nominatim API and GeoPy package.
3. Write a Python program to find the details of a given zip code using Nominatim API and GeoPy package.
4. Write a Python program to find the latitude and longitude of a given location using Nominatim API and GeoPy package.
5. Write a Python program to find the location address of a specified latitude and longitude using Nominatim API and Geopy package.
6. Write a Python function to get the city, state and country name of a specified latitude and longitude using Nominatim API and Geopy packages.

Week 13.

1. Write a Program in Python to implement Correlation.
2. Write a Program in Python to perform Multiple Regression.

Week 14.

1. Write a Program to apply Multiple Linear Regression Using Python and Scikit-learn.

Text Books:

1. Applied Univariate, Bivariate, and Multivariate Statistics Using Python, Daniel J. Denis, Wiley, First Edition.

Reference Books:

1. Applied Multivariate Statistical Analysis, Richard. A. Johnson and Dean.W. Wichern, Pearson Prentice Hall, 6th Edition, 2007.
2. An Introduction to Multivariate Statistical Analysis, T.W. Anderson, Wiley, 3rd Edition, 2003.

Web Resource:

1. https://www.westga.edu/academics/research/vrc/univariate-bivariate-analyses.php

---

Prepared By

Mr. R. Siva Kiran, Assistant Prof, Dept of CSE
Dr. K. Prasad, Assistant Prof, Dept of CSE
Mr. P. Krishnanjaneyulu, Assistant Prof, Dept of CSE
Mr. S. Bosubabu, Assistant Prof, Dept of CSE

DATABASE MANAGEMENT SYSTEMS LAB	
CSD 317	**Credits : 1.5**
Instruction **:** 3 Periods/Week	Sessional Marks : 50
End Exam : 3 Hours	End Exam Marks : 50

Prerequisites:

- Elementary knowledge about computers including some experience using UNIX or Windows. Knowledge about data structures and algorithms, corresponding to the basic course on Data Structures and Algorithms.

Course Objectives

- To understand the basics of SQL and construct queries using SQL.
- To learn connectivity between web pages, OLAP, OLTP.

Course Outcomes:

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

CO1	Make use of basic SQL queries to solve simple problems.
CO2	Solve complex queries using nested queries and joins.
CO3	Construct triggers, views and stored procedures for different scenarios
CO4	Apply the principles of ER model and normalization for schema refinement in logical database design.

.

Articulation Matrix for CO-PO Mapping:

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

SYLLABUS

List of Experiments:

EXPERIMENT NAME	MAPPED TO
1. SQL DDL ,DML Statements	CO1
2. SQL Constraints.	CO1
3. Inbuilt functions in RDBMS.	CO2
4. Aggregate functions	CO2
5. Nested Queries & Join Queries.	CO2
6. Creation and dropping of Views.	CO3
7. Creating Triggers.	CO3
8. Stored Procedures.	CO3
9. Developing a sample application which includes all database design steps like requirements analysis, logical database design, normalization, developing user interface to access database from the application.	CO4

Sample Applications:

1. Development of an Online Course Portal for a campus
2. Book Bank Management System
3. Car Rental Management System
4. Exam/academic system for College Management
5. Real estate Management system
6. University Management System
7. Database manager for a Magazine agency or a newspaper agency
8. Ticket booking for performances
9. Inventory Control System
10. Students management System

REFERENCE BOOKS:

1. Raghu Ramakrishnan, Johannes *Gehrke "Database Management Systems*", 3rd
2. Edition,2003, McGraw- Hill,
3. A.Silberschatz.H.Korth, *"Database System Concepts"* , 6th Edition, 2010,McGraw-Hill

WEB REFERENCES:

1. https://dev.mysql.com/doc/refman/5.5/en/sql-syntax-data-definition.html
2. https://cse.iitkgp.ac.in/~pabitra/course/dbms/dbms_lab.html

---

Prepared By

Prof. K.V.S.V.N. Raju, Prof, Dept of CSE

Mr. V. Srinivasa Raju, Assistant Prof, Dept of CSE

COMPETITIVE PROGRAMMING LAB	
CSD 318	**Credits : 1.5**
Instruction **:** 3 Periods /Week	Sessional Marks : 50
End Exam : 3 Hours	End Exam Marks : 50

Prerequisites:

- C and data Structures

Course objectives:

- Learn to select appropriate algorithm for a given problem, integrate multiple algorithms
- Solve a complex problem, design new algorithms, and implement them.

Course outcomes:

By the end of the course, the student will be able to:	
1.	Apply the basic, sorting and searching techniques to solve the problem components etc.
2.	Analyze the concepts of path finding algorithms for flows and cuts, strings and greedy algorithms
3.	Develop solutions for the back tracking algorithms and bit manipulations
4.	Solve the number theory and knowledge of dynamic programming to the real time scenario.

Mapping of Course Outcomes with Program Outcomes:

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

List of Experiments

1. Maximum subarray sum **CO1**

Solve the Maximum subarray sum with different time complexities.

2. The Median of Two Sorted Arrays **CO1**

Program to find the median of two sorted arrays of same size and different size are discussed here. Firstly, let us see what is median of the array? Median is an element which divides the array into two parts - left and right. So the number of elements on the left side of the array will be equal or less than the number of elements on the right side. Now, let us consider the case of an array with odd number of elements. Array = [9,11,16,7,2] Sorted array = [2,7,9,11,16]. In this case, the median of this array is 9, since it divides the array into two parts: [2,7] and [11,16]. Further, let us consider the case of an array with even elements. Array = [1,2,3,4,5,6]. In such a case, we will take the average between the last element of the left part and the first element of the right part. In this case, the median equals = (3 + 4) / 2 = 3.5.

Input Format

The input should contain 3 lines.

I. First line of the input should contain two integer values which specify the number of elements in array1 and array2.

II. Second line of the input should contain the elements of the first array.

III. Third line of the input should contain the elements of the second array.

Constraints

All elements must be Integers

Output Format

The output should print only the median value.

Sample Input 1

```
5 6
-5 3 6 12 15
-12 -10 -6 -3 4 10
```

Sample Output 1

```
3
```

Sample Input 2

```
4 6
2 3 5 8
10 12 14 16 18 20
```

Sample Output 2

```
11
```

3. Division with Binary Search **CO1**

We can modify binary search algorithm to perform division of two numbers, by defining range [0, infinity] which serves as initial low and high for the binary search algorithm. Now we need to find a mid that satisfies x/y = mid or x*mid for given two numbers x and y. Based on the comparison result based on x and y*mid, we either update low, update high or return mid. 1. If y*mid almost equal to x, we return mid. 2. If y*mid is less than x, we update low to mid 3. If y*mid is more than x, we update high to mid We need to care about division by zero and sign of the result etc. Input: one line of input should contain two numbers separated by space. Output: should print division of the numbers as a result.

Input Format

Input: one line of input should contain two numbers separated by space. Input1: 22 7

Constraints

x,y < infinity

Output Format

Output: should print division of the numbers as a result.

Sample Input 1

22 7

Sample Output 1

3.14286

4. Find the Triplet **CO1**

Given an array of integers, find a triplet having maximum product in the array.

Input: First line of input should specify the number of elements in the array.
Second line of input should specify each element separated by space.

Output: should print Triplets.

Test Cases:

Sample Input 1

5
-4 1 -8 9 6

Sample Input 1

-4 -8 9

Sample Input 2

5
1 7 2 -2 5

Sample Input 2

7 2 5

5. **3n+1** **CO1, CO4**

Consider the following algorithm to generate a sequence of numbers. Start with an integer n. If n is even, divide by 2. If n is odd, multiply by 3 and add 1. Repeat this process with the new value of n, terminating when n = 1. For example, the following sequence of numbers will be generated for n = 22: 22 11 34 17 52 26 13 40 20 10 5 16 8 4 2 1 It is conjectured (but not yet proven) that this algorithm will terminate at n = 1 for every integer n. Still, the conjecture holds for all integers up to at least 1, 000, 000. For an input n, the cycle-length of n is the number of numbers generated up to and including the 1. In the example above, the cycle length of 22 is 16. Given any two numbers i and j, you are to determine the maximum cycle length over all numbers between i and j, including both endpoints. The input will consist of a series of pairs of integers i and j, one pair of integers per line. All integers will be less than 1,000,000 and greater than 0. Output For each pair of input integers i and j, output i, j in the same order in which they appeared in the input and then the maximum cycle length for integers between and including i and j. These three numbers should be separated by one space, with all three numbers on one line and with one line of output for each line of input.

Sample Input	**Sample Output**
1 10	1 10 20
100 200	100 200 125
201 210	201 210 89
900 1000	900 1000 174

6. Matching –String with wild card –pattern. **CO2**

Check the given string is matches with pattern containing wild card characters (‘*’ and ‘?’), where the ‘*’ can match to any number of characters including zero characters and ‘?’ can match to any single character in the given input string. Check if the given input string is matches with given input pattern or not.

Input: Input should contain two lines.

First line of input should contain input string.

Second line of input should contain pattern string.

Output: The output should print either ‘0’ or ‘1’.

‘1’ in the output indicates that the given string is matches with the given pattern.

‘0’ in the output indicates that the given string is not matched with the given pattern.

Sample Input 1:

abcabcccd

a?c*d

Sample Output1

1

Sample Input 2:

abcabcccd

a?c*c

Sample Output2

0

7. **Rotten Oranges** **CO2**

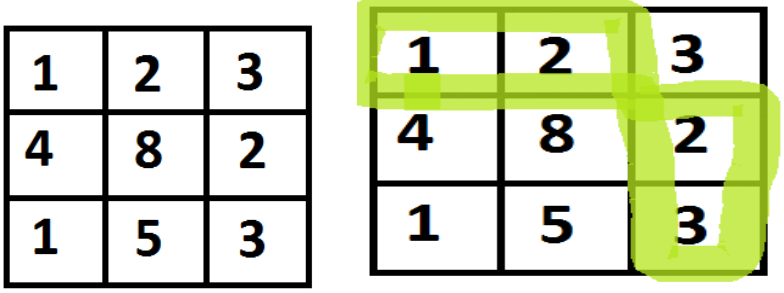
Given a grid of dimension **nxm** where each cell in the grid can have values 0, 1 or 2 which has the following meaning:
0 : Empty cell
1 : Cells have fresh oranges
2 : Cells have rotten oranges
We have to determine what is the minimum time required to rot all oranges. A rotten orange at index [i,j] can rot other fresh orange at indexes [i-1,j], [i+1,j], [i,j-1], [i,j+1]
(**up**, **down**, **left** and **right**) in unit time.

Example 1:

Input: grid = {{0,1,2},{0,1,2},{2,1,1}}
Output: 1
Explanation: The grid is-
0 1 2
0 1 2
2 1 1
Oranges at positions (0,2), (1,2), (2,0)
will rot oranges at (0,1), (1,1), (2,2) and
(2,1) in unit time.

8. Minimum-Cost –Path **CO2**

Find a path in an n×n grid from the upper-left corner to the lower-right corner such that we only move down and right and diagonally lower cells from a given cell, i.e., from a given cell (i, j), cells (i+1, j), (i, j+1) and (i+1, j+1) can be traversed. Assume that all costs are positive integers. Each square contains a number, and the path should be constructed so that the sum of numbers along the path is as small as possible.



The path is (0, 0) –> (0, 1) –> (1, 2) –> (2, 2). The cost of the path is 8 (1 + 2 + 2 + 3)

Sample Input 1

1 2 3
4 8 2
1 5 3

Sample Output 1

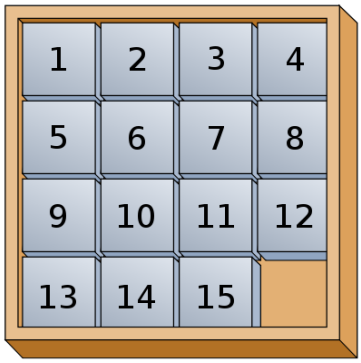
8

9. 15-Puzzle Problem **CO3**

The 15-puzzle is a very popular game: you have certainly seen it even if you don't know it by that name. It is constructed with 15 sliding tiles, each with a different number from 1 to 15, with all tiles packed into a 4 by 4 frame with one tile missing. The object of the puzzle is to arrange the e tiles so that they are ordered as below:

The only legal operation is to exchange the missing tile with one of the 2, 3, or 4 tiles it shares an edge with. Consider the following sequence of moves:

We denote moves by the neighbor of the missing tile is swapped with it. Legal values are "R," "L," "U," and "D" for right, left, up, and down, based on the movements of the hole. Given an initial configuration of a 15-puzzle you must determine a sequence of steps that take you to the final state. Each solvable 15-puzzle input requires at most 45 steps to be solved with our judge solution; you are limited to using at most 50 steps to solve the puzzle.



Input The first line of the input contains an integer n indicating the number of puzzle set inputs. The next 4n lines contain n puzzles at four lines per puzzle. Zero denotes the missing tile.

Output For each input set you must produce one line of output. If the given initial configuration is not solvable, print the line "This puzzle is not solvable." If the puzzle is solvable, then print the move sequence as described above to solve the puzzle.

Sample Input 1

2
2 3 4 0
1 5 7 8
9 6 10 12
13 14 11 15
13 1 2 4
5 0 3 7
9 6 10 12
15 8 11 14

Sample Output 1

LLLDRDRDR

This puzzle is not solvable

10. Tug of War **CO3**

Tug of war is a contest of brute strength, where two teams of people pull in opposite directions on a rope. The team that succeeds in pulling the rope in their direction is declared the winner. A tug of war is being arranged for the office picnic. The picnickers must be fairly divided into two teams. Every person must be on one team or the other, the number of people on the two teams must not differ by more than one, and the total weight of the people on each team should be as nearly equal as possible.

Input

The input begins with a single positive integer on a line by itself indicating the number of test cases following, each described below and followed by a blank line.

The first line of each case contains n, the number of people at the picnic. Each of the next n lines gives the weight of a person at the picnic, where each weight is an integer between 1 and 450. There are at most 100 people at the picnic. Finally, there is a blank line between each two consecutive inputs.

Output For each test case, your output will consist of a single line containing two numbers: the total weight of the people on one team, and the total weight of the people on the other team. If these numbers differ, give the smaller number first. The output of each two consecutive cases will be separated by a blank line.

Sample Input 1

1

3

100

90

200

Sample Output 1

190 200

11. **Find first set bit** **CO3**

Problem Statement : Given an integer an **N.** The task is to return the position of **first set bit found from the right side** in the binary representation of the number.

Note: If there is no set bit in the integer N, then return 0 from the function.

Test case 1

Input: N = 18

Output: 2

Test case 2

Input**: N = 12**

Output**: 3**

Expected Time Complexity: O(log N).

Constraints:

$0 <= N <= 10^8$

12. Euclid Problem **CO4**

From Euclid, it is known that for any positive integers A and B there exist such integers X and Y that AX + BY = D, where D is the greatest common divisor of A and B. The problem is to find the corresponding X, Y , and D for a given A and B.

Input The input will consist of a set of lines with the integer numbers A and B, separated with space (A, B < 1, 000, 000, 001).

Output For each input line the output line should consist of three integers X, Y , and D, separated with space. If there are several such X and Y , you should output that pair for which $X \leq Y$ and $|X| + |Y|$ is minimal.

Sample Input 1

4 6
17 17

Sample Output 1

-1 1 2
0 1 17

13. Coin Problem **CO4**

Given a value V. You have to make change for V cents, given that you have infinite supply of each of C{ C1, C2, .. , Cm} valued coins. Find the minimum number of coins to make the change and print the coins that appear in an optimal solution.

Input:

The first line of input contains an integer T denoting the number of test cases.

The first line of each test case is V and N, V is the value of cents and N is the number of coins.

The second line of each test case contains N input C[i], value of available coins.

Output:

Print the coins appear in an optimal solution and in a newline print the minimum number of coins to make the change and, if not possible print "-1".

Constraints:

$1 \leq T \leq 100$
$1 \leq V \leq 10^6$
$1 \leq N \leq 10^6$
$1 \leq C[i] \leq 10^6$

Sample Input 1

1
7 2
2 1

Sample Input 1

2 2 2 1
4

Explanation :

Testcase 1: We can use coin with value 2 three times, and coin with value 1 one times to change a total of 7.

14. **spirally traversing a matrix** **CO4**

Given a matrix of size r*c. Traverse the matrix in spiral form.

Test case 1:

Input:
r = 4, c = 4
matrix[][] = {{1, 2, 3, 4},
{5, 6, 7, 8},
{9, 10, 11, 12},
{13, 14, 15,16}}
Output:
1 2 3 4 8 12 16 15 14 13 9 5 6 7 11 10

Test case 2:

Input:
r = 3, c = 4
matrix[][] = {{1, 2, 3, 4},
{5, 6, 7, 8},
{9, 10, 11, 12}}
Output:
1 2 3 4 8 12 11 10 9 5 6 7

Text Books:

1. Steven Halim, Felix Halim, "Competitve programming 3", Handbook for ACMICPC and IOI contestants.

Reference Books:

1. Steven S. Skiena Miguel A. Revilla "programming challenges", The Programming Contest Training Manual, Springer.
2. Antti Laaksonen . "Competitive Programmer's Handbook".
3. Ahmed Shamsul Arefin, "Art of Programming Contest" , special online edition for UVA online judge users

Online Resources:

1. https://jadi.net/wp-content/uploads/2017/07/competetive-programmers-handbook.pdf
2. http://acm.cs.buap.mx/downloads/Programming_Challenges.pdf
3. https://www.comp.nus.edu.sg/~stevenha/myteaching/competitive_programming/cp1.pdf
4. https://www.comp.nus.edu.sg/~stevenha/database/Art_of_Programming_Contest_SE_for_uva.pdf

Problem reading & Practice:

https://uva.onlinejudge.org/index.php

Prepared By

Mrs. B. Siva Jyothi, Assistant Prof, Dept of CSE
Mr. S Ratan Kumar, Associate Prof, Dept of CSE
Dr K Prasad, Assistant Prof, Dept of CSE
Mrs. Marline Joys, Assistant Prof, Dept of CSE

QUANTITATIVE APTITUDE – I & SOFT SKILLS

Course Code – Category: CIV 319 - HS | Credits: 1.5

L	T	P	E	O
0	0	3	0	1

Sessional Marks: 100

End Exam: 3 Hours | End Exam Marks: --

Course objectives:

Quantitative Aptitude -I

1. To prepare the students on various principles related to numerical computations.
2. To explain concepts related to numerical estimation.
3. To illustrate and explain the fundamentals related to geometry and mensuration.

Course Outcomes:

Quantitative Aptitude –I

The student will be able to

1. Solve problems related to numerical computations in company specific and other competitive tests.
2. Able to recall and use the concepts to solve problems numerical estimation with respect to company specific and competitive tests.
3. Apply basic principles related to geometry and mensuration & solve questions in company specific and competitive tests.

Mapping of Course Outcomes with Program outcomes:

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

SYLLABUS

Section –A (Quantitative Aptitude –I)

UNIT I 6 Periods

Numerical computation:

Applications based on Numbers, Chain Rule, Ratio Proportion

UNIT II 6 Periods

Numerical estimation - I

Applications Based on Time and work, Time and Distance

UNIT III 4 Periods

Numerical estimation – II

Applications based on Percentages, Profit Loss and Discount, Simple interest and Compound Interest

Partnerships, Shares and dividends

UNIT IV 4 Periods

Data interpretation

Data interpretation related to Averages, Mixtures and allegations, Bar charts, Pie charts, Venn diagrams

UNIT V 4 Periods

Application to industry in Geometry and Mensuration

Books for practice

1. Quantitative Aptitude for Competitive Examinations - Quantitative Aptitude by rs agrawal (English, Aggarwal R. S.)-Published year – Revised Edition 2021 – s chand Publications -
2. A Modern Approach to Verbal & Non-Verbal Reasoning (R.S. Aggarwal) Latest Edition 2018 S Chand Publication January 2018

References

1. Quantitative Aptitude - For Competitive Examinations-U.Mohan Rao-5th Edition - 2017 -SCITECH publications
2. Quantitative Aptitude by Arun Sharma McGrawhill publications-How to prepare for Quantitative Aptitude -for CAT -9th Edition -Publications 2021
3. Data Interpretation and Data Sufficiency By Ananta Ashisha: Data Interpretation-2012 Last Edition

4. Quantitative Aptitude for Competitive Exams by Abhijit Guha -7th Edition – 2020 latest Edition
5. Quantitative Aptitude by Pearson publications-4th Edition -2019 publications
6. Elementary and Higher algebra by HS Hall and SR knight- GK Publications -2016 Edition

Websites:

1. www.m4maths.com
2. www.Indiabix.com
3. 800score
4. Official GRE site
5. Official GMAT site

Anil Neerukonda Institute of Technology & Sciences (Autonomous)

(Affiliated to AU, Approved by AICTE & Accredited by NBA & NAAC with 'A' Grade)
SANGIVALASA-531 162, BheemunipatnamMandal, Visakhapatnam District
Phone: 08933-225083/84/87 Fax: 226395
Website: www.anits.edu.in email: principal@anits.edu.in

III/IV B.TECH (W.e.f. 2019-20 admitted batch)

COURSE OBJECTIVES

- To inculcate effective communication skills with appropriate body language.
- To produce potent leaders, productive team players and effective individuals with proper professional ethics.
- To enable students to make successful oral presentations using relevant content.
- To train students for Group discussions and job Interviews which improves their employ-ability skills.
- To make the students understand the importance of setting realistic goals and achieving them using time management techniques.

COURSE OUTCOMES

At the end of the course, students will be able to:	
1	Comprehend the core engineering subjects using effective verbal and nonverbal communication skills and present spontaneously on the subjects.
2	Present accurate and relevant information efficiently on the given topics, using suitable material aids.
3	Participate in group discussions, present analytical perception on various issues with confidence.
4	Understand the different types of professional interviews and present with confidence the professional interviews.
5	Work effectively in teams, emerge as responsible leaders and able to set time bound goals.

Mapping of course outcomes with program outcomes:

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

JUSTIFICATIONS STATEMEBT FOR CO-PO MAPPING

COS	POS	LEVEL	DESCRIPTION
1	**1**	**1**	Using effective verbal and nonverbal communication skills to Comprehend the core engineering subjects using effective verbal and nonverbal communication skills.
1	**10**	**3**	Work effectively as individuals as well as in teams and emerge as responsible leaders with appropriate professional ethics
1	**12**	**3**	Recognize the need and prepare oneself for lifelong self learning to be abreast with rapidly changing technology Using effective verbal and nonverbal communication skills for lifelong self learning.
2	**10**	**3**	Present accurate and relevant information in engineering practices. Communicate effectively with range of audiences in both oral and written forms through technical papers, seminars, presentations, assignments, project reports etc.
2	**12**	**1**	Apply presentation skills to prepare oneself for lifelong self learning to be abreast with rapidly changing technology Using effective verbal and nonverbal communication skills for lifelong self learning.
3	**8**	**2**	Apply professional ethical practices and citizens with social concern as well as in teams and emerge as responsible leaders with appropriate professional ethics.
3	**9**	**2**	Acquire capacity to understand to various fields of engineering and be able to function effectively as an individual and as a member or leader in a team with appropriate professional ethics.
3	**10**	**3**	Communicate effectively with range of audiences in both oral and written forms through technical papers, seminars, presentations, assignments, project reports etc. using effective verbal and nonverbal communication skills.
4	**2**	**2**	Identify, analyze the problems in different domains and define the requirements appropriate to the solution. With theParticipation in group discussions and interviews using analytical and problem solving abilities, which enhance their employability skills.

4	**6**	**2**	Participate in discussions to solve to assess professional engineering practices.
4	**10**	**3**	Use in discussions to analyse in both oral and written forms of communication.
4	**12**	**2**	Use knowledge of interview skills and confidence in lifelong learning.
5	**9**	**3**	Participate in teams in various fields of engineering and be able to function effectively as an individual and as a member or leader in a team with appropriate professional ethics.
5	**10**	**3**	Set time bound goals and realize them through strategic plans for successful career. Apply computing knowledge to conduct experiments and solve complex problems.
5	**12**	**3**	Set time bound goals and realize them through strategic plans for successful career. Recognize the need and prepare oneself for lifelong self learning to be abreast with rapidly changing technology.

SYLLABUS

UNIT-I: Effective Communication Skills **9 Periods**
CO1

Theory- Formal and informal expressions in business communication, Telephone etiquette, Just a minute (JAM) procedure, Extempore – Tips
LAB:

1. Definition of Communication, Effective Communication
2. Types of Communication
3. Barriers To Communication
4. Non-verbal Communication-Postures and gestures
5. Listening skills- types, analyzing videos and news.
6. Feedback Etiquette
7. Role Plays

Activities- Just a Minute (JAM) practice of different postures and gestures and activity on giving feedback and role plays listening news and reading Newspapers.

UNIT- II: Presentation Skills **9 Periods**
CO2

Theory: Elocution, analogies, YES-NO statements (sticking to a particular line of reasoning (sticking to a particular line of reasoning). Paragraph writing, supplying a suitable beginning/ending/middle sentence to make the paragraphs.

LAB:

1. Content, outline Structure and Packaging
2. Effective presentation strategies
3. Analysis of audience
4. Using audio –visual aids
5. Presenting with confidence
6. Professional grooming and Body language.

Activities. -1.Group presentation by each team- Individual evaluation,
2. Paragraph writing tasks.

UNIT- III: Group Discussions 9 Periods
CO3

Theory: Fact- inference-judgment (to identify statements as FIJ), General essay writing, writing issues and arguments, story writing.

LAB:

1. Introduction- as a part of selection process-guidelines for GD
2. Types of GD
3. Nature of topics of G.D
4. Roles to be played by participants in a GD
5. Evaluation process

Activities–1.Group discussions and Individual evaluation
2. Essay writing assignments.

UNIT- IV: Job- Interviews Skills 9 Periods CO4

Theory: Resume writing-types, statement of purpose', 'letters of recommendation

LAB:

1. Different types of interviews (with emphasis on personal interview), Stress management
2. Self Esteem
3. Goal-Setting- setting- SMART goals
4. Decision making for a given situation
5. Frequently asked questions FAQ's – (preparation for an interview)
6. E mail etiquette-Email format-dos and don'ts.

Activities–1.Mock interviews evaluation,
2. Writing Resume, letters assignments.

UNIT – V: Team Building, Leadership Skills and Attitude 9 Periods
CO5

Theory: Information Transfer-(IELTS) tables, bar diagrams, and pie charts) Book Review Assignments/Contests

LAB:

1. Importance of team work
2. Different stages of team formation
3. Good team vs. effective team, Team player and Team leader
4. Decision making and negotiating skills
5. Types of leadership
6. Importance of time-management- Prioritization, The art of saying 'No',
7. Identifying Time Wasters

Activities – 1.Graphs Interpretation evaluation, Time- Bound and team activities
2. Debates –Teams participation.

REFERENCE BOOKS:

1. Sanjay Kumar and Pushpalata, ***Communication Skills***, Oxford University Press, 2011.
2. Peter Veruki, ***The 250 Job Interview Questions,*** Adams Media Corporation Avon, Massachusetts, 1999.
3. Text Books: 1. Hurlock, E.B **Personality Development**, 28th Reprint. New Delhi: Tata McGraw Hill. 2006.
4. Covey, Stephen R. The 7 Habits of Highly Effective People: Restoring the Character Ethic. 2004.

****

DEEP LEARNING	
CSD 322(A)	**Credits: 3**
Instruction : 3 Periods / Week	Sessional Marks : 40
End Exam : 3 Hours	End Exam Marks : 60

Prerequisites:

- Linear Algebra, Calculus, Statistics, General Programming Concepts

Course Objectives:

- Explore the concepts of neural networks and deep learning
- Examine the usage of neural networks
- Describe the data needs of deep learning
- Apply the working knowledge of neural networks and deep learning
- Explore the parameters for different neural networks

Course Outcomes:

By the end of the course, the student will be able to:	
1.	Interpret the fundamental concepts of neural network and Explore the concepts of feed forward & feedback neural networks
2.	Summarize the concepts of Deep Neural Networks.
3.	Identify the need for convolutional neural network and Examine the Various Architectures of Convolution Neural Network
4.	Examine the Various Architectures of Recurrent Neural Networks
5.	Describe the basic concepts of reinforcement learning and examine the applications of deep Learning

Mapping of course outcomes with program outcomes:

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

SYLLABUS

UNIT-I: **12 Periods**

BASICS OF ARTIFICIAL NEURAL NETWORKS:

Characteristics of Neural Networks: Biological Neural Networks, Comparison of Artificial Neural Network and Biological Neural Network, **Artificial Neural Networks:** Terminology, Models of Neuron, Basic Learning Laws, Applications of ANN, Pros and Cons of ANN, Challenges of ANN, **Activation Functions:** Binary Step Function, Linear Activation, Non Linear Activation Functions, Network Architecture

Feed forward Neural Network: Pattern Classification Network: Perceptron, Linear Inseparability: Hard Problems, Multilayer Feed forward neural network: Generalized Delta Rule-Back propagation learning, MNIST Example using backpropogation.

Learning Outcomes: At the end of this unit, the students will be able to

1. Explore the concept of Neural Network and Apply activation functions for different scenarios.
2. Examine the Single layer feed forward and Multi-Layer Feed forward Neural Network.

UNIT-II: **12 Periods**

Deep neural networks (DNNs): Introduction to Deep Learning, Perspectives and Issues of Deep Learning, Difficulty of training DNNs, Greedy layer wise training, Optimization for training DNNs, Newer optimization methods for neural networks (AdaGrad, RMSProp, Adam), Second order methods for training, Regularization methods (dropout, drop connect, batch normalization)

Learning Outcomes: At the end of this unit, the students will be able to

1. Describe the concepts of Deep Neural Networks
2. Analyze the optimization and regularization methods of neural networks

UNIT-III: **12 Periods**

Convolution Neural Network: From Fully-Connected Layers to Convolutions, Convolutions for Images, Padding and Stride, Multiple Inputs and Outputs Channels, Pooling, Different Deep Convolutional Neural Network Architectures-LeNet, VGG16

Learning Outcomes: At the end of this unit, the students will be able to

1. Describe the basic structure of convolutional neural network.
2. Analyze the Different CNN Architectures

UNIT-IV: **12 Periods**

Recurrent Neural Networks:

Sequence Models, RNN, Back propagation Through Time.

Modern Recurrent Neural Network: GRU, LSTM, Bidirectional RNN, Encoder-Decoder Learning,

Generative Models: Boltzmann Machine, Restricted Boltzmann Machine, Generative Adversarial Network

Learning Outcomes: At the end of this unit, the students will be able to

1. Describe the architecture of recurrent neural network.
2. Explore the concepts of Generative Models.

UNIT-V: **12 Periods**

Reinforcement Learning:

Introduction to Reinforcement Learning, Key Elements of Reinforcement Learning, Deep learning to Reinforcement learning and its applications, Applications of Deep Learning to NLP, Automatic Text Classification. Dialogue Generation, Sentiment Analysis and Opinion Mining using Deep Learning.

Learning Outcomes: At the end of this unit, the students will be able to

1. Describe the role of Deep Reinforcement learning.
2. Explain the applications of Deep learning to NLP

Text Books:

1. Yegnanarayana, B., "Artificial Neural Networks" PHI Learning Pvt. Ltd, 2009.(UNIT-I)
2. Goodfellow, I., Bengio,Y., and Courville, A., "Deep Learning", MIT Press, 2016.(UNIT-II)
3. Aston Zhang, Zachary C. Lipton, Mu Li, and Alexander J. Smola, "Dive into Deep Learning", 2021(UNIT-III,UNIT-IV,UNIT-V)

Reference Books:

1. Satish Kumar, "Neural Networks: A Classroom Approach", Tata McGraw-Hill Education, 2004.
2. Simon Haykin, "Neural Networks-A comprehensive Foundation" Second edition(UNIT-I)
3. Charu C. Aggarwal, "Neural Networks and Deep Learning-A Textbook", Springer, 2018.

Web Resources:

1. https://www.analyticsvidhya.com/blog/2021/10/a-comprehensive-guide-on-deep-learning-optimizers/(UNIT-II)
2. https://www.analyticsvidhya.com/blog/2018/04/fundamentals-deep-learning-regularization-techniques//(UNIT-II)
3. https://www.analyticsvidhya.com/blog/2021/05/a-comprehensive-tutorial-on-deep-learning-part-1//(UNIT-II)
4. https://viso.ai/deep-learning/deep-reinforcement-learning/(UNIT-V)
5. Michael Nielsen, Neural Networks and Deep Learning, http://neuralnetworksanddeeplearning.com/
6. Introduction to Deep Learning, MIT 6.S191 Alexander Amini January 28, 2019, http://introtodeeplearning.com/

Prepared By

Mr. S. Joshua Johnson, Assistant Prof, Dept of CSE
Dr. P.E.S.N. Krishna Prasad, Prof, Dept of CSE
Dr. K. S. Deepthi, Associate Prof, Dept of CSE
Dr. K. Prasad, Assistant Prof, Dept of CSE

MOBILE COMPUTING	
CSD 322(B)	**Credits : 3**
Instruction : 3 Periods /Week	**Sessional Marks : 40**
End Exam : 3 Hours	**End Exam Marks : 60**

Pre Requisite:

- **Data Communication, Computer Network**

Course Objective:

- To make the student understand the concept of mobile computing terminology and basic services.
- To understand the wireless protocols.
- To understand various routing mechanisms.

Course Outcomes:

By the end of the course, the student will be able to:	
1.	The student will be able to interpret GSM architecture and its services.
2.	The student will be able to analyze the various wireless application protocols and its different concepts for various mobile applications.
3.	The student will learn with the representation of mobile network layer protocols and its functionalities .
4.	The student will be able to understand, analyze & develop any existing or new models of mobile environments for 3G networks.
5.	The student will be able to understand, evaluate and create the platforms, protocols and related concepts along with along with mobile in mobile environment.

CO-PO MAPPING:

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

SYLLABUS

UNIT-1 **12 Periods**

Introduction: Mobile Communications, Mobile Computing – Paradigm, Promises/Novel Applications and Impediments and Architecture; Mobile and Handheld Devices, Limitations of Mobile and Handheld Devices.

Global System for Mobile communication(GSM): Services, System Architecture, Radio Interfaces, Protocols, Localization, Calling, Handover, New Data Services, GPRS Architecture, GPRS Network Nodes.

Learning Outcomes: At the end of this unit, the students will be able to

1. Explain mobile communication architecture with various characteristics.
2. Explain about GSM architecture

UNIT-2 **12 Periods**

Medium Access Control (MAC) : Motivation for a specialized MAC (Hidden and exposed terminals, Near and far terminals), Wireless LAN/(IEEE 802.11) architecture, key IEEE802.11 a/b/c/d/e/g/i/n/T/ac/ standards.

Wireless Application Protocol (WAP): The Mobile Internet standard, WAP Gateway and Protocols, wireless mark up Languages (WML). Wireless Local Loop(WLL): Introduction to WLL Architecture, wireless Local Loop Technologies.

Learning Outcomes: At the end of this unit, the students will be able to

1. Explain the importance of MAC and provide architecture details.
2. Compare details among WAP, WLL with WML.

UNIT-3 **12 Periods**

Mobile Network Layer : IP and Mobile IP Network Layers, Packet Delivery and Handover Management, Location Management, Registration, Tunneling and Encapsulation, Route Optimization using Soft computing techniques – ANT Bee colony, Support Vector Machine, Particle Swarm Optimization and Genetic Algorithm.

Learning Outcomes: At the end of this unit, the students will be able to

1. Compare between IP layer with Mobile IP network wrt packet delivery and hand-off management.
2. Explain various Optimization techniques.

UNIT-4 **12 Periods**

Mobile Transport Layer : Conventional TCP/IP Protocols, Indirect TCP, Snooping TCP, Mobile TCP.

Third Generation (3G) Mobile Services: Introduction to International Mobile Telecommunications 2000 (IMT 2000) vision, Wideband Code Division Multiple Access (W-CDMA) and CDMA 2000, Quality of services in 3G.

Learning Outcomes: At the end of this unit, the students will be able to

1. Compare between conventional TCP/IP protocol with Mobile TCP.
2. Explain about various Quality of services in 3G.

UNIT-5 **12 Periods**

Mobile Ad hoc Networks (MANETs) : Introduction, Applications & Challenges of a MANET, Routing, Classification of Routing Algorithms, Algorithms such as DSR, AODV, DSDV, etc. , Mobile Agents, Service Discovery ,case study using NS2 –traffic analysis using CBR and VBR
Wireless Enterprise Networks: Introduction to Virtual Networks, Blue tooth technology, Blue tooth Protocols.
Learning Outcomes: At the end of this unit, the students will be able to

1. Compare DSR algorithm with AODV algorithm.
2. Explain about virtual network and blue tooth technology.

Text books:

1. Jochen Schiller, "Mobile Communications", Addison-Wesley, Second Edition, 2009.
2. Raj Kamal, "Mobile Computing", Oxford University Press, 2007, ISBN: 0195686772

Reference books:

1. Asoke K Talukder, Hasan Ahmed, Roopa R Yavagal, "Mobile Computing, Technology Applications and Service Creation" Second Edition, Mc Graw Hill.
2. Martin Sauter, "From GSM to LTE-Advanced: An Introduction to Mobile Networks and Mobile Broadband," Second Edition, Wiley.

Web Resource:

1. https://onlinecourses.nptel.ac.in/noc18_cs09/preview
2. http://studentnet.cs.manchester.ac.uk/pgt/2014/WelcomeWeek/slides/MobCompIntro2013-NPF.pdf

---

Prepared By
Dr. G. Jagadish, Assistant Prof, Dept of CSE

DIGITAL IMAGE PROCESSING	
CSD 322(C)	**Credits : 3**
Instruction : 3Hours /Week	Sessional Marks : 40
End Exam : 3 Hours	Ena Exam Marks : 60

Prerequisites:

- Knowledge of linear algebra, basic probability and statistics, introductory knowledge of basic programming language, MATLAB/C are preferred.

Course Objectives:

- To make the students to be familiar with basic image processing techniques for solving real problems,
- To make the students have a general overview on digital image processing concept along with its uses and applications.
- To make the students gain knowledge about representation of a digital image in different domains and the transformations between those domains,
- To make the students learn about various morphological operations on a digital image.

Course Outcomes:

By the end of the course, the student will be able to:	
1.	Apply the basic concepts of 2D image acquisition, sampling, quantization, relationships between pixels and components of image.
2.	Analyze the filtering techniques in spatial domain for face reorganization, pattern reorganization and segmentation.
3.	Analyze and apply the filtering techniques in frequency domain for classify the images.
4.	Apply image morphological techniques for manipulating digital images.
5.	Apply the image Segmentation techniques on Edge detection and Region-Based Segmentation.

Mapping of course outcomes with program outcomes:

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

SYLLABUS

UNIT-I : **12 Hours**

Introduction :

Digital Image Processing and Applications – Image Representation and Modeling

Digital Image Fundamentals:

Elements of Visual perception – A simple Image Model – Sampling and Quantization – Some Basic Relationship between Pixels.

Elements of digital image processing systems, Digital Camera working principles, Elements of visual perception, brightness, contrast, hue, saturation.

Learning outcomes:

1. Review the fundamental concepts of a digital image processing system.
2. Describe and explain basic principles of digital image processing

UNIT-II : **15 Hours**

Image Transforms & **Color Image Processing:**

Background, Some Basic Intensity Transformation Functions, Histogram Processing, Histogram Equalization, Histogram Matching (Specification), Local Histogram Processing, Using Histogram Statistics for Image Enhancement, Color Fundamentals , Color Models , The RGB Color Model , The CMYand CMYK Color Models , The HSI Color Model , Pseudocolor Image Processing , Intensity Slicing , Intensity to Color Transformations , Basics of Full-Color Image Processing.

Learning outcomes:

1. Examine various types of images.
2. Analyze intensity transformations and color images.

UNIT- III : **8 Hours**

IMAGE ENHANCEMENT IN SPATIAL DOMAIN:

Fundamentals of Spatial Filtering, Smoothing Spatial Filters, Sharpening Spatial Filters, Combining Spatial Enhancement Methods.

Learning outcomes:

1. Analyze images in the frequency domain using various transforms.
2. Develop Fourier transform for image processing in frequency domain.

UNIT-IV : **12 Hours**

IMAGE ENHANCEMENT IN FREQUENCY DOMAIN:

Background, Preliminary Concepts, Sampling and the Fourier Transform of Sampled Functions, The Basics of Filtering in the Frequency Domain, Image Smoothing Using Frequency Domain Filters, Image Sharpening Using Frequency Domain Filters, Selective Filtering, Wavelets.

Learning outcomes:

1. Evaluate the techniques for image enhancement and image restoration.
2. Apply image processing algorithms in practical applications.

UNIT-V : **15 Hours**

IMAGE SEGMENTATION & Morphology:

Edge detection, Edge linking via Hough transform – Thresholding - Region based segmentation – Region Growing – Region splitting and Merging .

Preliminaries, Erosion and Dilation, Opening and Closing, the Hit-or-Miss Transformation, Some Basic Morphological Algorithms, Gray-Scale Morphology

Learning outcomes:

1. Interpret image segmentation and representation techniques.
2. Evaluate the methodologies for image segmentation, restoration etc.

Text Books:

1. Gonzalez Rafael C and Woods Richard E," *Digital Image Processing*", 3rd Edition, Prentice Hall, 2008.
2. Jain Anil K," *Fundamentals of Digital Image Processing"*, PrenticeHall, 1989. (TA1632.J25)

Reference Books:

1. Pratt William K, "*Digital Image Processing: PIKS Scientific Inside"*, 4th Edition, John Wiley, 2007. (TA1632.P917 2007)
2. Pitas Ioannis, Digital Image Processing Algorithms and Applications, John Wiley, 2000. (TA1637.P681)
3. Anil K. Jain, PHI. Pattern Recognition and Image Analysis, Earl Gose and Richard Johnsonbaugh Steve Jost, PHI," *Fundamentals of Digital Image Processing"*.

Web Resources:

1. **https://nptel.ac.in/courses/106/105/106105032/**
2. **https://www.coursera.org/courses?languages=en&query=digital%20image%20processing**

Prepared By
Mrs. S. A. Bhavani , Assistant Prof, Dept of CSE

DATA WAREHOUSING AND DATA MINING	
CSD 312(D)	**Credits: 3**
Instruction : 3 Periods / Week	Sessional Marks : 40
End Exam : 3 Hours	End Exam Marks : 60

Prerequisites:

- Database management systems concepts
- Probability and statistics

Course Objectives:

- To know the data storage in data warehousing.
- To handle real world data to pre-processing
- The importance of data analysis through data mining.

Course Outcomes:

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

1.	Identify the challenging issues in Data Mining data warehousing.
2.	Apply the data pre-processing approaches in the data mining.
3.	Analyse association rule mining in various dimensional databases.
4.	Apply classification by using decision tree induction, Bayesian, back propagation and prediction methods for data analysis.
5.	Apply various clustering techniques

CO-PO Mapping:

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

SYLLABUS

UNIT-1 **14 periods**

Data warehouse: Introduction to Data warehouse, Difference between operational database systems and Data warehouses. Differences between operational databases systems and data warehouses. **Multidimensional data model**: From Tables and spreadsheets to Data Cubes, stars, Snowflakes, and Fact Constellations schemas for Multidimensional databases. Examples for defining star, snowflake and fact constellation schemas. **Data Warehouse Architecture:** Steps for the design and construction of data warehouses. A three-tier data warehouse architecture. **From Data warehousing to data mining**: Data warehouse usage, from on-line analytical processing to online analytical mining.

Learning outcome: at the end of this unit student are able to

1. Discuss the data storage in the data warehouse and multidimensional model
2. Explain the architecture of data warehouse

UNIT-2 **14 periods**

Data Mining Introduction: Data mining-on what kind of data, Relational databases, data warehouses, transactional databases, advanced database systems and advanced database applications. Data mining functionalities, classification of data mining systems, Major issues in data mining.

Data Pre-processing: Data cleaning: Missing values, Noisy data, inconsistent Data, Data Integration and Transformation: Data Integration, Data transformation Data Reduction: Data cube aggregation, dimensionality reduction, data compression, Numerosity reduction.

Learning outcome: at the end of this unit student are able to

1. Motivate the importance of data mining, its functionalities and issues
2. Find the importance of data pre-processing methodologies

UNIT-3 **12 periods**

Association Rule mining in Large Databases: Association rule mining , mining single-dimensional Boolean association rules from transaction databases, Mining multilevel association rules from transaction databases. Mining multidimensional association rules from relational databases. From association mining to correlation analysis. Constraint based association mining.

Learning outcome: at the end of this unit student are able to

1. Generate association rules to given data and analyse market basket analysis
2. Analyze association mining to correlation analysis.

UNIT-4 **12 Periods**

Classification and Prediction: Issues regarding classification and prediction, Classification by decision tree induction, Bayesian classification, Classification by back propagation, Prediction, classification accuracy.

Learning outcome: at the end of this unit student are able to

1. Relate the importance of classification and prediction in data analysis.
2. Apply classification and regression techniques in real time data for analysis

UNIT-5 **10 Periods**

Cluster Analysis: Types of data in cluster analysis, a categorization of major clustering methods, Partition based methods: K-means, K-medoids. Hierarchical methods: BIRCH, CURE Density-based methods: DBSCAN.

Learning outcome: at the end of this unit student are able to

1. Outline various types of clustering methods
2. Apply and analyze various types of clustering methods to real time data set.

Text Book:

1. Jiawei Han and Micheline Kamber Morgan, "*Data Mining Concepts and Techniques*", Second Edition**,** Kaufman Publications.

Reference Books:

1. Adriaan, "*Introduction to Data Mining*" , Second Edition, Addison Wesley Publication.
2. A.K.Pujari, "*Data Mining Techniques*", University Press.
3. Dr.M.Ramarkrishna Murty, "*Introduction to data mining and soft computing*", First Edition, University Science Press.

E-Resources :

https://www.coursera.org/specializations/data-mining

https://www.udemy.com/course/data-mining/

Prepared By
Mrs. G. V. Gayathri, Assistant Prof, Dept of CSE
Mr. R. Sivakiran, Assistant Prof, Dept of CSE
Mrs. P. Spandhana , Assistant Prof, Dept of CSE
Mrs. V. Mamatha, Assistant Prof, Dept of CSE

DISTRIBUTED OPERATING SYSTEMS	
CSD 323(A)	**Credits : 3**
Instruction **:** 3 Periods /Week	Sessional Marks : 40
End Exam : 3 Hours	End Exam Marks : 60

Prerequisites:

- Fundamentals of Operating Systems and Computer networks and protocols.

Course Objectives:

- To provide an overview of the concepts of distributed operating systems and challenges that includes Architecture and Fundamental Models.
- To explore about various types of communication procedures and protocols in a distributed operating systems environment.
- To interpret the concept of communication between distributed objects and remote procedural calls.
- To analyse and understand the concepts of Distributed File system.
- To demonstrate the idea of Transactions and Replications in distributed operating system.

Course Outcomes:

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

1.	Analyze the system model, software layers of distributed operating systems and its challenges.
2.	Examine the inter-process communication, TCP stream communication procedures and protocols.
3.	Evaluate the concepts of Remote procedural calls and communication among objects in distributed operating system.
4.	Apply the knowledge of peer-to-peer system, distributed mutual exclusion of distributed file system in real world scenario.
5.	Apply concurrency control, deadlock management techniques in distributed operating system for group communication.

CO-PO Mapping:

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

SYLLABUS

UNIT-I: **12 periods**

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.

Learning Outcomes: At the end of this unit, Students are able to

1. Explore the concepts of Distributed Systems.
2. Analyze the Various System models in distributed systems.

UNIT-II: **12 periods**

Inter process Communication: Introduction, The API for the Internet Protocols- The Characteristics of Inter process communication, Sockets, UDP Datagram Communication, TCP Stream Communication; External Data Representation and Marshalling; Client Server Communication; Group Communication- IP Multicast- an implementation of group communication, Reliability and Ordering of Multicast.

Learning Outcomes: At the end of this unit, Students are able to

1. Analyse the concept of Inter process communication.
2. Compare the UDP datagram communication and TCP stream communication.

UNIT-III: **10 periods**

Distributed Objects and Remote Invocation: Introduction, Communication between Distributed Objects- Object Model, Distributed Object Modal, Design Issues for RMI, Implementation of RMI, Distributed Garbage Collection; Remote Procedure Call, Events and Notifications, Case Study: JAVA RMI

Learning Outcomes: At the end of this unit, Students are able to

1. Determine the design issues and implementation of RMI.
2. Summarize the concept of distributed garbage collection and remote procedure call.

UNIT-IV: **12 periods**

Distributed File Systems: Introduction, File Service Architecture; Peer-to-Peer Systems: Introduction, Napster and its Legacy, Peer-to-Peer Middleware, Routing Overlays. Coordination and Agreement: Introduction, Distributed Mutual Exclusion, Elections, Multicast Communication.

Learning Outcomes: At the end of this unit, Students are able to

1. Demonstrate the architecture of file systems and build knowledge on peer-to-peer systems.
2. Identify the various algorithms of Distributed mutual exclusion

UNIT-V: **12 periods**

Transactions & Replications: Introduction, System Model and Group Communication, Concurrency Control in Distributed Transactions, Distributed Dead Locks, Transaction Recovery; Replication-Introduction, Passive (Primary) Replication, Active Replication.

Learning Outcomes: At the end of this unit, Students are able to

1. Determine the concurrency control in distributed transactions.
2. Explore the concept of active and passive replication

Text Books:

1. Ajay D Kshemkalyani, Mukesh Sighal, "Distributed Computing, Principles, Algorithms and Systems", 1st Edition, 2008,Cambridge.
2. George Coulouris, Jean Dollimore, Tim Kindberg, "Distributed Systems- Concepts and Design", Fourth Edition, 2005,Pearson Publication.

Reference Books :

1. Makes Singhal and Niranjan G.Shivaratna ,"Advanced Concepts in Operating Systems" , 1st Edition,1994,Tata McGraw Hill Edition.

Web Resources:

1. https://www.coursera.org/learn/distributed-programming-in-java
2. https://www.edx.org/course/javacheng-xu-she-ji-java-programming-pekingx-04830340x
3. https://www.coursera.org/courses?languages=en&query=java

Prepared By:
Mr.Ch.Seshadri Rao, Assistant Prof, Dept of CSE
Mr.M. Kranthi Kiran, Assistant Prof, Dept of CSE
Mrs.Pritee Parwekar, Assistant Prof, Dept of CSE

EMBEDDED SYSTEMS	
CSE323(B)	Credits : 3
Instruction : 3 Periods /Week	Sessional Marks : 40
End Exam : 3 Hours	End Exam Marks : 60

Prerequisites:

- Basic Knowledge of Programming Fundamentals
- Knowledge of Programming Languages (such as C, C++)

Course Objectives:

- To provide in-depth knowledge about embedded processor, its hardware and software
- To explain programming concepts and embedded programming in C and assembly language
- To explain real time operating systems, inter-task communication and an embedded software development tool

Course Outcomes:

By the end of the course, the student will be able to:	
1.	Identify the basic components required to build an embedded system.
2.	Select an appropriate software architecture to build an embedded system.
3.	Design embedded software-using RTOS.
4.	Build embedded software using different software tools.
5.	Debug embedded software using different software and hardware tools.

Mapping of Course Outcomes with Program Outcomes:

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

SYLLABUS

UNIT I **(8 Periods)**

Introduction to embedded systems hardware needs; timing diagrams, memories (RAM, ROM, EPROM). Tristate devices, Buses, PLD's. Built-ins on the microprocessor. Interrupts basics, ISR; Context saving, shared data problem. . Atomic and critical section.

Learning Outcomes: At the end of this unit, Students are able to

1. Describe different types of memories used in Embedded System Development
2. Interpret the role of Interrupts in Embedded Systems Performance.

UNIT II **(8 Periods)**

Survey of software architectures, Round Robin, Function queue scheduling architecture, Use of real time operating system .RTOS, Tasks , Scheduler, Shared data reentrancy, priority inversion, mutex binary semaphore and counting semaphore.

Learning Outcomes: At the end of this unit, Students are able to

1. Describe the role of software architecture in Embedded Systems Performance
2. Explain the application of the Real Time Operating System in Embedded System Development.

UNIT III **(10 Periods)**

Inter task communication, message queue, mailboxes and pipes, timer functions, events. Interrupt routines in an RTOS environment. Embedded system software design using an RTOS. Hard real time and soft real time system principles, Task division, need of interrupt routines, Interrupt latency, Introduction to Device Drivers.

Learning Outcomes: At the end of this unit, Students are able to

1. Describe the different RTOS features used in Embedded Systems Development.
2. Differentiate Hard real time and Soft real time systems

UNIT IV **(9 Periods)**

Embedded Software development tools. Host and target systems, cross compilers, linkers, locators for embedded systems. Getting embedded software into the target system.

Learning Outcomes: At the end of this unit, Students are able to

1. Differentiate host and target systems
2. Describe the need of Cross Compilers, linkers and locators for embedded system development.

UNIT V **(10 Periods)**

Debugging techniques. Testing on host machine ,Instruction set emulators, logic analyzers. In-circuit emulators and monitors.

Case Study

DEVELOPING EMBEDDED C APPLICATIONS THROUGH KEIL SOFTWARE, Embedded PROGRAMMING IN C++, java.

Learning Outcomes: At the end of this unit, Students are able to

1. Explain different types of debugging techniques used in Embedded System Development.
2. Describe the difference between simulation and emulation.

Text Books:

1. David A. Simon, An Embedded Software Primer, Pearson Education, Inc., 1999
2. Raj Kamal, Embedded Systems, Architecture, Programming and Design, TMH, 2003

Reference Books:

1. Sriram V Iyer and Pankaj Gupta, Embedded Real Time Systems programming, TMH, 2004.

Web Resources:

1. https://www.coursera.org/learn/introduction-embedded-systems#about

---

Prepared By
Mr. S. Ratan Kumar, Associate Prof, Dept of CSE
Mr.B.RaviKiran, Assistant Prof, Dept of CSE

HUMAN COMPUTER INTERACTION	
CSD323(C)	**Credits : 3**
Instruction **:** 3 Periods /Week	Sessional Marks : 40
End Exam : 3 Hours	End Exam Marks : 60

Prerequisites:
Student should have an idea about User Interface Design and Programming

Course Objectives:
The main objective is to get students to think constructively and analytically about how to design and evaluate interactive technologies.

Course Outcomes:

By the end of the course, the student will be able to:	
1.	Identify the capabilities of both humans and computers from the viewpoint of human information processing and describe typical HCI models and various paradigms.
2.	Apply an interactive design process and universal design principles to design HCI systems.
3.	Describe and use HCI design principles, standards and guidelines.
4.	Analyze and identify user models, user support, socio-organizational issues, and stakeholder requirements of HCI systems.
5.	Demonstrate searching methodologies, filtering concepts in databases and text documents.

Mapping of Course Outcomes with Program Outcomes:

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

SYLLABUS

UNIT I: **13 Periods**

Introduction: Usability of Interactive Systems- introduction, usability goals and measures, usability motivations, universal usability, goals for our profession.
Managing Design Processes: Introduction, Organizational design to support usability, Four pillars of design, development methodologies, Ethnographic observation, Participatory design, Scenario Development, Social impact statement for early design review, legal issues, Usability Testing and Laboratories.
Learning Outcomes: By the end of the course, the student will be able to:

1. Explain the capabilities of both humans and computers from the viewpoint of human information processing
2. Recognise typical HCI models and various paradigms

UNIT II: **16 Periods**

Menu Selection, Form Fill-In and Dialog Boxes: Introduction, Task- Related Menu Organization, Single menus, Combinations of Multiple Menus, Content Organization, Fast Movement Through Menus, Data entry with Menus: Form Fill-in, dialog Boxes, and alternatives, Audio Menus and menus for Small Displays.
Command and Natural Languages: Introduction, Command organization Functionality, Strategies and Structure, Naming and Abbreviations, Natural Language in Computing
Interaction Devices: Introduction, Keyboards and Keypads, Pointing Devices, Speech and Auditory Interfaces, Displays- Small and large.
Learning Outcomes: By the end of the course, the student will be able to:

1. Use an interactive design process and universal design principles to design HCI systems
2. Demonstrate universal design principles to design HCI systems

UNIT III: **15 Periods**

Quality of Service: Introduction, Models of Response-Time impacts, Expectations and attitudes, User Productivity, Variability in Response Time, Frustrating Experiences
Balancing Function and Fashion: Introduction, Error Messages, Non anthropomorphic Design, Display Design, Web Page Design, Window Design, Color.
Learning Outcomes: By the end of the course, the student will be able to:

1. Describe quality of service standards and guidelines.
2. Explain HCI design principles

UNIT IV: **8 Periods**

User Documentation and Online Help: Introduction, Online Vs Paper Documentation, Reading from paper Vs from Displays, Shaping the content of the Documentation, Accessing the Documentation, Online tutorials and animated documentation, Online communities for User Assistance, The Development Process.

Learning Outcomes: By the end of the course, the student will be able to:

1. Identify user models, user support, socio-organizational issues
2. Explain stakeholder requirements of HCI systems.

UNIT V: **8 Periods**

Information Search: Introduction, Searching in Textual Documents and Database Querying, Multimedia Document Searches, Advanced Filtering and Searching Interfaces Information Visualization: Introduction, Data Type by Task Taxonomy, Challenges for Information Visualization

Learning Outcomes: By the end of the course, the student will be able to:

1. Discuss searching methodologies in databases and text documents.
2. Discuss filtering concepts in databases and text documents.

Text Books:

1. Ben Shneiderman, Catherine Plaisant, Maxine Cohen, Steven M Jacobs, "Designing the User Interface, Strategies for Effective Human Computer Interaction", 5 Edition, Pearson
2. Wilbert O Galitz, Wiley DreamaTech, "The Essential guide to user interface design", 2 Edition, Wiley Publication.

Reference Books :

1. Dan R.Olsan, "Human Computer, Interaction", Cengage Publication, 2010.
2. Ben Shneidermann, "Designing the user interface", 4 Edition, Catherine Plaisant.2004
3. Soren Lauesen, "User Interface Design", Addison Wesley Paperback, 2005

Web Resources:

- https://onlinecourses.nptel.ac.in/noc18_cs23/preview

Prepared By:
Mrs R Sivaranjini, Professor,HOD, CSE Dept
Mrs G V Gayathri, Asst Professor,CSE Dept
Mr. K Chandra Sekhar, Asst Professor,CSE Dept
Ms.N.Lokeswari, Asst Professor,CSE Dept

PATTERN RECOGNITION	
CSD 323(D)	**Credits: 3**
Instruction : 3 Periods / Week	Sessional Marks : 40
End Exam : 3 Hours	End Exam Marks : 60

Prerequisites:

- Number Theory and Linear Algebra, Data Structures, Artificial Intelligence**.**

Course Objectives:

- To analyze and apply the foundations of pattern recognition, various classification and clustering techniques, ensemble classifiers used in Pattern Recognitions

Course Outcomes:

By the end of the course, the student will be able to:	
1.	Discuss the basic foundations of Pattern Recognition, mathematical concepts
2.	Discuss Bayesian decision theory and parameter estimation methods
3.	Explain unsupervised learning and hidden markov models
4.	Illustrate dimensionality reduction and discriminant functions
5.	Explain non-metric methods and ensemble classifiers

Mapping of course outcomes with program outcomes:

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

SYLLABUS

Unit 1: Basics of Probability, Random Processes and Linear Algebra: 12 Periods

Introduction to pattern Recognition, Feature Detection, Classification,
Probability: independence of events, conditional and joint probability, Bayes' theorem; Random Processes: Stationary and nonstationary processes, Expectation, Autocorrelation, Cross-Correlation, spectra; Linear Algebra: Inner product, outer product, inverses, eigen values, eigen vector.

Learning Outcomes:

- To discuss foundations of Pattern Recognition.
- To apply various estimation metrics, functions and operations used in pattern recognition.

Unit 2: Bayes Decision Theory and Parameter Estimation Methods: 12 Periods

Bayes Decision Theory: Minimum-error-rate classification, Classifiers, Discriminant functions, Decision surfaces, Normal density and discriminant functions, discrete features
Parameter Estimation Methods: Maximum-Likelihood estimation - Gaussian case, Maximum a Posteriori estimation, Bayesian estimation - Gaussian case.

Learning Outcomes:

1. To discus and apply Bayesian decision theory and its models for classification.
2. To Explain different parameter estimation methods.

Unit 3: Unsupervised Learning and Hidden Markov Models: 12 Periods

Unsupervised learning - Criterion functions for clustering. Algorithms for clustering: K-Means, Hierarchical and other methods. Cluster validation. Gaussian mixture models, Expectation-Maximization method for parameter estimation, Maximum entropy estimation, Sequential Pattern Recognition.
Hidden Markov Models (HMMs): Introduction, Discrete HMMs. Continuous HMMs, Nonparametric techniques for density estimation, Parzen-window method, K-Nearest Neighbour method

Learning Outcomes:

1. To Explain unsupervised learning methods in Pattern Recognition.
2. To illustrate basic concepts of Hidden Markov models and its importance in PR.

Unit 4: Dimensionality Reduction and Discriminant Functions: 12 Periods

Dimensionality Reduction: Principal component analysis - it relationship to eigen analysis, Fisher discriminant analysis - Generalized Eigen analysis, Eigen vectors/Singular vectors as dictionaries, Factor Analysis, Total variability space - a dictionary learning methods, Non negative matrix factorization - a dictionary learning method.

Linear discriminant functions: Gradient descent procedures, Perceptrons, Multi-Layer Perceptrons, Support Vector Machines.

Learning Outcomes:

1. To apply various dimensionality reduction techniques.
2. To apply linear discriminant functions and methods in Pattern Recognition.

Unit 5: Non-Metric Methods for Pattern Recognition: 12 Periods

Non-metric methods for pattern classification: Non-numeric data or nominal data, Decision trees, CART, ID3, C4.5, Random Forests, Genetic Algorithms, Reinforcement Learning with Human Interaction, Ensemble Classifiers –"Committee Machines ", Applications of Pattern Recognition.

Learning Outcomes:

1. To discuss non-metric methods and various decision tree methods in Pattern Recognition
2. To apply various decision tree and ensemble classifiers in Pattern Recognition

Textbooks:

1. Duda, Richard O., Peter E. Hart, and David G. Stork. *Pattern Classification*, John Wiley & Sons, 2001.
2. S.Theodoridis and K.Koutroumbas, Pattern Recognition, 4th Ed., Academic Press, 2009
3. M.Bishop, Pattern Recognition and Machine Learning, Springer, 2007

Reference Books:

1. Earl Gose, Richard Johnsonbaugh and Steve Jost, Pattern Recognition and Image Analysis", Prentice Hall of India, 2002.
2. Bishop, C. M. Neural Networks for Pattern Recognition. Oxford University Press. 1995.
3. Hastie, T., Tibshirani, R. and Friedman, J. The Elements of Statistical Learning. Springer. 2001.
4. Koller, D. and Friedman, N. Probabilistic Graphical Models. MIT Press. 2009.
5. T. Hastie et al., The Elements of Statistical Learning, Springer-Verlag, 2001.
6. K. Murphy, Machine Learning: A probabilistic Perspective, MIT Press, 2012.

Online Resources:

1. https://lme.tf.fau.de/teaching/free-deep-learning-resources/
2. https://lme.tf.fau.de/teaching/free-deep-learning-resources/

Prepared By

Dr. P.E.S.N. Krishna Prasad, Prof, Dept of CSE

Dr. K.Prasad, Assistant Prof, Dept of CSE

OBJECT ORIENTED SOFTWARE ENGINEERING	
CSD 324	**Credits : 3**
Instruction: 3 Periods /Week	Sessional Marks: 40
End Exam: 3 Hours	End Exam Marks: 60

Prerequisites:

- Basic Mathematical Knowledge.
- Basic knowledge on procedural and object oriented programming.
- Basic knowledge on problem solving.

Course Objectives:

The course should enable the students:

- To Explain the importance of OOSE in Software development.
- To Provide knowledge on software Life Cycle and Development Models with object oriented concepts.
- To Explain the role of UML and Testing in Software Development.
- To Plan and manage the development of software projects.

Course Outcomes:

By the end of the course, the student will be able to:	
1	Analyze the different software process models and their significance.
2	Interpret the functional, non–functional requirements and requirement Engineering Process.
3	Choose the Architecture for a given software application & Design UML diagrams.
4	Demonstrate skills in Object-oriented Modelling and Plan software project management activities.
5	Identify the Testing Strategies and design test suits for the given Test Scenarios.

Mapping of Course Outcomes with Program Outcomes:

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

SYLLABUS

UNIT-I **(10 Hours)**

Importance of Software Engineering: Problem Solving Activity, Modelling Activity, Knowledge Acquisition activity, Rationale Driven Activity, Umbrella Activities , Software Engineering Challenges ,Software Development Life Cycle.
Process Models: Waterfall Model, Incremental Model, Prototype Model, V Model, Spiral Model, Rapid Model, Agile Model.
Learning Outcomes: At the end of this unit, Students are able to

1. List Various Software Development activities.
2. Classify the Working of Software Process Models.

UNIT-II **(14 hours)**

Software Requirement Analysis & Specification: Need for SRS, Characteristics of Requirements, Functional Requirements, Non Functional Requirements, Requirement Specification Document -IEEE Format.
Function Oriented Design: Structured Design Methodology –DFD Diagram, Design Principles, Module-Level Concepts- Cohesion & Coupling.
Learning Outcomes: At the end of this unit, Students are able to

1. Find Functional and Non-functional Requirements and prepare SRS Document..
2. Design Data flow in Functional Approach (DFD).

UNIT-III **(18 Hours)**

Architectural Design: Role of Software Architecture, Architecture Views, Architectural Styles for C & C View- Pipe and Filter, Client and Server, Shared Data.
Object Oriented Design: OO Concepts, Importance of Modelling.
Overview of UML: Building Blocks of UML (Things, Relationships, Diagrams), Class Diagram, Class Relationships, Design Axioms, Corollaries, Use Case- Use Case Scenario, Use Case Diagram, Relationships among Use Cases.
Interaction Diagrams: Sequence Diagram, Collaboration Diagram.
Learning Outcomes: At the end of this unit, Students are able to

1. Make use of Various Software Architectures.
2. Design UML Diagrams with Object Oriented Approach.

UNIT – IV **(15 Hours)**

UML Diagrams: Activity Diagram, State Chart Diagram, Component Diagram & Deployment Diagram.
Object Oriented Methodologies: Unified Methodology, Rambaugh Methodology, Booch Methodology.
Software Project Management: Definition of Software Project, Need of Software Project Management, Software Project management Plan, Effort Estimation ,Case Study on Effort

Estimation with COCOMO, Schedule - Overall and detailed scheduling, Risk Management Plan, Project Tracking & Control.

Learning Outcomes: At the end of this unit, Students are able to

1. Illustrate Various object oriented Methodologies.
2. Adapt the approaches in Software Project Management to Develop the Software Project.

UNIT-V **(10 Hours)**

Testing: Testing Fundamentals, Testing Process, Levels of Testing ,Test Plan, Test Case Specification, Test Case Execution & Analysis, Defect Logging & Tracking, Black Box Testing, White Box Testing.

Learning Outcomes: At the end of this unit, Students are able to

1. Summarize various Software Testing Techniques.
2. Design Test Suites for Test Scenarios.

Text Books:

1. Timothy C. Lethbridge, "Object Oriented Software Engineering" (Practical Software Development using UML and Java" Tata McGraw-Hill, 2nd Edition, 2019.
2. Booch, Maksimchuk, Engle, Young, Conallen and Houston, "Object Oriented Analysis and Design with Applications ", Pearson Education, 3rd Edition, 2009.
3. Pankaj Jalote, An integrated Approach to Software Engineering, Springer, 3rd edition, 2005.

Reference Books:

1. Ivar Jacobson, "Object Oriented Software Engineering", Pearson, 2009.
2. Rumbaugh et. al, "Object Oriented Modeling and Design", Pearson.
3. Bertrand Meyer, *Object-Oriented Software Construction*, Prentice Hall, 2nd edition, 1998.
4. Edwards Yourdon, Carl Argila,"Case Studies in object oriented analysis and design" Prentice Hall.

Prepared By

Mrs. S. A. Bhavani, Assistant Prof, Dept of CSE
Mrs. S.V.S.S.Lakshmi, Assistant Prof, Dept of CSE
Mrs. N. Lokeswari, Assistant Prof, Dept of CSE
Mr. P. Krishnanjaneyulu, Assistant Prof, Dept of CSE

WEB TECHNOLOGIES	
CSD 325	**CREDITS : 3**
INSTRUCTION : 2 Periods & 1 Tut/Week	**SESSIONAL MARKS : 40**
END EXAM : 3 Hours	**END EXAM MARKS : 60**

Prerequisites:

- Basic Knowledge of Programming Fundamentals
- Knowledge of Programming Languages (C,JAVA,SQL)

Course Objectives:

The course should enable the students:

- To learn designing of dynamic web pages by embedding with CSS, Java Script.
- To know how to design and develop simple database driven web application.
- Making the web applications using Node JS and Express JS
- To know how to design the web pages using Python
- Understanding with NoSQL database technology.

.

Course Outcomes:

By the end of the course, the student will be able to

1.	Design the Webpages and templates.
2.	Demonstrate on creating a website using PHP and MySQL functionalities.
3.	Solve complex type of web applications using Node JS and Express JS
4.	Develop the Web applications using Python.
5.	Create the Document Oriented Database to interact with NoSQL

CO-PO Mapping:

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

SYLLABUS

UNIT 1: **16 Periods**

HTML: Basic Tags, fonts, hyperlink, lists, tables, images, frames forms, Cascading style sheets.
Java Script: Introduction, Objects, Events, DHTML.
Learning Outcomes: At the end of this unit the Students will be able

1. To design webpages and validation of webpages.
2. Test and debug JavaScript web applications.

UNIT 2: **12 Periods**

PHP: Introduction, Variables, Data types, control statements, Arrays, Functions, Strings, Form processing, File Handling, Cookies and Sessions, PHP with Database.
Learning Outcomes: At the end of this unit the Students will be able

1. To develop the PHP scripts to handle HTML forms.
2. To Create PHP programs that use various PHP library functions, and that manipulate files and directories.

UNIT 3: **12 Periods**

NodeJS: Basics and Setup, REPL terminal, NPM and Command Utilities, Global Objects, Modules – (OS, Path, DNS, Net, Domain),File System.
ExpressJS : Environment, Routing, Http Methods, URL Binding, Middleware, Templating, Static files, Form data.
Learning Outcomes: At the end of this unit the Students will be able

1. To Develop interactive Web applications.
2. To Develop single page applications.

UNIT 4 **12 Periods**

Flask: Introduction, Installation, Routing , URL Building, Http methods, Templates, Request Object, Sessions and cookies, File Uploading, Redirect & Errors, Flashing.
Learning Outcomes: At the end of this unit the Students will be able

1. To design webpages using python and validation of webpages.
2. Apply the routing techniques to access the desired pages.

UNIT 5: **12 Periods**
MongoDB: Introduction**,** Installation, Data Types. Data Modelling. **Database**-Create, Drop. **Collection**-Create, Drop. Aggregation, Indexing, CRUD-Operations,

Learning Outcomes: At the end of this unit the Students will be able

1. To learn how to store and access data from database and dynamic database activities.
2. To learn how to run queries against a MongoDB instance in order to store, manipulate, and retrieve data on it.

Text Books:

1. Thomas A. Powell, - “HTML & CSS Complete Reference”,Mc Graw Hill, Fifth Edition,2017.
2. Steven Holzner ,”Php: The Complete Reference “, McGrawHillEducation , fifth edition ,2008.
3. Reg Lim, “ Beginning Node.js, Express & MongoDB Development”, Kindle Edition, 2019.
4. Miquel Grinberg ,-Flask Web Development, O'Reilly,Second Edition,2018

Reference Books:

1. Kogent Learning Solutions Inc. Web Technologies: HTML, JAVASCRIPT , PHP , JAVA , JSP , ASP.NET, XML and Ajax,Black Book, Dreamtech Press , 1st edition (1 January 2009).
2. Kyle Banker, MongoDB in Action, Manning, Second Edition, March 2016.

Web Resources:

1. https://nptel.ac.in/courses/106/105/106105084/
2. https://www.coursera.org/learn/html-css-javascript-for-web-developers
3. https://www.coursera.org/learn/server-side-nodejs

Prepared By

Mrs. G. Pranitha, Assistant Prof, Dept of CSE
Mr. Bosu Babu, Assistant Prof, Dept of CSE
Mr. D. Prasad, Assistant Prof, Dept of CSE
Mrs. Swathi, Assistant Prof, Dept of CSE

MACHINE LEARNING	
CSD 326	**Credits: 3**
Instruction:3Periods&1Tut/Week	SessionalMarks: 40
EndExam: 3Hours	EndExamMarks:60

Course Objectives:

- Understand the basic concepts of Machine Learning, in particular focusing on the core concepts of supervised and unsupervised learning.
- Solve classification problems using concept learning and multiclass classification
- Apply tree based and rule-based model to classify real world data
- Choose appropriate feature extraction methods in the data handing

Course Outcomes: By the end of the course, the student will be able to

Upon successful completion of the course the students will be able to:

1. Illustrate the characteristics of machine learning models
2. Develop classification models using concept learning.
3. Apply Tree based and rule models to real world problems
4. Build machine learning models based on linear and distance-based models
5. Apply feature extraction methods for real world data handling

Mapping of Course Outcomes with Program Outcomes and Program Specific Outcomes:

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

SYLLABUS

UNIT -I: **12 Periods**

The ingredients of machine learning, Tasks: the problems that can be solved with machine learning, Models: the output of machine learning, Features, the workhorses of machine learning. **Binary classification and related tasks:** Classification, Scoring and ranking, Class probability estimation

Learning Outcomes: at the end of this unit students are able to

1. Understand the basic concepts of machine learning models
2. To know the features and tasks of Binary Classification.

UNIT-II: **12 Periods**

Beyond binary classification: Handling more than two classes, Regression, Unsupervised and descriptive learning. **Concept learning**: The hypothesis space, Paths through the hypothesis space, Beyond conjunctive concepts

Learning Outcomes: at the end of this unit student are able to

1. Identify the models based on concept learning
2. Recognize the difference between regression and ordinal regression and Represent complex prediction problems in a formal learning setting.

UNIT- III: **12 Periods**

Tree models: Decision trees, Ranking and probability estimation trees, Tree learning as variance reduction. **Rule models:** Learning ordered rule lists, Learning unordered rule sets, Descriptive rule learning, First-order rule learning

Learning Outcomes: at the end of this unit student are able to

1. Analyze Tree models and Rule models.
2. Examine various algorithms related to problem solving

UNIT-IV: **12 Periods**

Linear models: The perception: heuristic learning algorithm for linear classifiers, Support vector machines, obtaining probabilities from linear classifiers, Going beyond linearity with kernel methods. **Distance Based Models:** Introduction, Neighbors and exemplars, Nearest Neighbors' classification, Distance Based Clustering, Hierarchical Clustering.

Learning Outcomes: at the end of this unit student are able to

1. Work on linear and distance-based models
2. Apply Clustering Techniques to various applications

UNIT-V: **12 Periods**

Features: Kinds of feature, Feature transformations, Feature construction and selection. Model ensembles: Bagging and random forests, Boosting. **Dimensionality Reduction:** Principal Component Analysis (PCA), Implementation and demonstration.

Learning Outcomes: at the end of this unit students are able to

1. Understand the concept of feature construction and selection.
2. Implement Dimensionality Reduction

TEXTBOOKS:

1. Peter Flach "Machine Learning: The Art and Science of Algorithms that Make Sense of Data", Cambridge University Press, Edition 2012.
2. Tom. M. Mitchell, "Machine Learning", McGraw Hill, 1997.

REFERENCEBOOKS:

1. Understanding Machine Learning: From Theory to Algorithms, Shai Shalev Shwartz, ShaiBen- David, Cambridge.
2. Machine Learning in Action, Peter Harington, 2012,Cengage.

Web Resource:

1. https://www.coursera.org/specializations/machine-learning

Prepared By

Dr. M. Rama Krishan Murthy, Prof, Dept of CSE
Dr. K. Prasad, Assistant Prof, Dept of CSE
Mrs. Asha Grace Roy, Assistant Prof, Dept of CSE
Mrs. D. Anusha, Assistant Prof, Dept of CSE

WEB TECHNOLOGIES LAB	
CSD 327	**CREDITS:1.5**
INSTRUCTION: 3 Peiods Lab/ week	**SESSIONAL MARKS: 50**
END EXAM: 3 Periods	**END EXAM MARKS: 50**

Prerequisites:

- Basic knowledge of Computer Networks
- Exposure to Problem solving techniques and programming skills
- Basic knowledge of JAVA and Python Programming.

Course objectives:

- Introducing new web application development languages and tools to students.
- Introducing Open Source Technologies- HTML, CSS, JAVASCRIPT, PHP, MYSQL, FLASK.

Course outcomes:

By the end of the course, Student will be able to:

1.	Design Static Web pages and Dynamic Web pages using HTML and validate withJavaScript respectively.
2.	Create website using server side scripting language PHP
3.	Develop interactive Web applications using Node JS and ExpressJS
4.	Demonstrate the CRUD application using Flask and MongoDB.

Mapping of Course Outcomes with Program Outcomes:

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

SYLLABUS

LIST OF EXPERIMENTS

1	Design Static Webpage using HTML Components	CO1
2	Design Webpage using CSS	CO1
3	Create Dynamic Webpage using JavaScript	CO1
4	Develop Dynamic Webpage Using PHP Script	CO2
5	Develop PHP application with Database connection	CO2
6	Implement Modules in NodeJS	CO3
7	Develop mini application using Express and NodeJS	CO3
8	Implement HTTP methods using Flask	CO4
9	Implement Sessions concept using Flask	CO4
10	Develop CRUD operations using MongoDB	CO4
11	Develop Main Project using Python/Express and MongoDB/MySql.	CO4

Text Books:

1. Thomas A. Powell, - "HTML & CSS Complete Reference",Mc Graw Hill, Fifth Edition,2017.
2. Steven Holzner ,"Php: The Complete Reference ", McGrawHillEducation , fifth edition ,2008.
3. Reg Lim, " Beginning Node.js, Express & MongoDB Development", Kindle Edition, 2019.
4. Miquel Grinberg ,-Flask Web Development, O'Reilly,Second Edition,2018

Reference Books:

1. Kogent Learning Solutions Inc. Web Technologies: HTML, JAVASCRIPT , PHP , JAVA , JSP , ASP.NET, XML and Ajax,Black Book, Dreamtech Press; 1st edition (1 January 2009).
2. Kyle Banker, MongoDB in Action, Manning, Second Edition, March 2016.

Web Resources:

1. https://nptel.ac.in/courses/106/105/106105084/
2. https://www.coursera.org/learn/html-css-javascript-for-web-developers
3. https://www.coursera.org/learn/server-side-nodejs

Prepared By

Mrs. G. Pranitha, Assistant Prof, Dept of CSE
Mr. Bosu Babu, Assistant Prof, Dept of CSE
Mr. D. Prasad, Assistant Prof, Dept of CSE
Mrs. Swathi, Assistant Prof, Dept of CSE

MACHINE LEARNING LAB	
CSD 328	**Credits:2**
Instruction:3 Hours	Sessional Marks: 40
End Exam: 3Hours	EndExamMarks:60

Description

1. The programs can be implemented in either JAVA or Python/R
2. Data sets can be taken from standard repositories, UCI, Kaggle,etc

Course Objectives:

1. Understand the basic concepts of Machine Learning, in particular focusing on the core concepts of supervised and unsupervised learning.
2. Solve classification problems using concept learning and multiclass classification
3. Apply tree based and rule-based model to classify real world data

Course Outcomes: By the end of the course, the student will be able to

Upon successful completion of the course the students will be able to:

1. Illustrate the characteristics of data and statistical measures
2. Apply Tree based and rule models to real world problems
3. Build machine learning models based on linear and distance-based models
4. Apply feature extraction methods for real world data handling

Mapping of Course Outcomes with Program Outcomes and Program Specific Outcomes:

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

LIST OF EXPERIMENTS:

1. Get and load a dataset. Study the structure of the dataset .Visualize all statistical measures like- mean, median, mode,range,Inter quartile range,Histograms,boxplots and scatterplots.

2. Implement and demonstratethe FIND-Salgorithm for finding the most specific hypothesis based on a given set of training data samples. Read the training data from a .CSV file.

3. For a given set of training data examples stored in a .CSV file, implement and demonstrate the Candidate-Elimination algorithmto output a description of the set of all hypotheses consistent with the training examples.

4. Write a program to demonstrate the working of the decision tree based ID3 algorithm. Use an appropriate data set for building the decision tree and apply this knowledge toclassify a new sample.

5. Write a program to implement the Naive Bayesian classifier for a sample training data set stored as .CSV file. Compute the accuracy of the classifier, considering few test data sets.

6. Assuming a set of documents that need to be classified, use the Naive Bayesian Classifier model to perform this task.

7. Write a program to construct a Bayesian network considering medical data. Use this model to demonstrate the diagnosis of heart patients using standard Heart Disease Data Set. You can use Java/Python ML library classes/API.

8. Apply EM algorithm to cluster a set of data stored in a .CSV file. Use the same data set for clustering using k-Means algorithm. Compare the results of these two algorithms and comment on the quality of clustering. You can add Java/Python ML library classes/API in the program.

9. Implement the non-parametric Locally Weighted Regression algorithm in order to fit data points. Select appropriate data set for your experiment and draw graphs.

10.Write a program to implement k-Nearest Neighbour algorithm to classify the iris data set. Print both correct and wrong predictions. Python ML library classes can be used for this problem.

11. Train SVM model for a dataset and calculate its accuracy.

12. Write a program for implementing Density based clustering algorithm.

13. Write a program to predict medical expenses using simple linear regression.

Prepared By
Dr. M. Rama Krishan Murthy, Prof, Dept of CSE
Dr. K. Prasad, Assistant Prof, Dept of CSE
Mrs. Asha Grace Roy, Assistant Prof, Dept of CSE
Mrs. D. Anusha, Assistant Prof, Dept of CSE

QUANTITATIVE APTITUDE – II & VERBAL ABILITY

Course Code – Category: CIV 329 - HS | Credits: 1.5

L	T	P	E	O
0	0	3	0	1

Sessional Marks: 100

End Exam: 3 Hours | End Exam Marks: --

Course Objectives:

Quantitative aptitude-II

1. To categorize, apply and use thought process to distinguish between concepts of reasoning
2. To prepare and explain the fundamentals related to various possibilities and probabilities related to quantitative aptitude.
3. To critically evaluate numerous possibilities related to puzzles.

Verbal Aptitude:

1. To categorize and explain principles of grammar in order to minimize errors in English.
2. To list and quote high frequency words by giving relevant examples.
3. To categorize, apply and use data as per the requirement.
4. To construct and make use of idioms, phrasal verbs and other expressions used in professional contexts.
5. To critically evaluate reading material for better comprehension

Course Outcomes:

Quantitative Aptitude-II

The student will be able to

1. Use their logical thinking and analytical abilities to solve reasoning questions from company specific and other competitive tests.
2. Solve questions related to permutation & combinations and probabilities from company specific and other competitive tests.
3. Understand and solve puzzle related questions from specific and other competitive tests.

Verbal Aptitude:

The student will be able to

1. Detect grammatical errors in the text/sentences and rectify them while answering their competitive/ company specific tests and frame grammatically correct sentences while writing.
2. Answer questions on synonyms, antonyms, hyponyms, hypernyms and other vocabulary-based exercises while attempting company specific and other competitive tests.
3. Use their logical thinking ability and solve questions related to reasoning based exercises.

4. Choose the appropriate word/s/phrases suitable to the given context in order to make the sentence/paragraph coherent.
5. Analyse the given data/text and find out the correct responses to the questions asked based on the reading exercises; identify relationships or patterns within groups of words or sentences.

Mapping of Course Outcomes with Program outcomes:

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

SYLLABUS

Section –A (Quantitative Aptitude –II)

UNIT I 8 Periods

Numerical Reasoning:

Problems related to Number series, Analogy of numbers, Classification of numbers, Letter series, Seating arrangements, Directions, blood relations and puzzle test.

UNIT II 4 Periods

Combinatorics:

Counting techniques, Permutations, Combinations and Probability

UNIT III 4 Periods

Data sufficiency

Syllogisms

UNIT IV 4 Periods

Application of Base system:

Clocks (Base 24), Calendars (Base7), Cutting of Cubes and cuboids

UNIT V 4 Periods

Puzzle Solving & Time Management using various problems solving tools and techniques:

Selective puzzles from previous year placement papers

Selective puzzles from book Puzzles to puzzle you by shakunataladevi

Selective puzzles from book more puzzles by shakunataladevi

Selective puzzles from book puzzles by George summers

Books for practice

1. Quantitative aptitude by RS Agarwal, S Chand Publications-Latest Edition-2021 - Revised
2. Verbal and non verbal Reasoning by RS Agarwal from S Chand publications-2018 publications -Revised Edition
3. More puzzles by shakunatala devi orient paper back publication-Old Edition -1976

References:

1. Barron's by Sharon Welner Green and Ira K Wolf - Galgotia Publications pvt. Ltd.)-28th Edition-2017
2. A new Approach to Reasoning Verbal & Non-Verbal by BS Sijwali Arihant publications -2017 Edition Revised
3. Logical Reasoning for CAT Arun Sharma McGraw hill publications -4TH Edition-2017 Edition

Websites:

1. www.m4maths.com
2. www.Indiabix.com
3. 800score
4. Official GRE site
5. Official GMAT site

Note: The concepts learnt in Semester I will be tested in the Mid-term and Semester end exams during the II Semester as well.

Reading/ Listening material:

1. Newspapers like 'The Hindu', 'Times of India', 'Economic Times'.
2. Magazines like Frontline, Outlook and Business India.
3. News channels NDTV, National News, CNN

References:

1. Books written by Stephen Covey and Dale Carnegie-Seven Habits of Highly Effective People etc-Simon & Schuster, Running Press book publishers
2. Books written by Bertrand Russell-Oxford University Press

Suggested General Reading

1. Who Moved My Cheese? By Spencer Johnson-GP Putnam's Sons

2. The art of War-Sun Tzu by Nabla, Barnes & Noble
3. The Monk Who Sold Ferrari-Robin Sharma by Harper Collins, Jaico Publishers
4. The Hobbit and other books by JRR Tolkein-Harper Collins

Suggested Authors

1. William Dalrymple
2. V.S.Naipaul
3. Kushwanth Singh
4. Ernest Hemingway
5. Charles Dickens
6. Leo Tolstoy
7. R.K. Narayan
8. Amitav Ghosh
9. Vikram Seth
10. Oscar Wilde

Section –B **(Verbal Ability)**

UNIT I 7 Periods

Grammar:

Parts of speech(with emphasis on appropriate prepositions, co-relative conjunctions, pronouns-number and person, relative pronouns), articles(nuances while using definite and indefinite articles), tenses(with emphasis on appropriate usage according to the situation), subject – verb agreement (to differentiate between number and person) , clauses (use of the appropriate clause , conditional clauses), phrases(use of the phrases, phrasal verbs), degrees of comparison(comparing apples and oranges, comparison and number), modifiers(misplaced and dangling modifiers, absence of modifiers), determiners, parallelism in structure(symmetry in two part sentences), word order, subjunctive mood, redundancy, special types of sentences, miscellaneous types, identifying errors in a given sentence, correcting errors in sentences.

UNIT II 4 Periods

Vocabulary:

Synonyms and synonym variants (with emphasis on high frequency words), antonyms and antonym variants (with emphasis on high frequency words), homonyms, hyponyms, hypernyms and General idioms.

UNIT III 5 Periods

Reasoning:

Critical reasoning (understanding the terminology used in CR- premise, assumption, inference, conclusion), Sequencing of sentences (to form a coherent paragraph, to construct a meaningful and grammatically correct sentence using the jumbled text), to use logical reasoning and eliminate the unrelated word from a group.

UNIT IV 4 Periods

Usage:

Sentence completion (with emphasis on signpost words and structure of a sentence), contextual meanings (to use the appropriate word according to the situation), supplying a suitable beginning/ending/middle sentence to make the paragraph coherent, idiomatic language (with emphasis on business communication), punctuation depending on the meaning of the sentence, run on errors, sentence fragments, coma splices.

UNIT V 4 Periods

Reading Comprehension:

Types of passages (to understand the nature of the passage), types of questions (with emphasis on inferential and analytical questions), style and tone (to comprehend the author’s intention of writing a passage), strategies for quick and active reading(importance given to skimming,

scanning), summarizing ,reading between the lines, reading beyond the lines, techniques for answering questions related to vocabulary (with emphasis on the context), supplying suitable titles to the passage, identifying the theme and central idea of the given passages.

Books for Practice

1. Practical English Grammar A. J. Thomson, A. V. Martinet by Oxford University press ~~4th Edition~~ -2021
2. Remedial English Grammar for Foreign Students by FT wood published by Macmillan *Publishers-2019 Revised Edition*
3. Objective English-Edgar Torpe, Showick Thorpe-Pearson Education-5th Edition-2016
4. Cambridge and Oxford Dictionaries-Revised Editions -2021

Reference Books and websites:

1. Barron's by Sharon Welner Green and Ira K Wolf (Galgotia Publications Pvt.Ltd.)-28th Edition-2017
2. Websites: Indiabix, 800 score, official CAT, GRE and GMAT sites-Updated
3. Word Power Made Easy by Norman Lewis-Goyal Publishers-Revised 2021

PYTHON PROGRAMMING (Common to CIVIL, CHEM, MECH, EEE and ECE)	
Code: OE-I	**Credits : 3**
Instruction **:** 3 Periods & 1 Tut/Week	Sessional Marks : 40
End Exam : 3 Hours	End Exam Marks : 60

Prerequisites:

Basic Knowledge of Programming Fundamentals

Course Objectives:

The course should enable the students:

COURSE OBJECTIVES:

- Describe the core syntax and semantics of Python programming language.
- Illustrate the process of structuring the data using lists, dictionaries, tuples, strings and sets.
- Discover the need for working with the functions, modules and packages.
- Infer the Object-oriented Programming concepts in Python.
- Familiarize the advanced concepts like Iterators, generators, decorators and Indicate the use of regular expressions and built-in functions to navigate the file system.

.**Course Outcomes:**

By the end of the course, the student will be able to:	
1.	Interpret the fundamental Python syntax and semantics and able to solve, test and debug python programs
2.	Fluency in the use of Python control flow statements and Determine the methods to create and manipulate Python programs by utilizing the data structures like lists, dictionaries, tuples, strings and sets.
3.	Express proficiency in the handling of functions, modules and packages.
4.	Articulate the Object-Oriented Programming concepts such as encapsulation, inheritance and polymorphism as used in Python.
5.	List the usage and application of iterators, generator, decorators and Identify the commonly used operations involving file systems and regular expressions.

Mapping of Course Outcomes with Program Outcomes:

S.No	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO 11	PO12	PSO1	PSO2
CO 1														
CO 2														
CO 3														
CO 4														

CO5														

SYLLABUS

UNIT-I:

10 periods

Introduction: Installation, Keywords and Identifiers, Statement, Indentation, Comments, Variables, Constants, Literals, Data Types, Type Conversion, I/O, Import, Operators (Arithmetic operators, Comparison operators, Logical operators, Bitwise operators, Assignment operators, Identity operators, Membership operators), Namespace and Scope.

Learning Outcome: At the end of this Unit the student will be able to

- Analyse fundamental advantages of python over the other programming languages.
- Solve, test and debug basic problems using python script.

UNIT-II:

14 periods

Flow control & Collections: If, If...else, if...elif...else, Nested if, for loop, while loop, Break, Continue and Pass. Numbers, Decimal, Fractions, Mathematics, List, Tuple, String, Set and Dictionary. Data types manipulations (create, Index, Negative indexing, Slicing, change or add elements, delete or remove elements, Methods, Comprehension, Membership Test, Iteration, Operations and Built in Functions)

Learning Outcome: At the end of this Unit the student will be able to

- Implement Flow control statements required real world problems.
- Manipulate python programs by using the python data structures like lists, dictionaries, tuples, strings and sets.

UNIT-III:

12 periods

Functions: Function, Function argument, Recursion, Anonymous / Lambda functions, Global, Local and Nonlocal variables, Global keyword, Modules and Packages.

Learning Outcome: At the end of this Unit the student will be able to

- Resolve real world problems using python functions.
- Familiarize the usage of Modules and packages to enhance the problem solving.

UNIT-IV:

12 periods

Object oriented programming: Introduction to OOPs, Class, Object, Constructors, Methods, Inheritance, Method Overriding, Multiple Inheritance, Operator overloading, Encapsulation and Polymorphism.

Learning Outcome: At the end of this Unit the student will be able to

- Design object-oriented programs with Python classes.
- Usage of inheritance, encapsulation, inheritance and polymorphism for reusability.

UNIT-V:

12 periods

Advanced topics: Iterators, Building Your Own Iterator, Infinite Iterators, Generators, Generator Expression, Closure Function, Decorators, @property decorator, Getters and Setters, RegEx, Match object, datetime, Files(Open, Read, Write, Close) and File Methods,

Learning Outcome: At the end of this Unit the student will be able to

- Interpret the advantages of advanced concepts like iterators, generator, decorators and regular expressions.
- Identify the commonly used operation involved in files for I/O processing.

TEXT BOOKS

1. Core Python programming, by W.Chun, Pearson
2. Python Programming : A Modern Approach by Vamsi Kurama, Pearson

REFERENCE BOOKS

1. 1. How To Think Like A Computer Scientist, Learning With Python, by Allen Downey, Jeffrey Elnker and Chris Meyers
2. Introduction to Python Programming, Gowrishankar S, Veena A, CRC Press/Taylor & Francis.
3. A Beginners Guide to Python 3 Programming by John Hunt, Springer

---

Prepared By: Dr K.Prasad

CASE STUDIES

1. Jack and his three friends have decided to go for a trip by sharing the expenses of the fuel equally. Write a Python program to calculate the amount (in Rs) each of them need to put in for the complete (both to and fro) journey.
The program should also display True, if the amount to be paid by each person is divisible by 5, otherwise it should display False. (Hint: Use the relational operators in print statement.) Assume that mileage of the vehicle, amount per litre of fuel and distance for one way are given.

Test your code by using the given sample inputs.
Verify your code by using the 2nd sample input(highlighted) given below:

Sample Input			Expected Output
Mileage of the vehicle (km/litre of fuel)	Amount per litre of fuel (Rs)	Distance for one way (kms)	
12	65	96	260.0 True
12	40	190	

2. A three digit number is said to be an "Armstrong number" if the sum of the third power of its individual digits is equal to the number itself.
Example: 371 is an Armstrong number as 371 = 33 + 73 + 13
407 is an Armstrong number as 407 = 43 + 03 + 73
Write a pseudo-code to check whether a given three digit number is an Armstrong number.

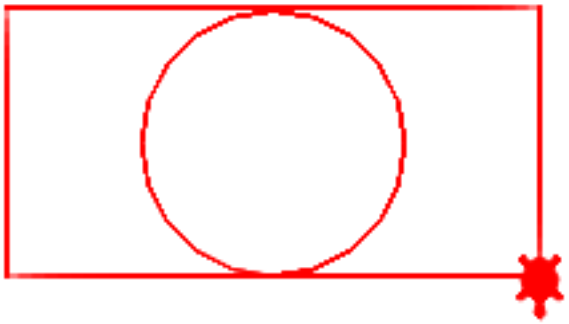
3. A University offering degree courses to students has decided to provide scholarship based on the following details:

Branch of study	Score (%)	Scholarship %	Remarks
Arts	Score is at least 90	50	The student is eligible only for one scholarship% even if both the score conditions are valid for the given branch of study. In such cases, students are eligible for the highest scholarship% applicable among the two.
Arts	Score is an odd number	5	
Engineering	Score is more than 85	50	

Branch of study	Score (%)	Scholarship %	Remarks
Engineering	Score is divisible by 7	5	

If there are 500 students who have joined the university, write a pseudo-code to calculate and display the final fees to be paid by each student. You may accept the branch of study, score and course fee as inputs for each student and calculate the final fees to be paid by each student based on formulae given below:
Scholarship amount=course fee * (scholarship%)
Final fee= course fee - scholarship amount

4. Write a program to create the following pattern:



5. Write a python program to find and display the product of three positive integer values based on the rule mentioned below: It should display the product of the three values except when one of the integer value is 7. In that case, 7 should not be included in the product and the values to its left also should not be included. If there is only one value to be considered, display that value itself. If no values can be included in the product, display -1.

Note: Assume that if 7 is one of the positive integer values, then it will occur only once. Refer the sample I/O given below.

Sample Input	Expected Output
1, 5, 3	15
3, 7, 8	8
7, 4, 3	12
1, 5, 7	-1

6. A traveller on a visit to India is in need of some Indian Rupees (INR) but he has money belonging to another currency. He wants to know how much money he should provide in the currency he has, to get the specified amount in INR.

Write a python program to implement a currency calculator which accepts the amount needed in INR and the name of the currency which the traveller has. The program should identify and display the amount the traveller should provide in the currency he has, to get the specified amount in INR.

Note: Use the forex information provided in the table below for the calculation. Consider that only the currency names mentioned in the table are valid. For any invalid currency name, display -1.

Currency	Equivalent of 1.00 INR
Euro	0.01417
British Pound	0.0100
Australian Dollar	0.02140
Canadian Dollar	0.02027

7. Write a python program to generate and display the next date of a given date.

Assume that
Date is provided as day, month and year as shown in below table.
The input provided is always valid. Output should be day-month-year.
Hint: print(day,"-",month,"-",year) will display day-month-year

	Sample Input	Expected Output
Day	1	2-9-2020
Month	9	
Year	2020	

8. Write a python program which finds the maximum number from num1 to num2 (num2 inclusive) based on the following rules.

1.Always num1 should be less than num2
2.Consider each number from num1 to num2 (num2 inclusive). Populate the number into a list, if the below conditions are satisfied
    a. Sum of the digits of the number is a multiple of 3
    b. Number has only two digits
    c. Number is a multiple of 5
3.Display the maximum element from the list
In case of any invalid data or if the list is empty, display -1.

9. Given a string containing uppercase characters (A-Z), compress the string using Run Length encoding. Repetition of character has to be replaced by storing the length of that run.

Write a python function which performs the run length encoding for a given String and returns the run length encoded String.

Provide different String values and test your program.

Sample Input	**Expected Output**
AAAABBBBCCCCCCCC	4A4B8C
AABCCA	2A1B2C1A

10. A hospital wants to know the medical speciality visited by the maximum number of patients. Assume that the patient id of the patient along with the medical speciality visited by the patient is stored in a list. The details of the medical specialities are stored in a dictionary as follows:

```
{
"P":"Pediatrics",
"O":"Orthopedics",
"E":"ENT
}
```

Write a function to find the medical speciality visited by the maximum number of patients and return the name of the speciality.

Note:

1. Assume that there is always only one medical speciality which is visited by maximum number of patients.
2. Perform case sensitive string comparison wherever necessary.

Sample Input	Expected Output
[101,P,102,O,302,P,305,P]	Pediatrics
[101,O,102,O,302,P,305,E,401,O,656,O]	Orthopedics
[101,O,102,E,302,P,305,P,401,E,656,O,987,E]	ENT

11. Write a python program to display all the common characters between two strings. Return -1 if there are no matching characters.
Note: Ignore blank spaces if there are any. Perform case sensitive string comparison wherever necessary.

Sample Input	Expected output
"I like Python" "Java is a very popular language"	lieyon

12. A teacher is in the process of generating few reports based on the marks scored by the students of her class in a project based assessment.
Assume that the marks of her 10 students are available in a tuple. The marks are out of 25.

Write a python program to implement the following functions:

1.find_more_than_average(): Find and return the percentage of students who have scored more than the average mark of the class.
2. generate_frequency(): Find how many students have scored the same marks. For example, how many have scored 0, how many have scored 1, how many have scored 3….how many have scored 25. The result should be populated in a list and returned.
3. sort_marks(): Sort the marks in the increasing order from 0 to 25. The sorted values should be populated in a list and returned.

Sample Input	Expected Output
list_of_marks = (12,18,25,24,2,5,18,20,20,21)	70.0 [0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 2, 0, 2, 1, 0, 0, 1, 1] [2, 5, 12, 18, 18, 20, 20, 21, 24, 25]

13. Write a python function, **check_double(number**) which accepts a whole number and returns True if it satisfies the given conditions.

1. The number and its double should have exactly the same number of digits.
2. Both the numbers should have the same digits ,but in different order.

Otherwise it should return False.

Example: If the number is 125874 and its double, 251748, contain exactly the same digits, but in a different order.

14. Given a number n, write a program to find the sum of the largest prime factors of each of nine consecutive numbers starting from n.
g(n) = f(n) + f(n+1) + f(n+2) + f(n+3) + f(n+4) + f(n+5) + f(n+6) + f(n+7) + f(n+8)
where, g(n) is the sum and f(n) is the largest prime factor of n

For example,
g(10)=f(10)+f(11)+f(12)+f(13)+f(14)+f(15)+f(16)+f(17)+f(18)
=5 + 11 + 3 + 13 + 7 + 5 + 2 + 17 + 3
=66

15. Write a python function, **nearest_palindrome()** which accepts a number and returns the nearest palindrome greater than the given number.

Sample Input	Expected Output
12300	12321
12331	12421

16. Assume that a poem is given. Write the regular expressions for the following:

1. Print how many times the letter 'v' appears in the poem.
2. Remove all the newlines from the poem and print the poem in a single line.
3. If a word has 'ch' or 'co', replace it with 'Ch' or 'Co'.
4. If the pattern has characters 'ai' or 'hi', replace the next three characters with **.

Test your code by using the given sample inputs.
Verify your code by using the 2nd sample input(highlighted) given below:

Sample Input	Expected Output
If I can stop one heart from breaking, I shall not live in vain; If I can ease one life the aching, Or cool one pain,	4 If I can stop one heart from breaking, I shall not live in vain; If I can ease one life the aching, Or cool one pain,

Or help one fainting robin Unto his nest again, I shall not live in vain.	Or help one fainting robin Unto his nest again, I shall not live in vain. If I can stop one heart from breaking, I shall not live in vain; If I can ease one life the aChing, Or Cool one pain, Or help one fainting robin Unto his nest again, I shall not live in vain. If I can stop one heart from breaking, I shall not live in vain; If I can ease one life the achi** Or cool one pain, Or help one fai**ng robin Unto hi**est again, I shall not live in vain.
It takes strength for being certain, It takes courage to have doubt. It takes strength for challenging alone, It takes courage to lean on another. It takes strength for loving other souls, It takes courage to be loved. It takes strength for hiding our own pain, It takes courage to help if it is paining for someone.	

17. A university wants to automate their admission process. Students are admitted based on marks scored in a qualifying exam.
A student is identified by student id, age and marks in qualifying exam. Data are valid, if:

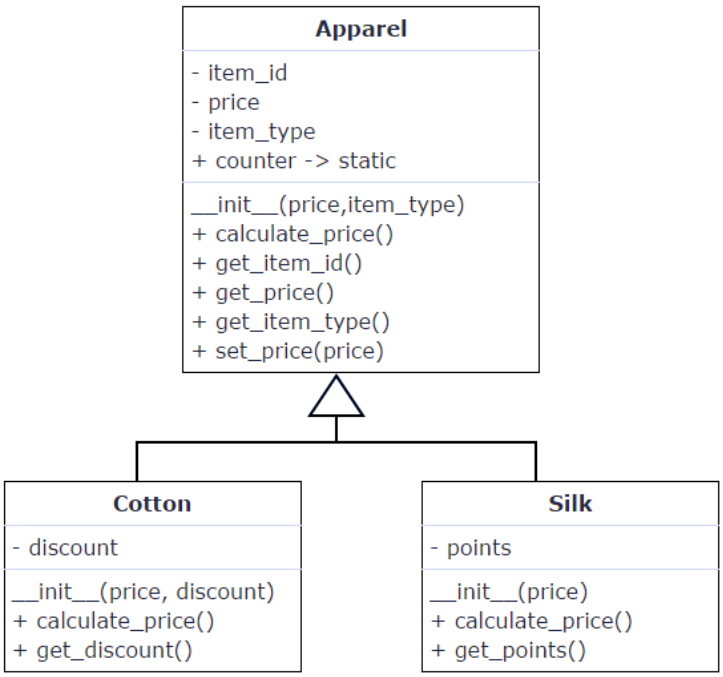
- Age is greater than 20
- Marks is between 0 and 100 (both inclusive)

A student qualifies for admission, if

- Age and marks are valid and
- Marks is 65 or more

Write a python program to represent the students seeking admission in the university.

18. An apparel shop wants to manage the items which it sells. Write a python program to implement the class diagram given below.



19. Write a python program to Find Resolution of JPEG Image

20. Royal Orchid is a florist. They want to be alerted when stock of a flower goes below a particular level.
The flowers are identified using name, price per kg and stock available (in kgs).
Write a Python program to implement the above requirement.

Introduction to Machine Learning	
CSM OE2	**Credits: 3**
Instruction : 3 Periods / Week	Sessional Marks : 40
End Exam : 3 Hours	End Exam Marks : 60

Prerequisites:

- Pythonprogramming skill.
- Preliminary concepts of probability, algebra and statistics.

Course Objectives:

- Understand the main categories and fundamental concepts of Machine Learning systems
- Familiarize the main steps in a typical Machine Learning project
- Effective utilization of regression, clustering and classification algorithms.
- Ability to apply machine learning techniques to solve real world problems

Course Outcomes:

By the end of the course, the student will be able to:	
1.	Definethe machine learning, types of ML and challenges of ML.
2.	Describe the extraction of data, withdraw insights of data and fine tune the data for applying ML model.
3.	Implementation of Classification models on MNIST Data set and Regression models.
4.	Apply SVM, DT and Ensemble Learning techniques for solving Real world problems.
5.	Solve curse of dimensionality problemand Apply K-Means, DBSCAN and Gaussian Mixtures to solve Unlabeled data.

Mapping of course outcomes with program outcomes:

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

SYLLABUS

UNIT-I: **10 Periods**

Machine learning landscape:
What Is Machine learning?, Why Use Machine Learning?, Types of Machine Learning Systems, Main Challenges of Machine Learning, Testing and Validating.
Learning Outcomes: At the end of this unit, the students will be able to

- Explain ML and Category of ML.
- Define Challenges of ML, Testing and validating ML model.

UNIT-II: **12 Periods**

End-to-End Machine learning Project:
Working with Real Data, Get the Data, Discover and Visualize the Data to Gain Insights, Prepare the Data for Machine Learning Algorithms, Select and Train a Model, Fine-Tune Your Model.

Learning Outcomes: At the end of this unit, the students will be able to

- Explore the real data, extraction of relevant data and gain the insights of data.
- Selection, Train a model and Finetuning the model for real data .

UNIT-III: **12 Periods**

Training Model and MNIST Data set Classification:
Linear Regression, Gradient Descent, Polynomial Regression, Logistic Regression.
MNIST, Training a Binary Classifier, Performance Measures, Multiclass Classification, Error Analysis, Multilabel Classification, Multioutput Classification.
Learning Outcomes: At the end of this unit, the students will be able to

- Explore the Regression models and MNIST data set.
- Implement Regression models and classifications on MNIST dataset.

UNIT-IV: **15 Periods**

SVM, Decision Trees and Ensemble Learning:
Linear SVM Classification, Nonlinear SVM Classification, SVM Regression.
Training and Visualizing a Decision Tree, Making Predictions, Estimating Class Probabilities, The CART Training Algorithm, Computational Complexity, Gini Impurity or Entropy?, Regularization Hyperparameters.

Learning Outcomes: At the end of this unit, the students will be able to

- Describe the concept of SVM, DT, RF techniques.
- Apply SVM, DT and RF on Real world data.

UNIT-V: **14 Periods**

Dimensionality Reduction and Clustering:
The Curse of Dimensionality, Main Approaches for Dimensionality Reduction, PCA(Principal Components Analysis).
Clustering, K-Means, Clustering for image segmentation, Clustering for Pre-processing, Clustering for Semi-Supervised Learning, DBSCAN, Gaussian Mixtures.

Learning Outcomes: At the end of this unit ,the students will be able to

- Construct PCA and LLE.
- Apply the K-Means, DBSCAN and Gaussian Mixtures on unlabeled data.

Text Books:

1. AurelionGeron, "Hands-on Machine Learning with Scikit-Learn, Keras, and Tensor Flow: Concepts, Tools and Techniques to build Intelligent Systems", 2/e, O'Reilly Media, 2019.
2. Sebastian Raschka, "Python Machine Learning"Packt Publishing 2015.

Reference Books:

1.Tom M. Mitchell, "Machine Learning" First Edition by Tata McGraw- Hill Education.

2. EthemAlpaydin,"Introduction to Machine Learning " 2nd Edition, The MIT Press, 2009

Web Resources:

1. https://nptel.ac.in/courses/106106139
2. https://www.youtube.com/watch?v=PPLop4L2eGk

Prepared by: Dr M Rama Krishna Murthy, Prof & HOD Dept of CSM,CSD