DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CONVENING ORDER - BOARD OF STUDIES

The Board of Studies is the basic constituent of the academic system of an autonomous college. Its functions will include framing the syllabi for various courses, reviewing and updating syllabi from time to time, introducing new courses of study, determining details of continuous assessment, recommending panels. of examiners under the semester system, etc. The Board of studies is constituted with the following members:

S.No	Name of the Member	Designation/occupation	category
1	Mr. V V B Chari	Head of the Department	Chairman
2	Dr. A S N Chakravarthy	Professor, CSE Department, UCEK, JNTUK Kakinada	University Nominee
3	Dr. P Srinivasa Rao	Asst. prof, Sr. Grade – II, Dept. of CSE, VIT - AP	Subject experts outside parent university
4	Dpr. Srilatha Chebrolu	Asst. prof, Dept. of CSE, NIT - AP	
5	Mr. K Ashok	Technical Lead, Hyundai Capital America	Industrialist
6	Mr. M Sk Subhani	Asst. Professor	Faculty Member
7	Mrs. Sd. Nasrath	Asst. Professor	Faculty Member
8	Mr. A. Madhava Reddy	Asst. Professor	Faculty Member
9	Mrs. K. Madhavi	Asst. Professor	Faculty Member
10	Mr. N. Anjaneyulu	Asst. Professor	Faculty Member
11	Miss. Sk Reshma	Associate Engineer, Cognizant, Hyderabad	Alumni Member

P.T.O.

Term: The term of the nominated members shall be two years.

Meetings: The Board of Studies shall meet at least twice a year.

Functions:

The Board of Studies of a Department in the college shall:

- Prepare syllabi for various courses keeping in view the objectives of the college, interest of the stakeholders and national requirement for consideration and approval of the Academic Council;
- Suggest methodologies for innovative teaching and evaluation techniques;
- Suggest panel of names to the Academic Council for appointment of examiners;
- Coordinate research, teaching, extension and other academic activities in the Department! College.

HoD

Dept.Of CSE

Copy to:

1. Principal.
2. IQAC

AMR/CSE/BOS/2025-26/Circular/1 Date: 15-06-2026.

Meeting Notice

Greetings from A.M. Reddy Memorial College of Engineering and Technology We request you to participate in Computer Science and Engineering Department Board of Study meeting scheduled on 17-06-2026 at 10.00 AM through online (Zoom Meeting)

S.No	Name of the Member	Designation/occupation	category
1	Mr. V V B Chari	Head of the Department	Chairman
2	Dr. A S N Chakravarthy	Professor, CSE Department, UCEK, JNTUK Kakinada	University Nominee
3	Dr. P Srinivasa Rao	Asst. prof, Sr. Grade – II, Dept. of CSE, VIT - AP	Subject experts outside parent university
4	Dr. Srilatha Chebrolu	Asst. prof, Dept. of CSE, NIT - AP	
5	Mr. K Ashok	Technical Lead, Hyundai Capital America	Industrialist
6	Mr. M Sk Subhani	Asst. Professor	Faculty Member
7	Mrs. Sd. Nasrath	Asst. Professor	Faculty Member
8	Mr. A. Madhava Reddy	Asst. Professor	Faculty Member
9	Mrs. K. Madhavi	Asst. Professor	Faculty Member
10	Mr. N. Anjaneyulu	Asst. Professor	Faculty Member
11	Miss. Sk Reshma	Associate Engineer, Cognizant, Hyderabad	Alumni Member

Agenda of the meeting– Introducing the members of Board of Studies.

- Finalization of AMR- 26 CBCS (Choice Based Credit System) – Syllabus for I year I & II Sem which are related to CSE Department.
- Course structure modifications (If any)
- Discuss on Data Structures and Algorithm Analysis (T), Data Structures and Algorithm Analysis (L), Advanced Data warehousing and Data Mining (T), Advanced Data warehousing and Data Mining (L) , Machine Learning(T), Machine Learning Lab(L) , Natural Language Processing(T) , Natural Language Processing Lab(L) in I year - I sem and II Sem.
- Any other matters.

HoD
Dept. Of CSE

Copy to:

1. **Principal**
2. **IQAC**

AMR/CSE/BOS/2025-26/MOM/1 Date : 17-06-2026.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

MINUTES OF MEETING - BOARD OF STUDIES (BOS)

The Meeting of the Board of Studies of CSE was held on 17-06-2026 at 10.00 AM through online (Zoom Meeting Platform). The following members were attended the online meeting.

S.No	Name of the Member	Designation/occupation	category	Signature
1	Mr. V V B Chari	Head of the Department	Chairman	
2	Dr. A S N Chakravarthy	Professor, CSE Department, UCEK, JNTUK Kakinada	University Nominee	
3	Dr. P Srinivasa Rao	Asst. prof, Sr. Grade – II, Dept. of CSE, VIT - AP	Subject experts outside parent university	
4	Dr. Srilatha Chebrolu	Asst. prof, Dept. of CSE, NIT - AP		
5	Mr. K Ashok	Technical Lead, Hyundai Capital America	Industrialist	
6	Mr. M Sk Subhani	Technical Lead, Hyundai Capital America	Faculty Member	
7	Mrs. Sd. Nasrath	Asst. Professor	Faculty Member	
8	Mr. A. Madhava Reddy	Asst. Professor	Faculty Member	
9	Mrs. K. Madhavi	Asst. Professor	Faculty Member	
10	Mr. N. Anjaneyulu	Asst. Professor	Faculty Member	
11	Miss. Sk Reshma	Asst. Professor	Alumni Member	

The Meeting began with chairman, Board of studies extending a warm welcome to all the members of participating in the meeting.

The following points were discussed and Ratified during the meeting

1. The following proposed AMR 25 Course Structure and the detailed syllabi of III-I were presented, discussed and Ratified.

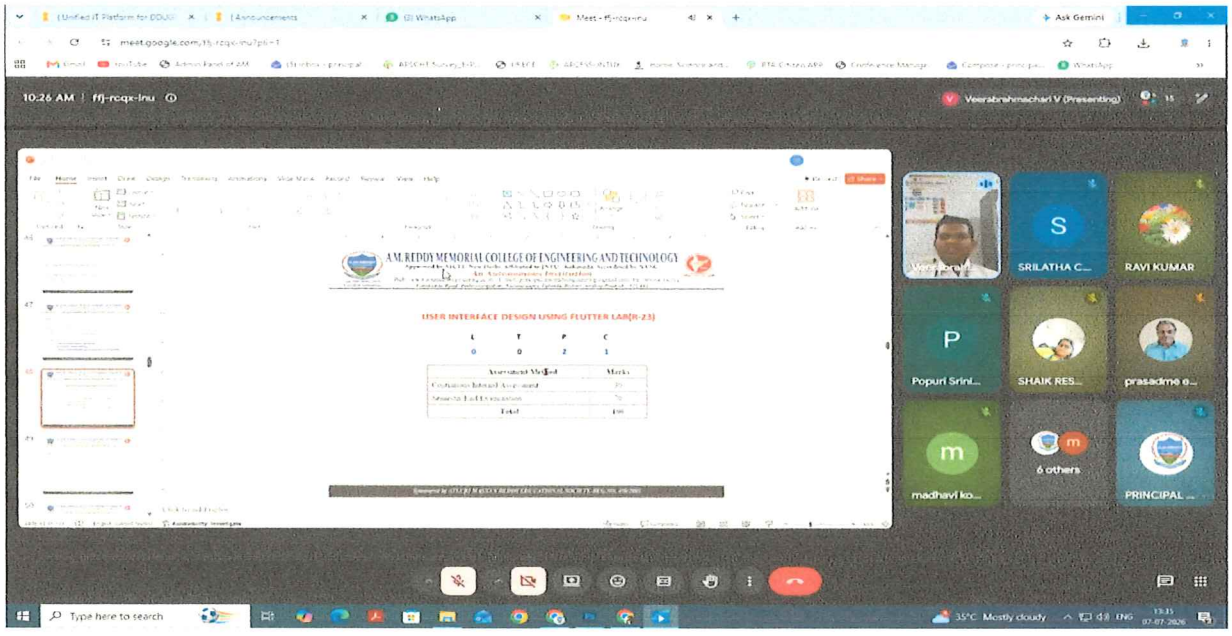
Suggestions:

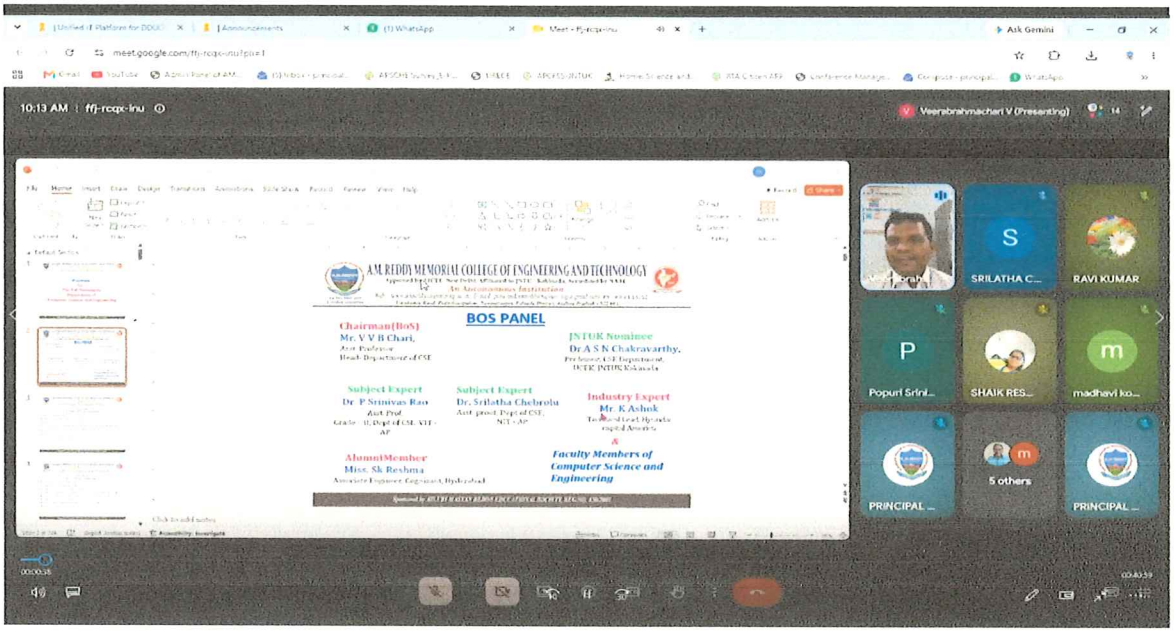
1. **University Nominee's Suggestion:**

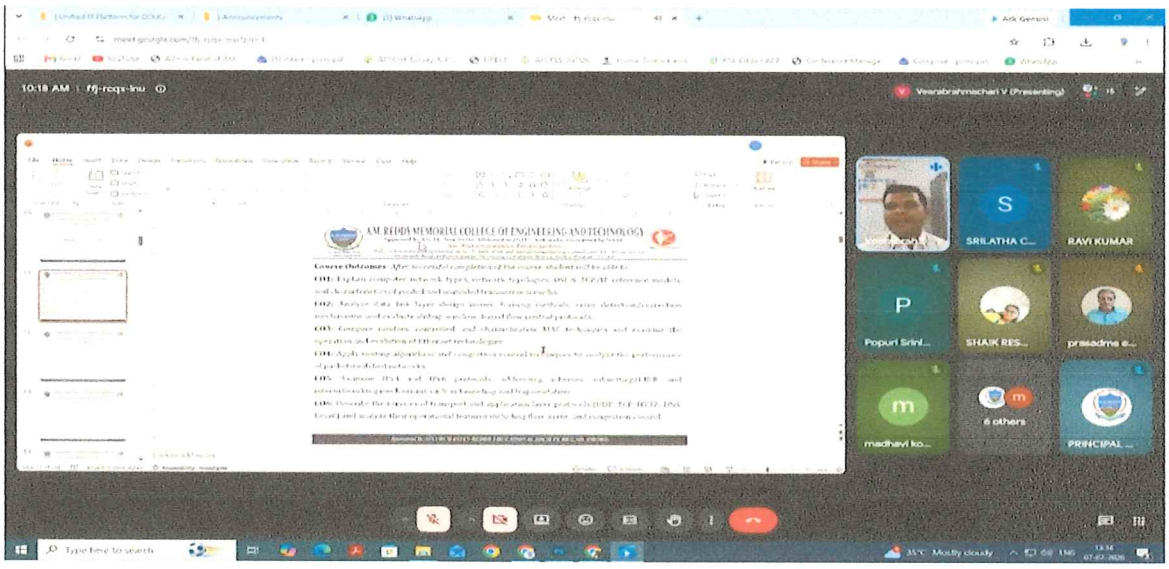
   The University Nominee suggested incorporating the latest editions of prescribed textbooks and reference books in the curriculum to enhance students' knowledge, ensure exposure to recent developments in the field, and improve the overall quality of learning.

2. **Subject Expert's Suggestion:**

   The Subject Expert recommended introducing topics related to Generative Artificial Intelligence (GenAI) in relevant courses, considering its rapid growth and increasing significance across industries. The inclusion of GenAI concepts and applications will help students stay updated with emerging technologies and improve their industry readiness.







Chairman BoS –

Dept. of CSE

Copy to:

1. **Principal**
2. **IQAC**

B.Tech. – III Year I Semester

S.No	Category	Title	L	T	P	C
1	Professional Core	Data Warehousing & Data Mining	3	0	0	3
2	Professional Core	Computer Networks	3	0	0	3
3	Professional Core	Formal Languages and Automata Theory	3	0	0	3
4	Professional Elective-I	1. Object Oriented Analysis and Design 2. Artificial Intelligence 3. Microprocessors & Microcontrollers 4. Quantum Computing 5. 12 week MOOC Swayam/ NPTEL course recommended by the BoS	3	0	0	3
5	Open Elective-I	OR Entrepreneurship Development & Venture Creation	3	0	0	3
6	Professional Core	Data Mining Lab	0	0	3	1.5
7	Professional Core	Computer Networks Lab	0	0	3	1.5
8	Skill Enhancement course	Full Stack Development-2	0	1	2	2
9	Engineering Science	User Interface Design using Flutter / SWAYAM Plus - Android Application Development (with Flutter)	0	0	2	1
10	Evaluation of Community Service Internship		-	-	-	2
	Total		15	1	10	23
MC	Minor Course (Student may select from the same specialized minors pool)		3	0	3	4.5
MC	Minor Course through SWAYAM/NPTEL (minimum 12 week, 3 credit course)		3	0	0	3
HC	Honors Course (Student may select from the same honors pool)		3	0	0	3
HC	Honors Course (Student may select from the same honors pool)		3	0	0	3

Sk. Rasheed

P. Srinivasarao

B.Tech. III Year II Semester

S.No	Category	Title	L	T	P	C
1	Professional Core	Compiler Design	3	0	0	3
2	Professional Core	Cloud Computing	3	0	0	3
3	Professional Core	Cryptography & Network Security	3	0	0	3
4	Professional Elective-II	1. Software Testing Methodologies 2. Cyber Security 3. DevOps 4. Machine Learning 5. 12 week MOOC Swayam/NPTEL course recommended by the BoS	3	0	0	3
5	Professional Elective-III	1. Software Project Management 2. Mobile Adhoc Networks 3. Natural Language Processing 4. Big Data Analytics 5. Distributed Operating System 6. 12 week MOOC Swayam/NPTEL course recommended by the BoS	3	0	0	3
6	Open Elective – II		3	0	0	3
7	Professional Core	Cloud Computing Lab	0	0	3	1.5
8	Professional Core	Cryptography & Network Security Lab	0	0	3	1.5
9	Skill Enhancement course	Soft skills / / SWAYAM Plus - 21st Century Employability Skills	0	1	2	2
10	Audit Course	Technical Paper Writing &IPR	2	0	0	-
		Total	**20**	**1**	**08**	**23**
Mandatory Industry Internship / Mini Project of 08 weeks duration during summer vacation						
MC		Minor Course (Student may select from the same specialized minors pool)	3	0	3	4.5
MC		Minor Course (Student may select from the same specialized minors pool)	3	0	0	3
HC		Honors Course (Student may select from the same honors pool)	3	0	0	3
HC		Honors Course (Student may select from the honors pool)	3	0	0	3

P. Srinivasarao

* Under Industry Internship interested students can pursue SWAYAM Plus courses viz., Hands-on Masterclass on Data Analytics OR Artificial Intelligence for Real-World Application

Open Electives, offered to other department students:

Open Elective I: Principles of Operating Systems/ Computer Organization and Architecture
Open Elective II: Principles of Database Management Systems
Open Elective III: Object Oriented Programming Through Java
Open Elective IV: Principles of Software Engineering /Computer Networks

Minor Engineering

Note:

1. *To obtain Minor Engineering, student needs to obtain 18 credits by successfully completing any of the following courses in the concern stream.*
2. *During Minor/Honors Course selection, there should not be any overlapping with Regular/Major/OPEN Electives*

Minor in CSE

1. Principles of Database Management Systems 3-0-3-4.5 (II-II)
2. Principles of Software Engineering 3-0-0-3 (III-I)
3. Advanced Data Structures & Algorithm Analysis 3-0-3-4.5 (III-II)
4. Principles of Operating Systems 3-0-0-3 (IV-I)

Any of the following 12 Week 3 credit NPTEL MOOC Courses

5. Artificial Intelligence: Knowledge Representation and Reasoning
6. Computer Networks and Internet Protocol
7. Machine Learning and Deep Learning - Fundamentals and Applications
8. Fundamentals of Object Oriented Programming
9. Discrete Mathematics for CS
10. Software Engineering

COURSES OFFERED FOR HONORS DEGREE IN CSE

Note: ***To obtain Honor's degree, student needs to obtain 18 credits by successfully completing any of the following courses in the concern stream.***

1. Social Network Analysis 12 Week 3 Credit Course, MOOCS
2. Applied Linear Algebra in AI & ML 12 Week 3 Credit Course, MOOCS
3. Design & Implementation of Human-Computer Interfaces – NPTEL MOOCS
4. Cryptography and Network Security 12 Week 3 Credit Course, MOOCS
5. Privacy and Security in Online Social Media 12 Week 3 Credit Course, MOOCS

Sk. Reshma
P. Srinivasarao

6. Deep Learning for Natural Language Processing - 12 Week 3 Credit Course, MOOCS
7. Computer Vision - 12 Week 3 Credit Course, MOOCS
8. Applied Time-Series Analysis 12 Week 3 Credit Course, MOOCS
9. Parallel Computer Architecture 12 Week 3 Credit Course, MOOCS
10. Reinforcement Learning 12 Week 3 Credit Course, MOOCS
11. GPU Architecture and Programming 12 Week 3 Credit Course, MOOCS
12. Computational Complexity 12 Week 3 Credit Course, MOOCS
13. Quantum Algorithms and Cryptography 12 Week 3 Credit Course, MOOCS
14. Unmanned Arial Systems & Robotics 12 Week 3 Credit Course, MOOCS
15. Prompt Engineering for Generative AI (III - II)

P. Srinivasarao

III Year I Semester	DATA WAREHOUSING & DATA MINING	L	T	P	C
		3	0	0	3

Pre-requisites: Data Structures, Algorithms, Probability & Statistics, Data Base Management Systems

Course Objectives: The main objective of the course is to

- Introduce basic concepts and techniques of data warehousing and data mining
- Examine the types of the data to be mined and apply pre-processing methods on raw data
- Discover interesting patterns, analyze supervised and unsupervised models and estimate the accuracy of the algorithms.

Course Outcomes (COs)

CO1: Explain the concepts of data warehousing, OLAP, data cubes, cloud data warehouses, and data visualization to support decision-making. (K2)

CO2: Apply data preprocessing techniques—including cleaning, integration, transformation, reduction, and discretization—to prepare datasets for analysis. (K3)

CO3: Analyze classification problems using decision trees, Bayesian and rule-based classifiers, and evaluate models for accuracy and scalability. (K4)

CO4: Develop association rules from transactional data using algorithms such as Apriori and FP-Growth, and perform pruning and compact representation of frequent itemsets (K5) CO5: Analyze clustering techniques such as K-means, hierarchical clustering, and DBSCAN to group data and interpret the resulting clusters for knowledge discovery. (K4)

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H	M	L	L	-	-	-	-	-	-	-	-			
CO2	H	H	M	M	-	-	-	-	-	-	-	-			
CO3	H	H	H	H	M	-	-	-	-	-	-	-			
CO4	H	H	H	H	M	-	-	-	-	-	-	-			
CO5	H	H	M	H	M	-	-	-	-	-	-	-			

UNIT–I: Data Warehousing and Online Analytical Processing: Basic concepts, Data Warehouse Modeling: Data Cube and OLAP, Data Warehouse Design and Usage, Data Warehouse Implementation, Cloud Data Warehouse, Data Mining and Patten Mining, Technologies, Applications, Major issues, Data Objects & Attribute Types, Basic Statistical Descriptions of Data, Data Visualization, Measuring Data Similarity and Dissimilarity. (Text Book- 1)

UNIT II: Data Preprocessing: An Overview, Data Cleaning, Data Integration, Data Reduction, Data Transformation and Data Discretization. (Text Book- 1)

P. Srinivasarao

UNIT–III: Classification: Basic Concepts, General Approach to solving a classification problem, Decision Tree Induction: Attribute Selection Measures, Tree Pruning, Scalability and Decision Tree Induction, Visual Mining for Decision Tree Induction, Bayesian Classification Methods: Bayes Theorem, Naïve Bayes Classification, Rule-Based Classification, Model Evaluation and Selection. (Text Book- 2)

UNIT–IV: Association Analysis: Problem Definition, Frequent Itemset Generation, Rule Generation: Confident Based Pruning, Rule Generation in Apriori Algorithm, Compact Representation of frequent item sets, FP-Growth Algorithm. (Text Book- 2)

UNIT–V: Cluster Analysis: Overview, Basics and Importance of Cluster Analysis, Clustering techniques, Different Types of Clusters; K-means: The Basic K-means Algorithm, K-means Additional Issues, Bi-secting K Means, Agglomerative Hierarchical Clustering: Basic Agglomerative Hierarchical Clustering Algorithm DBSCAN: Traditional Density Center-Based Approach, DBSCAN Algorithm, Strengths and Weaknesses. (Text Book- 2)

Text Books:

1. Data Mining concepts and Techniques, 3rd edition, Jiawei Han, Michel Kamber, Elsevier, 2011.
2. Introduction to Data Mining: Pang-Ning Tan & Michael Steinbach, Vipin Kumar, Pearson, 2012. **Reference Books:**

1. Data Mining: VikramPudi and P. Radha Krishna, Oxford Publisher.
2. Data Mining Techniques, Arun K Pujari, 3rd edition, Universities Press,2013.
3. (NPTEL course by Prof.PabitraMitra) http://onlinecourses.nptel.ac.in/noc17_mg24/preview
4. http://www.saedsayad.com/data_mining_map.htm

P. Srinivasarao

III Year I Semester	COMPUTER NETWORKS	L	T	P	C
		3	0	0	3

Course Objectives:

- To provide insight about networks, topologies, and the key concepts.
- To gain comprehensive knowledge about the layered communication architectures (OSI and TCP/IP) and its functionalities.
- To understand the principles, key protocols, design issues, and significance of each layers in ISO and TCP/IP.
- To know the basic concepts of network services and various network applications.

Course Outcomes: *After successful completion of the course, student will be able to* CO1: Explain computer network types, network topologies, OSI & TCP/IP reference models, and characteristics of guided and unguided transmission media.
CO2: Analyze data link layer design issues, framing methods, error detection/correction mechanisms, and evaluate sliding-window–based flow control protocols.
CO3: Compare random, controlled, and channelization MAC techniques and examine the operation and evolution of Ethernet technologies.
CO4: Apply routing algorithms and congestion-control techniques to analyze the performance of packet-switched networks.
CO5: Examine IPv4 and IPv6 protocols, addressing schemes, subnetting/CIDR, and internetworking mechanisms such as tunneling and fragmentation.
CO6: Describe the services of transport and application layer protocols (UDP, TCP, HTTP, DNS, Email) and analyze their operational features including flow, error, and congestion control.

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H	M	-	-	-	-	-	-	-	M	-	L			
CO2	H	H	-	M	L	-	-	-	-	M	-	L			
CO3	H	M	M	-	L	-	-	-	-	M	-	L			
CO4	H	H	-	H	L	-	-	-	-	M	-	L			
CO5	H	H	L	M	L	-	-	-	-	M	-	L			
CO6	H	M	-	M	L	-	-	-	-	H	-	L			

UNIT I: Introduction: Network Types, LAN, MAN, WAN, Network Topologies Reference models- The OSI Reference Model- the TCP/IP Reference Model - A Comparison of the OSI and TCP/IP Reference Models, OSI Vs TCP/IP.
Physical Layer –Introduction to Guided Media- Twisted-pair cable, Coaxial cable and Fiber optic cable and introduction about unguided media.

UNIT II: Data link layer: Design issues, **Framing**: fixed size framing, variable size framing, flow control, error control, error detection and correction codes, CRC, Checksum: idea, one's complement internet checksum, services provided to Network Layer, **Elementary Data Link**

P. Srinivasarao

Layer protocols: simplex protocol, Simplex stop and wait, Simplex protocol for Noisy Channel.

Sliding window protocol: One bit, Go back N, Selective repeat-Stop and wait protocol, Data link layer in HDLC, Point to point protocol (PPP)

UNIT – III: Media Access Control: Random Access: ALOHA, Carrier sense multiple access (CSMA), CSMA with Collision Detection, CSMA with Collision Avoidance, **Controlled Access:** Reservation, Polling, Token Passing, **Channelization:** frequency division multiple Access(FDMA), time division multiple access(TDMA), code division multiple access(CDMA).

Wired LANs: Ethernet, Ethernet Protocol, Standard Ethernet, Fast Ethernet(100 Mbps), Gigabit Ethernet, 10 Gigabit Ethernet.

UNIT – IV: The Network Layer Design Issues – Store and Forward Packet Switching- Services Provided to the Transport layer- Implementation of Connectionless Service- Implementation of Connection Oriented Service- Comparison of Virtual Circuit and Datagram Networks,

Routing Algorithms-The Optimality principle-Shortest path, Flooding, Distance vector, Link state, Hierarchical, Congestion Control algorithms-General principles of congestion control, Congestion prevention polices, Approaches to Congestion Control-Traffic Aware Routing-Admission Control-Traffic Throttling-Load Shedding. Traffic Control Algorithm-Leaky bucket & Token bucket.

Internet Working: How networks differ- How networks can be connected- Tunnelling, internetwork routing-, Fragmentation, network layer in the internet – IP protocols-IP Version 4 protocol-IPV4 Header Format, IP addresses, Class full Addressing, CIDR, Subnets-IP Version 6-The main IPV6 header, Transition from IPV4 to IPV6, Comparison of IPV4 & IPV6.

UNIT –V: The Transport Layer: Transport layer protocols: Introduction-services- port number-User data gram protocol-User datagram-UDP services-UDP applications-Transmission control protocol: TCP services- TCP features- Segment- A TCP connection-windows in TCP- flow control-Error control, Congestion control in TCP.

Application Layer –- World Wide Web: HTTP, Electronic mail-Architecture- web based mail-email security- TELENET-local versus remote Logging-Domain Name System.

Text Books:

1. Computer Networksm, Andrew S Tanenbaum, Fifth Edition. Pearson Education/PHI 2. Data Communications and Networks, Behrouz A. Forouzan, Fifth Edition TMH.

References Books:

1. Data Communications and Networks- Achut S Godbole, AtulKahate
2. Computer Networks, Mayank Dave, CENGAGE

P. Srinivasarao

III Year I Semester	FORMAL LANGUAGES AND AUTOMATA THEORY	L	T	P	C
		3	0	0	3

Course Objectives:

- To learn fundamentals of Regular and Context Free Grammars and Languages
- To understand the relation between Regular Language and Finite Automata and machines
- To learn how to design Automata's and machines as Acceptors, Verifiers and Translators
- To understand the relation between Contexts free Languages, PDA and TM
- To learn how to design PDA as acceptor and TM as Calculators

Course Outcomes: *After successful completion of the course, student will be able to* CO1: Explain the foundational concepts of automata theory and construct various finite automata (DFA, NFA, ε–NFA, Mealy & Moore machines), including minimization and applications.

CO2: Apply regular expressions, regular grammars, pumping lemma, and closure properties to describe and analyze regular languages.

CO3: Develop and simplify context-free grammars, analyze derivations and parse trees, and convert grammars into normal forms while evaluating ambiguity.

CO4: Design pushdown automata and demonstrate their equivalence with context-free grammars for recognizing context-free languages.

CO5: Construct Turing Machines and analyze decidability, undecidability, PCP, the halting problem, and complexity classes (P, NP, NP-hard, NP-complete).

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H	M	M	-	-	-	-	-	-	L	-	L			
CO2	H	H	-	M	L	-	-	-	-	L	-	L			
CO3	H	H	M	M	L	-	-	-	-	L	-	L			
CO4	H	H	M	M	L	-	-	-	-	L	-	L			
CO5	H	H	L	H	L	-	-	-	-	L	-	L			

UNIT I

Finite Automata: Need of Automata theory, Central Concepts of Automata Theory, Automation, Finite Automation, Transition Systems, Acceptance of a String, DFA, Design of DFAs, NFA, Design of NFA, Equivalence of DFA and NFA, Conversion of NFA into DFA, Finite Automata with Є-Transitions, Minimization of Finite Automata, Finite Automata with output-Mealy and Moore Machines, Applications and Limitation of Finite Automata.

UNIT II

Regular Expressions, Regular Sets, Identity Rules, Equivalence of two RE, Manipulations of REs, Finite Automata and Regular Expressions, Inter Conversion, Equivalence between FA and RE, Pumping Lemma of Regular Sets, Closure Properties of Regular Sets, Grammars, Classification of Grammars, Chomsky Hierarchy Theorem, Right and Left Linear Regular Grammars, Equivalence between RG and FA, Inter Conversion.

P. Srinivasarao

UNIT III

Formal Languages, Context Free Grammar, Leftmost and Rightmost Derivations, Parse Trees, Ambiguous Grammars, Simplification of Context Free Grammars-Elimination of Useless Symbols, Є-Productions and Unit Productions, Normal Forms-Chomsky Normal Form and Greibach Normal Form, Pumping Lemma, Closure Properties, Applications of Context Free Grammars.

UNIT IV

Pushdown Automata, Definition, Model, Graphical Notation, Instantaneous Description, Language Acceptance of Pushdown Automata, Design of Pushdown Automata, Deterministic and Non – Deterministic Pushdown Automata, Equivalence of Pushdown Automata and Context Free Grammars, Conversion, Two Stack Pushdown Automata, Application of Pushdown Automata.

UNIT V

Turning Machine: Definition, Model, Representation of TMs-Instantaneous Descriptions, Transition Tables and Transition Diagrams, Language of a TM, Design of TMs, Types of TMs, Church's Thesis, Universal and Restricted TM, Decidable and Un-decidable Problems, Halting Problem of TMs, Post's Correspondence Problem, Modified PCP, Classes of P and NP, NP-Hard and NP-Complete Problems.

Text Books:

1. Introduction to Automata Theory, Languages and Computation, J. E. Hopcroft, R. Motwani and J. D. Ullman, 3rd Edition, Pearson, 2008

2. Theory of Computer Science-Automata, Languages and Computation, K. L. P. Mishra and N. Chandrasekharan, 3rd Edition, PHI, 2007

Reference Books:

1. Elements of Theory of Computation, Lewis H.P. & Papadimition C.H., Pearson /PHI
2. Theory of Computation, V. Kulkarni, Oxford University Press, 2013
3. Theory of Automata, Languages and Computation, Rajendra kumar, McGraw Hill, 2014

e- Resources:

1) https://nptel.ac.in/courses/106/104/106104028/

III Year I Semester	OBJECT ORIENTED ANALYSIS AND DESIGN	L	T	P	C
		3	0	0	3

Course Objectives: The main objective is the students to □
Become familiar with all phases of OOAD.

- Master the main features of the UML.
- Master the main concepts of Object Technologies and how to apply them at work and develop the ability to analyze and solve challenging problem in various domains.
- Learn the Object design Principles and understand how to apply them towards Implementation.

Course Outcomes: *After successful completion of the course, student will be able to* CO1: Explain the structure, attributes, and complexities of large software systems and apply principles for designing complex systems (K2)
CO2: Explain UML fundamentals, its conceptual model, modeling principles, and apply basic structural modeling using classes, relationships, and diagrams (K2)
CO3: Develop class and object diagrams by applying advanced structural modeling concepts such as interfaces, packages, roles, and relationships for diverse case studies (K3)
CO4: Construct behavioral models, including use case, interaction, and activity diagrams, to accurately represent system behavior for real-world applications (K3)
CO5: Design advanced behavioral and architectural models using state machines, events, components, and deployment diagrams for complex software systems (K4)

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H	M	L	-	-	-	-	-	-	-	-	L	M	L	-
CO2	H	M	M	-	L	-	-	-	-	-	-	L	M	L	-
CO3	H	M	H	L	M	-	-	-	-	L	-	L	H	L	L
CO4	M	M	H	M	M	-	-	-	-	L	-	M	M	M	L
CO5	M	H	H	M	H	-	-	-	-	L	-	M	H	M	M

UNIT I:

Introduction: The Structure of Complex systems, The Inherent Complexity of Software, Attributes of Complex System, Organized and Disorganized Complexity, Bringing Order to Chaos, Designing Complex Systems. **Case Study:** System Architecture: Satellite-Based Navigation

UNIT II:

Introduction to UML: Importance of modeling, principles of modeling, object oriented modeling, conceptual model of the UML, Architecture, and Software Development Life Cycle. **Basic Structural Modeling:** Classes, Relationships, common Mechanisms, and diagrams. **Case Study:** Control System: Traffic Management.

UNIT III:

P. Srinivasarao

Class & Object Diagrams: Terms, concepts, modeling techniques for Class & Object Diagrams. **Advanced Structural Modeling:** Advanced classes, advanced relationships, Interfaces, Types and Roles, Packages. **Case Study:** AI: Cryptanalysis.

UNIT IV:

Basic Behavioral Modeling-I: Interactions, Interaction diagrams Use cases, Use case Diagrams, Activity Diagrams. **Case Study:** Web Application: Vacation Tracking System

UNIT V:

Advanced Behavioral Modeling: Events and signals, state machines, processes and Threads, time and space, state chart diagrams. **Architectural Modeling:** Component, Deployment, Component diagrams and Deployment diagrams. **Case Study:** Weather Forecasting

Text Books:

1. Grady BOOCH, Robert A. Maksimchuk, Michael W. ENGLE, Bobbi J. Young, Jim Conallen, Kellia Houston , "Object- Oriented Analysis and Design with Applications", 3rd edition, 2013, PEARSON.
2. Grady Booch, James Rumbaugh, Ivar Jacobson: The Unified Modeling Language User Guide, Pearson Education.

Reference Books:

1. Meilir Page-Jones: Fundamentals of Object Oriented Design in UML, Pearson Education.
2. Pascal Roques: Modeling Software Systems Using UML2, WILEY- Dreamtech India Pvt. Ltd.
3. Atul Kahate: Object Oriented Analysis & Design, The McGraw-Hill Companies.
4. Appling UML and Patterns: An introduction to Object – Oriented Analysis and Design and Unified Process, Craig Larman, Pearson Education.

III Year I Semester	ARTIFICIAL INTELLIGENCE	L	T	P	C
		3	0	0	3

Pre-requisite:

1. Knowledge in Computer Programming.
2. A course on "Mathematical Foundations of Computer Science".
3. Background in linear algebra, data structures and algorithms, and probability.

Course Objectives:

1. The student should be made to study the concepts of Artificial Intelligence.
2. The student should be made to learn the methods of solving problems using Artificial Intelligence.
3. The student should be made to introduce the concepts of Expert Systems.
4. To understand the applications of AI, namely game playing, theorem proving, and machine learning.
5. To learn different knowledge representation techniques

Course Outcomes: *After successful completion of the course, student will be able to* CO1: Describe AI problems, foundations of AI, and agent architectures, and analyze how agents perceive, act, and make rational decisions in different environments (K2)
CO2: Apply uninformed and heuristic search strategies—including BFS, DFS, Hill Climbing, A*, AO*, and game-playing algorithms—to solve well-formulated problems (K3) CO3: Develop suitable knowledge representation schemes using predicate logic, semantic networks, frames, rules, and probabilistic models to support reasoning under uncertainty (K3) CO4: Perform inference using first-order logic, unification, resolution, forward/backward chaining, and apply learning approaches such as decision trees, EBL, statistical methods, and reinforcement learning (K4)
CO5: Examine the architecture, components, knowledge acquisition techniques, and shells of expert systems, and evaluate typical expert system applications (K4)

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H	M	L	-	-	-	-	-	-	-	-	L	-	-	-
CO2	H	H	H	M	L	M	-	-	-	-	-	L	-	-	-
CO3	H	M	H	M	M	M	-	-	-	L	-	L	-	-	-
CO4	H	H	H	H	M	M	-	-	-	L	-	M	-	-	-
CO5	M	M	M	M	H	L	-	-	-	L	-	M	-	-	-

UNIT - I

Introduction: 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.

Sk. Rasheed P. Srinivasarao

UNIT - II

Searching- Searching for solutions, uniformed search strategies – Breadth first search, depth first Search. Search with partial information (Heuristic search) Hill climbing, A* ,AO* Algorithms, Problem reduction, Game Playing-Adversial search, Games, mini-max algorithm, optimal decisions in multiplayer games, Problem in Game playing, Alpha-Beta pruning, Evaluation functions.

UNIT - III

Representation of Knowledge: Knowledge representation issues, predicate logic- logic programming, semantic nets- frames and inheritance, constraint propagation, representing knowledge using rules, rules based deduction systems. Reasoning under uncertainty, review of probability, Bayes' probabilistic interferences and dempstershafer theory.

UNIT - IV

Logic concepts: First order logic. Inference in first order logic, propositional vs. first order inference, unification & lifts forward chaining, Backward chaining, Resolution, Learning from observation Inductive learning, Decision trees, Explanation based learning, Statistical Learning methods, Reinforcement Learning.

UNIT - V

Expert Systems: Architecture of expert systems, Roles of expert systems – Knowledge Acquisition Meta knowledge Heuristics. Typical expert systems – MYCIN, DART, XCON: Expert systems shells.

Textbooks:

1. S. Russel and P. Norvig, "Artificial Intelligence – A Modern Approach", SecondEdition, Pearson Education.
2. Kevin Night and Elaine Rich, Nair B., "Artificial Intelligence (SIE)", Mc Graw Hill

Reference Books:

1. David Poole, Alan Mackworth, Randy Goebel,"Computational Intelligence: a logical approach", Oxford University Press.
2. G. Luger, "Artificial Intelligence: Structures and Strategies for complex problemsolving", Fourth Edition, Pearson Education.
3. J. Nilsson, "Artificial Intelligence: A new Synthesis", Elsevier Publishers.
4. Artificial Intelligence, SarojKaushik, CENGAGE Learning.

Online Learning Resources:

1. https://ai.google/
2. https://swayam.gov.in/nd1_noc19_me71/preview

Sk. Rasheed P. Srinivasarao

III Year I Semester	MICROPROCESSORS & MICROCONTROLLERS	L	T	P	C
		3	0	0	3

Course Objectives:

- To introduce fundamental architectural concepts of microprocessors and microcontrollers.
- To impart knowledge on addressing modes and instruction set of 8086 and 8051
- To introduce assembly language programming concepts To explain memory and I/O interfacing with 8086 and 8051
- To introduce16 bit and 32 bit microcontrollers.

Course Outcomes: *After successful completion of the course, student will be able to*

CO1: Explain the architecture, operational modes, timing, and interrupt mechanisms of the 8086 microprocessor (K2)

CO2: Apply addressing modes, instruction set, and assembler directives to develop assembly-level programs for the 8086 microprocessor (K3)

CO3: Design interfacing circuits for memories, I/O devices, interrupts, serial communication, and peripheral controllers commonly used with 8086 (K4)

CO4: Describe the architecture, SFRs, instruction set, and programming model of the 8051 microcontroller and develop basic assembly programs (K3)

CO5: Develop interface routines for timers, serial communication, interrupts, sensors, ADC/DAC, and external devices using 8051, and compare microprocessor/microcontroller families (K3)

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H	M	L	-	-	-	-	-	-	L	-	-	-	-	-
CO2	H	M	H	M	M	-	-	-	-	L	-	-	-	-	-
CO3	H	H	H	H	H	-	-	-	-	L	-	-	-	-	-
CO4	H	M	M	L	M	-	-	-	-	L	-	-	-	-	-
CO5	H	H	H	M	H	-	-	-	-	M	-	-	-	-	-

UNIT I:

8086 Architecture: Main features, pin diagram/description, 8086 microprocessor family, internal architecture, bus interfacing unit, execution unit, interrupts and interrupt response, 8086 system timing, minimum mode and maximum mode configuration.

UNIT II:

8086 Programming: Program development steps, instructions, addressing modes, assembler directives, writing simple programs with an assembler, assembly language program development tools.

UNIT III:

8086 Interfacing: Semiconductor memories interfacing (RAM, ROM), Intel 8255 programmable peripheral interface, Interfacing switches and LEDS, Interfacing seven segment

displays, software and hardware interrupt applications, Intel 8251 USART architecture and interfacing, Intel 8237a DMA controller, stepper motor, A/D and D/A converters, Need for 8259 programmable interrupt controllers.

UNIT IV:

Microcontroller, Architecture of 8051, Special Function Registers(SFRs), I/O Pins Ports and Circuits, Instruction set, Addressing modes, Assembly language programming.

UNIT V:

Interfacing Microcontroller, Programming 8051 Timers, Serial Port Programming, Interrupts Programming, LCD & Keyboard Interfacing, ADC, DAC & Sensor Interfacing, External Memory Interface, Stepper Motor and Waveform generation, Comparison of Microprocessor, Microcontroller, PIC and ARM processors

Textbooks:

1. Microprocessors and Interfacing – Programming and Hardware by Douglas V Hall, SSSP Rao, Tata McGraw Hill Education Private Limited, 3rd Edition,1994.
2. K M Bhurchandi, A K Ray, Advanced Microprocessors and Peripherals, 3rd edition, McGraw Hill Education, 2017.
3. Raj Kamal, Microcontrollers: Architecture, Programming, Interfacing and System Design, 2nd edition, Pearson, 2012.

Reference Books:

1. Ramesh S Gaonkar, Microprocessor Architecture Programming and Applications with the 8085, 6th edition, Penram International Publishing, 2013.
2. Kenneth J. Ayala, The 8051 Microcontroller, 3rd edition, Cengage Learning, 2004.

III Year I Semester	QUANTUM COMPUTING	L	T	P	C
		3	0	0	3

Pre-requisites:
Basic knowledge of Linear Algebra (vectors, matrices, eigenvalues, operations)
Fundamental concepts of Probability & Statistics
Basic understanding of Classical Computing (binary operations, logic gates, algorithmic thinking)
Introductory Physics knowledge (wave-particle concepts, basic mechanics/electromagnetics)
Basic Foundations of Programming (ability to understand algorithms and computational models)

Course Objectives:
To introduce the fundamentals of quantum computing, the problem-solving approach using finite dimensional mathematics

Course Outcomes: *After successful completion of the course, student will be able to* CO1: Explain the evolution of quantum computing and distinguish between classical and quantum computing models, including bits vs qubits and logical operations (K2)
CO2: Apply foundational mathematics, physics, and biological concepts—including linear algebra, Hilbert spaces, superposition, entanglement, and genomics/proteomics—to analyze quantum computational principles (K3)
CO3: Demonstrate the representation, physical realization, and manipulation of qubits, and construct simple quantum circuits using single and multi-qubit gates (K3)
CO4: Analyze and implement foundational quantum algorithms such as Deutsch, Deutsch–Jozsa, Grover, and Shor, and compare their performance with classical algorithms (K4) CO5: Examine noise sources, apply quantum error-correction and fault-tolerant mechanisms, and interpret modern applications of quantum information and cryptography (K4)

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H	M	L	-	-	-	-	-	-	L	-	L	-	-	-
CO2	H	H	H	M	L	M	-	-	-	L	-	L	-	-	-
CO3	H	H	H	M	M	M	-	-	-	L	-	L	-	-	-
CO4	H	H	H	H	M	M	-	-	-	M	-	M	-	-	-
CO5	M	H	M	M	L	L	-	-	-	M	-	M	-	-	-

UNIT - I
History of Quantum Computing: Importance of Mathematics, Physics and Biology. Introduction to Quantum Computing: Bits Vs Qubits, Classical Vs Quantum logical operations

UNIT - II
Background Mathematics: Basics of Linear Algebra, Hilbert space, Probabilities and measurements.

Background Physics: Paul's exclusion Principle, Superposition, Entanglement and super-symmetry, density operators and correlation, basics of quantum mechanics, Measurements in bases other than computational basis. Background Biology: Basic concepts of Genomics and Proteomics (Central Dogma)

UNIT - III

Qubit: Physical implementations of Qubit. Qubit as a quantum unit of information. The Bloch sphere Quantum Circuits: single qubit gates, multiple qubit gates, designing the quantum circuits. Bell states.

UNIT - IV

Quantum Algorithms: Classical computation on quantum computers. Relationship between quantum and classical complexity classes. Deutsch's algorithm, Deutsch's-Jozsa algorithm, Shor's factorization algorithm, Grover's search algorithm.

UNIT - V

Noise and error correction: Graph states and codes, Quantum error correction, fault-tolerant computation. Quantum Information and Cryptography: Comparison between classical and quantum information theory. Quantum Cryptography, Quantum teleportation

Text Books:

1. Quantum Computation and Quantum Information, Nielsen M. A.,Cambridge
2. Programming Quantum Computers, Essential Algorithms and Code Samples, Eric R Johnson, Nic Harrigan, Mercedes Ginemo, Segovia, Oreilly

Reference Books:

1. Quantum Computing for Computer Scientists, Noson S. Yanofsk, Mirco A. Mannucci
2. Principles of Quantum Computation and Information,Benenti G., Casati G. and Strini G., Vol.I: Basic Concepts, Vol II
3. Basic Tools and Special Topics, World Scientific. Pittenger A. O., An Introduction to Quantum Computing Algorithms

P. Srinivasarao

III Year I Semester	DATA MINING LAB	L	T	P	C
		0	0	3	1.5

Pre-requisites: Data Base Management Systems, Python Programming

Course Objectives: The main objective of the course is to

- Inculcate Conceptual, Logical, and Physical design of Data Warehouses OLAP applications and OLAP deployment
- Design a data warehouse or data mart to present information needed by management in a form that is usable
- Emphasize hands-on experience working with all real data sets.
- Test real data sets using popular data mining tools such as WEKA, Python Libraries □ Develop ability to design various algorithms based on data mining tools.

Course Outcomes: *After successful completion of the course, student will be able to*

CO1: Construct data warehouses, design multidimensional models (Star/Snowflake/Fact Constellation), and perform ETL and OLAP operations using industrial tools (K3)

CO2: Explore and operate WEKA tools to load, preprocess, visualize datasets, and analyze attribute characteristics for machine learning tasks (K3)

CO3: Apply data preprocessing, association rule mining (Apriori/FP-Growth), and interpretation of rules to derive meaningful insights from datasets (K4)

CO4: Implement and evaluate classification and clustering algorithms (ID3, J48, NB, k-NN, k-means) using WEKA/Python/R, and compare model performance using ROC curves, entropy, SSE, and confusion matrices (K4)

CO5: Develop machine learning programs in Python/Java/R for association mining, Naive Bayes, clustering, chi-square computation, and visualization using Matplotlib, and interpret results (K4)

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H	M	H	M	H	-	-	-	-	L	-	-	-	-	-
CO2	H	M	M	M	H	-	-	-	-	L	-	-	-	-	-
CO3	H	H	H	H	H	-	-	-	-	L	-	-	-	-	-
CO4	H	H	H	H	H	-	-	-	L	M	-	-	-	-	-
CO5	H	H	H	H	H	-	-	-	L	M	-	-	-	-	-

Software Requirements: WEKA Tool/Python/R-Tool/Rapid Tool/Oracle Data mining **List of Experiments:**

1. Creation of a Data Warehouse.
   - Build Data Warehouse/Data Mart (using open source tools like Pentaho Data Integration Tool, Pentaho Business Analytics; or other data warehouse tools like Microsoft-SSIS, Informatica, Business Objects, etc.,)

P. Srinivasarao

- Design multi-dimensional data models namely Star, Snowflake and Fact Constellation schemas for any one enterprise (ex. Banking, Insurance, Finance, Healthcare, manufacturing, Automobiles, sales etc).
- Write ETL scripts and implement using data warehouse tools.
- Perform Various OLAP operations such slice, dice, roll up, drill up and pivot

2. Explore machine learning tool "WEKA"
   - Explore WEKA Data Mining/Machine Learning Toolkit.
   - Downloading and/or installation of WEKA data mining toolkit.
   - Understand the features of WEKA toolkit such as Explorer, Knowledge Flow interface, Experimenter, command-line interface.
   - Navigate the options available in the WEKA (ex. Select attributes panel, Preprocess panel, Classify panel, Cluster panel, Associate panel and Visualize panel)
   - Study the arff file format Explore the available data sets in WEKA. Load a data set (ex. Weather dataset, Iris dataset, etc.)
   - Load each dataset and observe the following:
     1. List the attribute names and they types
     2. Number of records in each dataset
     3. Identify the class attribute (if any)
     4. Plot Histogram
     5. Determine the number of records for each class.
     6. Visualize the data in various dimensions

3. Perform data preprocessing tasks and Demonstrate performing association rule mining on data sets
   - Explore various options available in Weka for preprocessing data and apply Unsupervised filters like Discretization, Resample filter, etc. on each dataset
   - Load weather. nominal, Iris, Glass datasets into Weka and run Apriori Algorithm with different support and confidence values.
   - Study the rules generated. Apply different discretization filters on numerical attributes and run the Apriori association rule algorithm. Study the rules generated.
   - Derive interesting insights and observe the effect of discretization in the rule generation process.

4. Demonstrate performing classification on data sets Weka/R
   - Load each dataset and run 1d3, J48 classification algorithm. Study the classifier output. Compute entropy values, Kappa statistic.
   - Extract if-then rules from the decision tree generated by the classifier, Observe the confusion matrix.
   - Load each dataset into Weka/R and perform Naïve-bayes classification and k-Nearest Neighbour classification. Interpret the results obtained.
   - Plot RoC Curves

P. Srinivasarao

- Compare classification results of ID3, J48, Naïve-Bayes and k-NN classifiers for each dataset, and deduce which classifier is performing best and poor for each dataset and justify.

5. Demonstrate performing clustering of data sets
   - Load each dataset into Weka/R and run simple k-means clustering algorithm with different values of k (number of desired clusters).
   - Study the clusters formed. Observe the sum of squared errors and centroids, and derive insights.
   - Explore other clustering techniques available in Weka/R.
   - Explore visualization features of Weka/R to visualize the clusters. Derive interesting insights and explain.

6. Demonstrate knowledge flow application on data sets into Weka/R
   - Develop a knowledge flow layout for finding strong association rules by using Apriori, FP Growth algorithms
   - Set up the knowledge flow to load an ARFF (batch mode) and perform a cross validation using J48 algorithm
   - Demonstrate plotting multiple ROC curves in the same plot window by using j48 and Random forest tree
7. Demonstrate ZeroR technique on Iris dataset (by using necessary preprocessing technique(s)) and share your observations
8. Write a java program to prepare a simulated data set with unique instances.
9. Write a Python program to generate frequent item sets / association rules using Apriori algorithm
10. Write a program to calculate chi-square value using Python/R. Report your observation.
11. Write a program of Naive Bayesian classification using Python/R programming language.
12. Implement a Java/R program to perform Apriori algorithm
13. Write a R program to cluster your choice of data using simple k-means algorithm using JDK
14. Write a program of cluster analysis using simple k-means algorithm Python/R programming language.
15. Write a program to compute/display dissimilarity matrix (for your own dataset containing at least four instances with two attributes) using Python
16. Visualize the datasets using matplotlib in python/R.(Histogram, Box plot, Bar chart, Pie chart etc.,)

III Year I Semester	COMPUTER NETWORKS LAB	L	T	P	C
		0	0	3	1.5

Course Objectives:

Learn basic concepts of computer networking and acquire practical notions of protocols with the emphasis on TCP/IP. A lab provides a practical approach to Ethernet/Internet networking: networks are assembled, and experiments are made to understand the layered architecture and how do some important protocols work

Course Outcomes: *After successful completion of the course, student will be able to* CO1: Configure and operate basic network devices to establish Local Area Networks and analyze packet-level behavior using tools such as Wireshark (K4)

CO2: Implement data link layer framing, error detection and correction techniques such as bit/character stuffing, checksum, Hamming code, and CRC algorithms (K3)

CO3: Develop and simulate reliable data transfer protocols including Stop-and-Wait, Go-Back-N, and Selective Repeat using programming constructs (K3)

CO4: Implement routing algorithms such as Dijkstra's shortest path and Distance Vector routing to compute optimal routing paths (K4)

CO5: Simulate congestion control, packet flow behavior, and throughput using NS2/Nmap and evaluate network performance under various scenarios (K5)

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	M	M	L	M	H	-	-	-	-	L	-	M	-	-	-
CO2	H	M	L	M	L	-	-	-	-	L	-	M	-	-	-
CO3	M	M	M	M	L	-	-	-	-	L	-	M	-	-	-
CO4	H	H	M	H	L	-	-	-	-	L	-	M	-	-	-
CO5	M	H	M	H	H	-	-	-	-	L	-	H	-	-	-

List of Experiments:

1. Study of Network devices in detail and connect the computers in Local Area Network.
2. Write a Program to implement the data link layer farming methods such as
   i) Character stuffing ii) bit stuffing.
3. Write a Program to implement data link layer farming method checksum.
4. Write a program for Hamming Code generation for error detection and correction.
5. Write a Program to implement on a data set of characters the three CRC polynomials – CRC 12, CRC 16 and CRC CCIP.
6. Write a Program to implement Sliding window protocol for Goback N.
7. Write a Program to implement Sliding window protocol for Selective repeat.
8. Write a Program to implement Stop and Wait Protocol.
9. Write a program for congestion control using leaky bucket algorithm
10. Write a Program to implement Dijkstra's algorithm to compute the Shortest path through a graph.

P. Srinivasarao

11. Write a Program to implement Distance vector routing algorithm by obtaining routing table at each node (Take an example subnet graph with weights indicating delay between nodes).
12. Write a Program to implement Broadcast tree by taking subnet of hosts.
13. Wireshark
    i) Packet Capture Using Wire shark
    ii) Starting Wire shark iii. Viewing Captured Traffic iv. Analysis and Statistics & Filters.
14. How to run Nmap scan
15. Operating System Detection using Nmap
16. Do the following using NS2 Simulator
    i) NS2 Simulator-Introduction ii. Simulate to Find the Number of Packets Dropped iii. Simulate to Find the Number of Packets Dropped by TCP/UDP iv. Simulate to Find the Number of Packets Dropped due to Congestion v. Simulate to Compare Data Rate& Throughput.

III Year I Semester	FULL STACK DEVELOPMENT - 2	L	T	P	C
		0	1	2	2

Course Objectives:

The main objectives of the course are to

- Make use of router, template engine and authentication using sessions to develop application in Express JS.
- Build a single page application using RESTful APIs in Express JS
- Apply router and hooks in designing React JS application
- Make use of MongoDB queries to perform CRUD operations on document database

Course Outcomes: *After successful completion of the course, student will be able to* CO1: Build server-side applications using ExpressJS with routing, middleware, templating, sessions, authentication, and RESTful APIs (K3)

CO2: Develop interactive client-side applications using ReactJS components, props, states, JSX, routing, forms, hooks, and event handling (K4)

CO3: Design and integrate MERN-based full-stack applications using ExpressJS, ReactJS, and MongoDB for end-to-end data management (K5)

CO4: Perform CRUD operations and manage collections, indexes, and aggregations using MongoDB and Mongoose (K3)

CO5: Implement and test real-world applications (e.g., To-Do, Quiz App) and evaluate security, performance, and usability considerations (K5)

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	M	M	M	L	H	-	-	-	-	L	-	M	-	-	-
CO2	M	H	M	M	H	-	-	-	-	M	-	M	-	-	-
CO3	H	H	H	M	H	-	-	-	-	M	-	H	-	-	-
CO4	M	M	M	M	H	-	-	-	-	L	-	M	-	-	-
CO5	M	H	H	H	H	-	-	-	-	M	-	H	-	-	-

Experiments covering the Topics:

- Express JS – Routing, HTTP Methods, Middleware, Templating, Form Data
- Express JS – Cookies, Sessions, Authentication, Database, RESTful APIs
- React JS – Render HTML, JSX, Components – function & Class, Props and States, Styles, Respond to Events
- React JS – Conditional Rendering, Rendering Lists, React Forms, React Router, Updating the Screen
- React JS – Hooks, Sharing data between Components, Applications – To-do list and Quiz
- MongoDB – Installation, Configuration, CRUD operations, Databases, Collections and Records

Sample Experiments:

1. Express JS – Routing, HTTP Methods, Middleware.

a. Write a program to define a route, Handling Routes, Route Parameters, Query Parameters and URL building.
b. Write a program to accept data, retrieve data and delete a specified resource using http methods.
c. Write a program to show the working of middleware.

2. Express JS – Templating, Form Data

a. Write a program using templating engine.
b. Write a program to work with form data.

3. Express JS – Cookies, Sessions, Authentication

a. Write a program for session management using cookies and sessions.
b. Write a program for user authentication.

4. Express JS – Database, RESTful APIs

a. Write a program to connect MongoDB database using Mangoose and perform CRUD operations.
b. Write a program to develop a single page application using RESTful APIs.

5. ReactJS – Render HTML, JSX, Components – function & Class

a. Write a program to render HTML to a web page.
b. Write a program for writing markup with JSX.
c. Write a program for creating and nesting components (function and class)

6. ReactJS – Props and States, Styles, Respond to Events

a. Write a program to work with props and states.
b. Write a program to add styles (CSS & Sass Styling) and display data.
c. Write a program for responding to events.

7. ReactJS – Conditional Rendering, Rendering Lists, React Forms

a. Write a program for conditional rendering.
b. Write a program for rendering lists.
c. Write a program for working with different form fields using react forms.

8. ReactJS – React Router, Updating the Screen

a. Write a program for routing to different pages using react router.
b. Write a program for updating the screen.

9. ReactJS – Hooks, Sharing data between Components

a. Write a program to understand the importance of using hooks.

P. Srinivasarao

b. Write a program for sharing data between components.

10. MongoDB – Installation, Configuration, CRUD operations

a. Install MongoDB and configure ATLAS

b. Write MongoDB queries to perform CRUD operations on document using insert(), find(), update(), remove()

11. MongoDB – Databases, Collections and Records

a. Write MongoDB queries to Create and drop databases and collections.

b. Write MongoDB queries to work with records using find(), limit(), sort(), createIndex(), aggregate().

12. Augmented Programs: (Any 2 must be completed)

a. Design a to-do list application using NodeJS and ExpressJS.

b. Design a Quiz app using ReactJS.

c. Complete the MongoDB certification from MongoDB University website.

Text Books:

1. Pro MERN Stack: Full Stack Web App Development with Mongo, Express, React, and Node, Vasan Subramanian, 2nd edition, APress, O'Reilly.
2. Node.Js in Action, Mike Cantelon, Mark Harter, T.J. Holowaychuk, Nathan Rajlich,Manning Publications. (Chapters 1-11)
3. React Quickly, AzatMardan,Manning Publications(Chapters 1-8,12-14)

Web Links:

1. ExpressJS - https://www.tutorialspoint.com/expressjs
2. ReactJS - https://www.w3schools.com/REACT (and) https://react.dev/learn#
3. MongoDB - https://learn.mongodb.com/learning-paths/introduction-to-mongodb

III Year I Semester	USER INTERFACE DESIGN USING FLUTTER	L	T	P	C
		0	0	2	1

Course Objectives:

- Learns to Implement Flutter Widgets and Layouts
- Understands Responsive UI Design and with Navigation in Flutter
- Knowledge on Widges and customize widgets for specific UI elements, Themes Understand to include animation apart from fetching data

Course Outcomes: *After successful completion of the course, student will be able to*

CO1: Apply Dart programming constructs and core Flutter widgets to build basic user interfaces (K3)

CO2: Develop responsive and multi-screen mobile applications using layouts, navigation, and routing techniques (K4)

CO3: Implement effective state management and custom widget design using Flutter frameworks such as setState and Provider (K4)

CO4: Integrate animations, form handling, validation, and REST API communication to build interactive and data-driven mobile applications (K5)

CO5: Perform debugging, testing, and performance optimization of Flutter applications using available tools, testing frameworks, and best practices (K5)

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	M	L	M	L	H	-	-	-	-	L	-	M	-	-	-
CO2	M	M	H	L	H	-	-	-	-	M	-	M	-	-	-
CO3	M	H	H	L	H	-	-	-	-	M	-	H	-	-	-
CO4	M	H	H	M	H	-	-	-	-	M	-	H	-	-	-
CO5	M	H	M	H	H	-	-	-	-	M	-	H	-	-	-

List of Experiments:

Students need to implement the following experiments

1. a) Install Flutter and Dart SDK.
   b) Write a simple Dart program to understand the language basics.
2. a) Explore various Flutter widgets (Text, Image, Container, etc.).
   b) Implement different layout structures using Row, Column, and Stack widgets.
3. a) Design a responsive UI that adapts to different screen sizes.
   b) Implement media queries and breakpoints for responsiveness.
4. a) Set up navigation between different screens using Navigator.
   b) Implement navigation with named routes.
5. a) Learn about stateful and stateless widgets.
   b) Implement state management using set State and Provider.
6. a) Create custom widgets for specific UI elements.
   b) Apply styling using themes and custom styles.

7. a) Design a form with various input fields.
   b) Implement form validation and error handling.
8. a) Add animations to UI elements using Flutter's animation framework.
   b) Experiment with different types of animations (fade, slide, etc.).
9. a) Fetch data from a REST API.
   b) Display the fetched data in a meaningful way in the UI.
10. a) Write unit tests for UI components.
    b) Use Flutter's debugging tools to identify and fix issues.

Text Books:

1. Marco L. Napoli, Beginning Flutter: A Hands-on Guide to App Development.
2. Rap Payne, Beginning App Development with Flutter: Create Cross-Platform Mobile Apps 1st Edition, Apres
3. Richard Rose, Flutter & Dart Cookbook, Developing Full stack Applications for the Cloud, Oreilly.

P. Srinivasarao

III Year II Semester	COMPILER DESIGN	L	T	P	C
		3	0	0	3

Course Objectives:

Understand the basic concept of compiler design, and its different phases which will be helpful to construct new tools like LEX, YACC, etc.

Course Outcomes: *After successful completion of the course, student will be able to* CO1: Explain lexical and syntax analysis mechanisms, token specification, and grammar properties used in compiler front-end design (K2)

CO2: Apply top-down and bottom-up parsing techniques to analyze context-free grammars and construct predictive and LR parsing tables (K3)

CO3: Develop syntax-directed definitions, translation schemes, and intermediate code representations such as TAC, syntax trees, and backpatching (K4)

CO4: Examine code optimization techniques, data-flow analysis, and loop optimizations to improve intermediate code efficiency (K4)

CO5: Construct runtime environments, activation records, and generate object code by addressing issues in code generation and register allocation (K5)

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H	M	L	L	L	-	-	-	-	L	-	M	-	-	-
CO2	H	H	M	L	M	-	-	-	-	L	-	M	-	-	-
CO3	H	H	H	M	H	-	-	-	-	L	-	H	-	-	-
CO4	M	H	M	H	H	-	-	-	-	L	-	H	-	-	-
CO5	H	H	H	M	H	-	-	-	-	L	-	H	-	-	-

UNIT I:

Lexical Analysis: Language Processors, Structure of a Compiler, Lexical Analysis, The Role of the Lexical Analyzer, Bootstrapping, Input Buffering, Specification of Tokens, Recognition of Tokens, Lexical Analyzer Generator-LEX, Finite Automata, Regular Expressions and Finite Automata, Design of a Lexical Analyzer Generator.

Syntax Analysis: The Role of the Parser, Context-Free Grammars, Derivations, Parse Trees, Ambiguity, Left Recursion, Left Factoring,

UNIT II:

Top Down Parsing: Pre Processing Steps of Top Down Parsing, Backtracking, Recursive Descent Parsing, LL (1) Grammars, Non-recursive Predictive Parsing, Error Recovery in Predictive Parsing.

Bottom Up Parsing: Introduction, Difference between LR and LL Parsers, Types of LR Parsers, Shift Reduce Parsing, SLR Parsers, Construction of SLR Parsing Tables, More Powerful LR Parses, Construction of CLR (1) and LALR Parsing Tables, Dangling Else Ambiguity, Error Recovery in LR Parsing, Handling Ambiguity Grammar with LR Parsers.

UNIT III:

Syntax Directed Translation: Syntax-Directed Definitions, Evaluation Orders for SDD's, Applications of Syntax Directed Translation, Syntax-Directed Translation Schemes, Implementing L-Attributed SDD's. **Intermediate Code Generation:** Variants of Syntax Trees, Three Address Code, Types and Declarations, Translation of Expressions, Type Checking, Control Flow, Backpatching, Intermediate Code for Procedures.

UNIT IV:

Code Optimization: The Principle Sources of Optimization, Basic Blocks, Optimization of Basic Blocks, Structure Preserving Transformations, Flow Graphs, Loop Optimization, Data-Flow Analysis, Peephole Optimization

UNIT V:

Run Time Environments: Storage Organization, Run Time Storage Allocation, Activation Records, Procedure Calls, Displays

Code Generation: Issues in the Design of a Code Generator, Object Code Forms, Code Generation Algorithm, Register Allocation and Assignment.

Text Books:

1.Compilers: Principles, Techniques and Tools, Second Edition, Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffry D. Ullman, Pearson, 2007.

Reference Books:

1. Compiler Construction, Principles and Practice, Kenneth C Louden, Cengage Learning, 2006
2. Modern compiler implementation in C, Andrew W Appel, Revised edition, Cambridge University Press.
3. Optimizing Compilers for Modern Architectures, Randy Allen, Ken Kennedy, Morgan Kauffmann, 2001.
4. Levine, J.R., T. Mason and D. Brown, Lex and Yacc, edition, O'Reilly & Associates, 1990

P. Srinivasarao

III Year II Semester	CLOUD COMPUTING	L	T	P	C
		3	0	0	3

Course Objectives:

- To explain the evolving utility computing model called cloud computing.
- To introduce the various levels of services offered by cloud.
- To discuss the fundamentals of cloud enabling technologies such as distributed computing, service-oriented architecture and virtualization.
- To emphasize the security and other challenges in cloud computing.
- To introduce the advanced concepts such as containers, serverless computing and cloud-centric Internet of Things.

Course Outcomes: *After successful completion of the course, student will be able to* CO1: Explain cloud computing concepts, service models, deployment models, and associated benefits in modern distributed systems (K2)

CO2: Apply cloud-enabling technologies such as parallel computing, distributed systems, SOA, and virtualization in cloud environments (K3)

CO3: Analyze virtualization techniques, container technologies, orchestration tools, and public cloud VM/container services (K4)

CO4: Evaluate cloud computing challenges related to interoperability, scalability, energy efficiency, and cloud security architecture (K5)

CO5: Assess advanced cloud paradigms including serverless computing, cloud-centric IoT, edge/fog computing, DevOps, and quantum cloud technologies (K5)

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	H	M	L	L	L	-	L	-	-	L	-	M	-	-	-
CO2	H	H	M	M	M	-	L	-	-	L	-	M	-	-	-
CO3	H	H	H	M	H	-	L	-	-	L	-	H	-	-	-
CO4	H	H	M	H	H	-	M	L	-	L	-	H	-	-	-
CO5	H	H	M	H	H	-	M	L	-	L	-	H	-	-	-

UNIT -I: Introduction to Cloud Computing Fundamentals

Cloud computing at a glance, defining a cloud, cloud computing reference model, types of services (IaaS, PaaS, SaaS), cloud deployment models (public, private, hybrid), utility computing, cloud computing characteristics and benefits, cloud service providers (Amazon Web Services, Microsoft Azure, Google AppEngine).

UNIT-II: Cloud Enabling Technologies

Ubiquitous Internet, parallel and distributed computing, elements of parallel computing, hardware architectures for parallel computing (SISD, SIMD, MISD, MIMD), elements of distributed computing, Inter-process communication, technologies for distributed computing, remote procedure calls (RPC), service-oriented architecture (SOA), Web services, virtualization.

P. Srinivasarao

UNIT-III: Virtualization and Containers Characteristics of virtualized environments, taxonomy of virtualization techniques, virtualization and cloud Computing, pros and cons of virtualization, technology examples (XEN, VMware), building blocks of containers, container platforms (LXC, Docker), container orchestration, Docker Swarm and Kubernetes, public cloud VM (e.g. Amazon EC2) and container (e.g. Amazon Elastic Container Service) offerings.

UNIT-IV: Cloud computing challenges
Economics of the cloud, cloud interoperability and standards, scalability and fault tolerance, energy efficiency in clouds, federated clouds, cloud computing security, fundamentals of computer security, cloud security architecture, cloud shared responsibility model, security in cloud deployment models.

UNIT -V: Advanced concepts in cloud computing
Serverless computing, Function-as-a-Service, serverless computing architecture, public cloud (e.g. AWS Lambda) and open-source (e.g. OpenFaaS) serverless platforms, Internet of Things (IoT), applications, cloud-centric IoT and layers, edge and fog computing, DevOps, infrastructure-as-code, quantum cloud computing.

Text Books:

1. Mastering Cloud Computing, 2nd edition, Rajkumar Buyya, Christian Vecchiola, Thamarai Selvi, Shivananda Poojara, Satish N. Srirama, Mc Graw Hill, 2024.
2. Distributed and Cloud Computing, Kai Hwang, Geoffery C. Fox, Jack J. Dongarra, Elsevier, 2012.

Reference Books:

1. Cloud Computing, Theory and Practice, Dan C Marinescu, 2nd edition, MK Elsevier, 2018.
2. Essentials of cloud Computing, K. Chandrasekhran, CRC press, 2014.
3. Online documentation and tutorials from cloud service providers (e.g., AWS, Azure, GCP)

III Year II Semester	CRYPTOGRAPHY & NETWORK SECURITY (Common to CSE, CS & IT)	L	T	P	C
		3	0	0	3

Course Objectives:

The main objectives of this course are to explore the working principles and utilities of various cryptographic algorithms including secret key cryptography, hashes and message digests, public key algorithms, design issues and working principles of various authentication protocols and various secure communication standards including Kerberos, IPsec, and SSL/TLS.

UNIT I:

Basic Principles: Security Goals, Cryptographic Attacks, Services and Mechanisms, Mathematics of Cryptography- integer arithmetic, modular arithmetic, matrices, linear conguence.

UNIT II:

Symmetric Encryption: Mathematics of Symmetric Key Cryptography-algebraic structures, GF(2^n) Fields, Introduction to Modern Symmetric Key Ciphers-modern block ciphers, modern stream ciphers, Data Encryption Standard- DES structure, DES analysis, Security of DES, Multiple DES, Advanced Encryption Standard-transformations, key expansions, AES ciphers, Analysis of AES

UNIT III:

Asymmetric Encryption: Mathematics of Asymmetric Key Cryptography-primes, primality testing, factorization, CRT, Asymmetric Key Cryptography- RSA crypto system, Rabin cryptosystem, Elgamal Crypto system, ECC

UNIT IV:

Data Integrity, Digital Signature Schemes & Key Management: Message Integrity and Message Authentication-message integrity, Random Oracle model, Message authentication, Cryptographic Hash Functions-whirlpool, SHA-512, Digital Signature- process, services, attacks, schemes, applications, Key Management-symmetric key distribution, Kerberos.

UNIT V:

Network Security-I: Security at application layer: PGP and S/MIME, Security at the Transport Layer: SSL and TLS, **Network Security-II :** Security at the Network Layer: IPSec-two modes, two security protocols, security association, IKE, ISAKMP, System Security-users, trust, trusted systems, buffer overflow, malicious software, worms, viruses, IDS, Firewalls.

Text Books:

1. Cryptography and Network Security, 3rd Edition Behrouz A Forouzan, Deb deep Mukhopadhyay, McGraw Hill,2015
2. Cryptography and Network Security,4th Edition, William Stallings, (6e) Pearson,2006
3. Everyday Cryptography, 1st Edition, Keith M.Martin, Oxford, 2016

Reference Books:

1. Network Security and Cryptography, 1st Edition, Bernard Meneges, Cengage Learning, 2018