D.N.R. COLLEGE OF ENGINEERING & TECHNOLOGY (Autonomous)
BALUSUMUDI, BHIMAVARAM, W.G. Dist., A.P., PIN-534 202

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

III Year – I Semester	**Course Code:** BT24CS3101	**L**	**T**	**P**	**C**
		3	**0**	**0**	**3**
DATAWARE HOUSING & DATA MINING					

Course Objectives:

- 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.

CO1:.	Explain the **structure and phases of a compiler** and design **lexical analyzers** using regular expressions, finite automata, and lexical analyzer generators.
CO2	Construct and analyze **context-free grammars** and perform **syntax analysis** by eliminating ambiguity, left recursion, and applying left factoring.
CO3:	Design and implement **top-down and bottom-up parsing techniques**, including LL(1), SLR, CLR, and LALR parsers with appropriate error recovery methods.
CO4:.	Apply **syntax-directed definitions and translation schemes** to generate **intermediate code**, including type checking, control flow, and procedure handling.
CO5: .	Analyze and apply **code optimization techniques** using basic blocks, flow graphs, loop optimization, data-flow analysis, and peephole optimization.
CO6:	Implement **runtime storage management** and **code generation techniques**, including register allocation and procedure calls.

-
-

CO \ PO	PO 1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	1	–	2	–	–	–	–	1	–	2
CO2	3	2	2	–	3	–	–	–	–	1	–	2
CO3	3	2	2	1	3	–	–	–	–	1	–	2
CO4	3	3	3	2	3	–	–	–	1	1	–	2
CO5	3	3	2	2	2	1	1	1	–	1	–	2
CO6	3	2	3	1	3	1	–	–	1	1	–	3

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)

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:BasicAgglomerativeHierarchicalClusteringAlgorithmDBSCAN:Traditional Density Center-Based Approach, DBSCAN Algorithm, Strengths and Weaknesses. (Text Book- 2)

TEXTBOOKS:

1. DataMiningconceptsandTechniques,3rdedition,JiaweiHan,MichelKamber, Elsevier, 2011.
2. Introduction to Data Mining: Pang-Ning Tan & Michael Steinbach, Vipin Kumar, Pearson, 2012.

ReferenceBooks:

1. DataMining:VikramPudiandP.RadhaKrishna, OxfordPublisher.
2. DataMiningTechniques,ArunKPujari,3rdedition, 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

D.N.R. COLLEGE OF ENGINEERING & TECHNOLOGY (Autonomous)
BALUSUMUDI, BHIMAVARAM, W.G. Dist., A.P., PIN-534 202

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

IIIYear-ISemester	Course Code: BT24CS3102	L	T	P	C
		2	1	0	3

COMPUTER NETWORKS

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 networks applications.

Course Outcomes:

At the end of the course students will be able to

CO1:.	Explain the fundamentals of computer networks, including network types (LAN, MAN, WAN), topologies, reference models (OSI and TCP/IP), and physical layer transmission media.
CO2	Analyze data link layer design issues and protocols, including framing techniques, flow and error control mechanisms, sliding window protocols, HDLC, and Point-to-Point Protocol (PPP).
CO3:	Describe medium access control techniques and wired LAN technologies by evaluating random access, controlled access, channelization methods, and Ethernet standards.
CO4:.	Apply network layer concepts to analyze packet switching, routing algorithms, congestion control mechanisms, traffic shaping algorithms, and internetworking principles.
CO5: .	Interpret Internet Protocol operations by explaining IPv4 and IPv6 addressing schemes, header formats, subnetting, CIDR, fragmentation, tunneling, and transition mechanisms.
CO6:	Demonstrate understanding of transport and application layer protocols by explaining UDP and TCP operations, congestion control in TCP, and application services such as HTTP, email, DNS, and remote login.

CO \ PO	PO 1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	1	-	–	2	–	–	–	–	–	–	–
CO2	3	3	2	1	3	–	–	–	–	–	–	–
CO3	3	2	2	1	3	–	–	–	–	–	–	–
CO4	3	3	3	2	3	–	–	–	–	–	–	–
CO5	3	2	2	1	2	-	-	-	-	-	–	-
CO6	3	2	2	1	3	-	–	–	–	–	–	–

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 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- Tunneling, 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 Networks*, Andrew S. Tanenbaum, Fifth Edition, Pearson Education / PHI.
2. *Data Communications and Networks*, Behrouz A. Forouzan, Fifth Edition, TMH.

Reference Books:

1. *Data Communications and Networks* – Achut S. Godbole, Atul Kahate.
2. *Computer Networks*, Mayank Dave, CENGAGE.

D.N.R. COLLEGE OF ENGINEERING & TECHNOLOGY(Autonomous)
BALUSUMUDI, BHIMAVARAM, W.G. Dist., A.P., PIN-534 202

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

IIIYear-ISemester	Course Code: BT24CS3103	L	T	P	C
		3	0	0	3

FORMAL LANGUAGES AND AUTOMATA THEORY

Course Objectives:
The main objectives of the course are to:
• To learn the fundamentals of **Regular and Context-Free Grammars and Languages**.
• To understand the relationship between **Regular Languages** and **Finite Automata and Machines**.
• To learn how to design **automata and machines** as **Acceptors, Verifiers, and Translators**.
• To understand the relationship between **Context-Free Languages**, **Pushdown Automata (PDA)**, and **Turing Machines (TM)**.
• To learn how to design **PDA** as acceptors and **TM** as calculators.

Course Outcomes:
At the end of the course students will be able to

CO1:.	Explain the fundamental concepts of automata theory and design finite automata (DFA, NFA, ε-NFA), including minimization techniques and Mealy and Moore machines.
CO2	Analyze regular expressions and regular grammars, and demonstrate equivalence and inter-conversion among regular expressions, finite automata, and regular grammars using formal techniques.
CO3:	Apply context-free grammar concepts to generate and analyze formal languages, including derivations, parse trees, ambiguity removal, grammar simplification, and normal forms.
CO4:.	Design and analyze pushdown automata and establish equivalence between pushdown automata and context-free grammars for language recognition.
CO5: .	Explain the structure and functioning of Turing machines and design Turing machines to solve computational problems, including variants such as universal and restricted Turing machines.
CO6:	Analyze computability and complexity concepts by classifying problems as decidable or undecidable and distinguishing between complexity classes such as P, NP, NP-Hard, and NP-Complete.

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	2									
CO2	3	3	2	1								
CO3	3	3	2	2								
CO4	3	2	2	1								
CO5	3	3	2	2								
CO6	3	3	2	2								3

UNITI

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.

UNITII

RegularExpressions,RegularSets,IdentityRules,EquivalenceoftwoRE,Manipulationsof 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.

UNITIII

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.

UNITIV

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.

UNITV

Turing 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. IntroductiontoAutomataTheory,LanguagesandComputation,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. Mishraand N. Chandrasekharan, 3rd Edition, PHI, 2007

Reference Books:

1. ElementsofTheoryofComputation,LewisH.P.&PapadimitionC.H.,Pearson/PHI
2. TheoryofComputation,V.Kulkarni,OxfordUniversityPress,2013
3. TheoryofAutomata,LanguagesandComputation,Rajendrakumar,McGrawHill,2014

e-Resources:

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

D.N.R. COLLEGE OF ENGINEERING & TECHNOLOGY(Autonomous)
BALUSUMUDI, BHIMAVARAM, W.G. Dist., A.P., PIN-534 202

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

IIIYear-ISemester	**Course Code:** BT24CS31P1A	**L**	**T**	**P**	**C**
		3	**0**	**0**	**3**

OBJECT ORIENTED ANALYSIS AND DESIGN

Course Objectives:

The main objective is for the students to:

• Become familiar with all phases of **Object-Oriented Analysis and Design (OOAD)**.
• Master the main features of the **Unified Modeling Language (UML)**.
• Master the main concepts of **Object Technologies** and learn how to apply them in practice, developing the ability to analyze and solve challenging problems in various domains.
• Learn **Object Design Principles** and understand how to apply them toward implementation.

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

CO1:.	Explain the nature of complex systems and software complexity, and apply system architecture concepts to analyze real-world case studies.
CO2	Describe UML fundamentals, object-oriented modeling principles, and software development life cycle concepts for effective system modeling.
CO3:	Develop class and object diagrams using basic and advanced structural modeling concepts such as relationships, interfaces, packages, and roles.
CO4:.	Apply basic behavioral modeling techniques by creating use case, interaction, and activity diagrams for software systems.
CO5: .	Analyze advanced behavioral modeling concepts including state machines, events, processes, and threads using state chart diagrams.
CO6:	Design architectural models using component and deployment diagrams to represent system structure and deployment for real-world applications.

CO \ PO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
CO1	3	2	2	1								
CO2	3	2	2									
CO3	3	3	3	1								
CO4	3	3	3	1								
CO5	3	3	2	2								
CO6	3	2	3	2								2

UNITI:

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

UNITII:

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.

UNITIII:

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.

UNITIV:

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

UNITV:

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. MeilirPage-Jones:FundamentalsofObjectOrientedDesigninUML,PearsonEducation.
2. Pascal Roques: Modeling Software Systems Using UML2, WILEY- Dreamtech IndiaPvt. Ltd.
3. AtulKahate: Object Oriented Analysis & Design, TheMcGraw-Hill Companies.
4. Appling UML and Patterns: An introduction to Object – Oriented Analysis and Design and Unified Process, Craig Larman, Pearson Education.

D.N.R. COLLEGE OF ENGINEERING & TECHNOLOGY(Autonomous)
BALUSUMUDI, BHIMAVARAM, W.G. Dist., A.P., PIN-534 202

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

IIIYear-ISemester	Course Code: BT24CS31P1B	L	T	P	C
		3	0	0	3

ARTIFICIAL INTELLIGENCE

Pre-requisite:

- Knowledge in **Computer Programming**.
- A course on **Mathematical Foundations of Computer Science**.
- Background in **Linear Algebra, Data Structures and Algorithms, and Probability**.

Course Objectives:

- The students should be made to study the concepts of **Artificial Intelligence**.
- The students should be made to learn the methods of solving problems using **Artificial Intelligence**.
- The students should be made to understand the concepts of **Expert Systems**.
- To understand the applications of **Artificial Intelligence**, namely **game playing, theorem proving, and machine learning**.
- To learn different **knowledge representation techniques**.

Course Outcomes:

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

CO1:.	Explain the foundations, history, and problem-solving aspects of Artificial Intelligence, including intelligent agents, environments, rationality, and problem formulation.
CO2	Apply uninformed and heuristic search techniques, game-playing strategies, and adversarial search algorithms to solve AI problems.
CO3:	Analyze knowledge representation techniques and reasoning methods using predicate logic, semantic networks, frames, rules, and probabilistic models.
CO4:.	Apply logical inference techniques and learning methods such as inductive learning, decision trees, statistical learning, and reinforcement learning.
CO5: .	Explain reasoning under uncertainty using probability theory, Bayesian inference, and Dempster–Shafer theory.
CO6:	Describe the architecture, components, and applications of expert systems and evaluate the role of expert systems in real-world problem domains.

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	1									
CO2	3	3	2	1								
CO3	3	3	2	2								
CO4	3	3	2	2								
CO5	3	3	1	2								
CO6	3	2	2	1								2

UNIT-I

Introduction: AI problems, foundation of AIand history of AI intelligent agents: Agents and Environments, the concept of rationality, the nature of environments, structure of agents, problem solving agents, problem formulation.

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-maxalgorithm, 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.RusselandP.Norvig,"ArtificialIntelligence–AModernApproach", SecondEdition,PearsonEducation.
2. KevinNightandElaineRich,NairB.,"ArtificialIntelligence(SIE)",McGrawHill

ReferenceBooks:

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:AnewSynthesis",Elsevier Publishers.
4. ArtificialIntelligence,SarojKaushik,CENGAGELearning.

OnlineLearning Resources:

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

D.N.R. COLLEGE OF ENGINEERING & TECHNOLOGY(Autonomous)
BALUSUMUDI, BHIMAVARAM, W.G. Dist., A.P., PIN-534 202

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

IIIYear-ISemester	Course Code: BT24CS31P1C	L	T	P	C
		0	0	3	1.5
MICROPROCESSORS & MICROCONTROLLERS					

Course Objectives:

- Tointroducefundamentalarchitecturalconceptsofmicroprocessorsand microcontrollers.
- Toimpartknowledgeon addressingmodesandinstructionsetof8086and8051
- Tointroduceassemblylanguageprogramming concepts
- Toexplainmemoryand I/Ointerfacingwith8086and8051

Tointroduce16 bit and32 bit microcontrollers.

Course Outcomes:

At the end of the course students will be able to

CO1:.	Explain the **structure and phases of a compiler** and design **lexical analyzers** using regular expressions, finite automata, and lexical analyzer generators.
CO2	Develop assembly language programs for the 8086 microprocessor using various instruction sets, addressing modes, and assembler directives.
CO3:	Design and analyze interfacing of the 8086 microprocessor with memory and peripheral devices such as PPI, USART, DMA controller, displays, motors, and converters.
CO4:.	Explain the architecture and instruction set of the 8051 microcontroller and develop assembly language programs for embedded control applications.
CO5: .	Apply interfacing and programming techniques of the 8051 microcontroller for timers, serial communication, interrupts, LCD, keyboard, sensors, ADC/DAC, and external memory.
CO6:	Compare microprocessors and microcontrollers with modern processors such as PIC and ARM, and analyze their applications in real-time embedded systems.

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	1									
CO2	3	3	2	1	2							
CO3	3	3	3	2	2							
CO4	3	2	2	1	2							
CO5	3	3	3	2	2							
CO6	3	2	2	1								2

UNITI:

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.

UNITII:

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

UNITIII:

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.

UNITIV:

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

UNITV:

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. RajKamal,Microcontrollers:Architecture,Programming,InterfacingandSystem Design, 2nd edition, Pearson, 2012.

Reference Books:

1. RameshSGaonkar,MicroprocessorArchitectureProgrammingandApplicationswith the 8085, 6thedition, Penram International Publishing, 2013.
2. KennethJ.Ayala,The8051 Microcontroller,3rd edition,CengageLearning,2004.

N.R. COLLEGE OF ENGINEERING & TECHNOLOGY(Autonomous)
BALUSUMUDI, BHIMAVARAM, W.G. Dist., A.P., PIN-534 202

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

IIIYear-ISemester	Course Code: BT24CS31P1D	L	T	P	C
		0	0	3	1.5
QUANTUM COMPUTING					

Course Objectives:

To introduce the fundamentals of Quantum Computing and the problem-solving approach using finite-dimensional mathematics.

CO1:.	Explain the historical evolution of quantum computing and differentiate between classical and quantum computation concepts such as bits, qubits, and logical operations.
CO2	Apply background mathematical, physical, and biological concepts including linear algebra, quantum mechanics principles, probability, and basic genomics relevant to quantum computing.
CO3:	Describe qubits as quantum information units and design quantum circuits using single-qubit and multi-qubit gates, Bloch sphere representation, and Bell states.
CO4:.	Analyze and apply quantum algorithms such as Deutsch, Deutsch–Jozsa, Shor's, and Grover's algorithms and relate quantum and classical complexity classes.
CO5: .	Explain the effects of noise in quantum systems and apply quantum error correction and fault-tolerant computation techniques.
CO6:	Analyze quantum information and cryptography concepts including quantum teleportation and compare classical and quantum information theories.

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	1										
CO2	3	2		1								
CO3	3	3	2	1								
CO4	3	3	2	2								
CO5	3	3	2	2								
CO6	3	2	1	1				1				3

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 qubits. Qubit as a quantum unit of information. The Bloch sphere.

Quantum Circuits: Single-qubit gates, multiple-qubit gates, designing 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,NielsenM.A.,Cambridge
2. ProgrammingQuantumComputers,EssentialAlgorithmsandCodeSamples,EricR Johnson, Nic Harrigan, Mercedes Ginemo, Segovia, Oreilly

Reference Books:

1. Quantum Computing for ComputerScientists,NosonS.Yanofsk,MircoA. MannucciPrinciplesofQuantumComputationandInformation,BenentiG.,CasatiG.andS trini G., Vol.I: Basic Concepts, Vol II
2. BasicToolsandSpecialTopics,WorldScientific.PittengerA.O.,AnIntroductionto Quantum Computing Algorithms

N.R. COLLEGE OF ENGINEERING & TECHNOLOGY(Autonomous)
BALUSUMUDI, BHIMAVARAM, W.G. Dist., A.P., PIN-534 202

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

IIIYear-ISemester	Course Code: BT24CS3105	L	T	P	C
		0	1	2	2
DATA MINING LAB					

Pre-requisites: Data Base Management Systems, Python Programming
Course Objectives: The main objectives of the course are 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 usable form.
• Emphasize **hands-on experience** working with real datasets.
• Test real datasets using popular **data mining tools** such as **WEKA** and Python libraries.
• Develop the ability to design various **algorithms based on data mining tools**.

Software Requirements: WEKA Tool/Python/R-Tool/RapidTool/OracleData 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 tools like **Microsoft SSIS, Informatica, Business Objects, etc.**
• Design **multi-dimensional data models**, namely **Star, Snowflake, and Fact Constellation schemas**, for any one enterprise (e.g., Banking, Insurance, Finance, Healthcare, Manufacturing, Automobiles, Sales, etc.).
• Write **ETL scripts** and implement them using data warehouse tools.
• Perform various **OLAP operations** such as **slice, dice, rollup, drill-down, and pivot**.

2. Explore Machine Learning Tool "WEKA"
• Explore **WEKA Data Mining/Machine Learning Toolkit**.
• Download and/or install the **WEKA toolkit**.
• Understand the features of WEKA, such as **Explorer, Knowledge Flow interface, Experimenter, and command-line interface**.
• Navigate the options available in WEKA (e.g., **Select Attributes panel, Preprocess panel, Classify panel, Cluster panel, Associate panel, Visualize panel**).
• Study the **ARFF file format**. Explore available datasets in WEKA and load a dataset (e.g., **Weather dataset, Iris dataset, etc.**).
• Load each dataset and observe the following:

1. List the attribute names and their 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 Association Rule Mining
• Explore preprocessing options in WEKA and apply **unsupervised filters** like **Discretization, Resample, etc.** on each dataset.
• Load **weather. nominal, Iris, Glass datasets** into WEKA and run the **Apriori Algorithm** with different support and confidence values.
• Study the rules generated. Apply different **discretization filters** on numerical attributes and run Apriori association rule algorithm again. Study the rules generated.
• Derive **insights** and observe the effect of discretization on rule generation.

4. Demonstrate Classification on Datasets (WEKA/R)
• Load each dataset and run **ID3, J48 classification algorithms**. Study classifier output and 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** and **k-Nearest Neighbor classification**. Interpret results.
• Plot **ROC curves**.
• Compare classification results of **ID3, J48, Naïve Bayes, and k-NN classifiers** for each dataset, and deduce which classifier performs best and worst, with justification.

5. Demonstrate Clustering of Datasets
• Load each dataset into WEKA/R and run **k-means clustering algorithm** with different values of **k**.
• Study clusters formed. Observe the **sum of squared errors** and **centroids**, and derive insights.
• Explore other clustering techniques available in WEKA/R.
• Explore visualization features to visualize clusters and derive insights.

6. Demonstrate Knowledge Flow Application on Datasets (WEKA/R)
• Develop a **Knowledge Flow layout** for finding strong association rules using **Apriori and FP-Growth algorithms**.
• Set up the knowledge flow to load an **ARFF file (batch mode)** and perform **cross-validation** using **J48 algorithm**.

7. Demonstrate plotting multiple ROC curves
• Plot multiple ROC curves in the same window using **J48** and **Random Forest tree**.

8. Demonstrate ZeroR technique on Iris dataset
• Apply necessary **preprocessing techniques** and share observations.

9. Write a Java program to prepare a simulated dataset with unique instances.

10. Write a Python program to generate frequent itemsets / association rules using Apriori algorithm.

11. Write a program to calculate chi-square value using Python/R and report observations.

12. Write a program of Naïve Bayesian classification using Python/R.

13. Implement a Java/R program to perform Apriori algorithm.

14. Write an R program to cluster your choice of data using simple k-means algorithm using JDK.

15. Write a program of cluster analysis using simple k-means algorithm in Python/R.

16. Write a program to compute/display dissimilarity matrix (for your own dataset containing at least four instances with two attributes) using Python.

17. Visualize datasets using Matplotlib in Python/R (Histogram, Box plot, Bar chart, Pie chart, etc.).

N.R. COLLEGE OF ENGINEERING & TECHNOLOGY(Autonomous)
BALUSUMUDI, BHIMAVARAM, W.G. Dist., A.P., PIN-534 202

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

IIIYear-ISemester	Course Code: BT24CS3106	L	T	P	C
		2	0	0	-

COMPUTER NETWORKS LAB

Course Objectives:
To learn the basic concepts of **computer networking** and acquire practical knowledge of **network protocols**, with emphasis on **TCP/IP**.
The laboratory provides a practical approach to **Ethernet/Internet networking**, where networks are assembled and experiments are conducted to understand the **layered architecture** and the working of important protocols.

List of Experiments:

1. Study of **network devices** in detail and connect computers in a **Local Area Network (LAN)**.
2. Write a program to implement **data link layer framing methods** such as:
   i) Character stuffing
   ii) Bit stuffing
3. Write a program to implement **data link layer framing method – checksum**.
4. Write a program for **Hamming code generation** for error detection and correction.
5. Write a program to implement, on a dataset of characters, the three **CRC polynomials**: **CRC-12, CRC-16, and CRC-CCIP**.
6. Write a program to implement **Sliding Window Protocol** for **Go-Back-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 the **Leaky Bucket algorithm**.
10. Write a program to implement **Dijkstra's algorithm** to compute the **shortest path** through a graph.
11. Write a program to implement **Distance Vector Routing Algorithm** by obtaining the routing table at each node (take an example subnet graph with weights indicating delay between nodes).
12. Write a program to implement a **Broadcast Tree** by taking a subnet of hosts.
13. **Wireshark**
    i) Packet capture using Wireshark
    ii) Starting Wireshark
    iii) Viewing captured traffic
    iv) Analysis, statistics, and filters
14. How to run an **Nmap scan**.
15. **Operating System Detection** using Nmap.
16. Do the following using **NS2 Simulator**:
    i) NS2 Simulator – Introduction
    ii) Simulation to find the number of packets dropped
    iii) Simulation to find the number of packets dropped by **TCP/UDP**
    iv) Simulation to find the number of packets dropped due to **congestion**
    v) Simulation to compare **Data Rate and Throughpu**

D.N.R. COLLEGE OF ENGINEERING & TECHNOLOGY(Autonomous)
BALUSUMUDI, BHIMAVARAM, W.G. Dist., A.P., PIN-534 202

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

IIIYear-ISemester	Course Code: BT24CS3107	L	T	P	C
		0	1	2	2
FULLSTACKDEVELOPMENT -2					

Course Objectives:

The main objectives of the course are to:
• Make use of **router, template engine, and authentication using sessions** to develop applications in **Express JS**.
• Build a **single-page application (SPA)** using **RESTful APIs** in Express JS.
• Apply **router and hooks** in designing **React JS applications**.
• Make use of **MongoDB queries** to perform **CRUD operations** on document databases.

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 demonstrate the working of **middleware**.

2. Express JS – Templating, Form Data
a. Write a program using a **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 **Mongoose** and perform CRUD operations.
b. Write a program to develop a **single-page application using RESTful APIs**.

5. React JS – Render HTML, JSX, Components (Function & Class)
a. Write a program to **render HTML** on 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. React JS – 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) and display data.
c. Write a program to **respond to events**.

7. React JS – 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. React JS – 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** dynamically.

9. React JS – Hooks, Sharing Data Between Components
a. Write a program to understand the importance of using **hooks**.
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 documents 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 Node JS and Express JS.
b. Design a **Quiz app** using React JS.
c. Complete the **MongoDB certification** from the MongoDB University website.

Text Books:

1. ProMERNStack:FullStackWebAppDevelopmentwithMongo,Express,React,and Node, Vasan Subramanian, 2nd edition, APress, O'Reilly.
2. Node.JsinAction,MikeCantelon,MarkHarter,T.J.Holowaychuk,NathanRajlich,Manning Publications. (Chapters 1-11)
3. ReactQuickly,AzatMardan,ManningPublications(Chapters1-8,12-14)

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

D.N.R. COLLEGE OF ENGINEERING & TECHNOLOGY(Autonomous)
BALUSUMUDI, BHIMAVARAM, W.G. Dist., A.P., PIN-534 202

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

IIIYear-ISemester	Course Code: BT24CS3108	L	T	P	C
		0	1	2	2

USER INTERFACE DESIGN USING FLUTTER

Course Objectives:

- Learn to implement **Flutter widgets and layouts**.
- Understand **responsive UI design** and **navigation** in Flutter.
- Gain knowledge of **widgets** and how to **customize widgets** for specific UI elements and themes.
- Understand how to include **animations**, apart from fetching and displaying data.

List of Experiments:
Students need to implement the following experiments:

1. Getting Started with Flutter and Dart
a) Install **Flutter** and **Dart SDK**.
b) Write a simple **Dart program** to understand the language basics.

2. Exploring Flutter Widgets and Layouts
a) Explore various Flutter widgets (**Text, Image, Container, etc.**).
b) Implement different layout structures using **Row, Column, and Stack widgets**.

3. Responsive UI Design
a) Design a **responsive UI** that adapts to different screen sizes.
b) Implement **media queries and breakpoints** for responsiveness.

4. Navigation in Flutter
a) Set up **navigation between different screens** using `Navigator`.
b) Implement **navigation with named routes**.

5. Stateful and Stateless Widgets
a) Learn about **stateful** and **stateless widgets**.
b) Implement **state management** using `setState` and **Provider**.

6. Custom Widgets and Theming
a) Create **custom widgets** for specific UI elements.
b) Apply **styling using themes and custom styles**.

7. Forms and Validation
a) Design a **form with various input fields**.
b) Implement **form validation and error handling**.

8. Animations in Flutter
a) Add **animations** to UI elements using Flutter's animation framework.
b) Experiment with different types of animations (**fade, slide, etc.**).

9. Working with REST APIs
a) Fetch data from a **REST API**.
b) Display the fetched data in a **meaningful way in the UI**.

10. Testing and Debugging
a) Write **unit tests** for UI components.
b) Use Flutter's **debugging tools** to identify and fix issues.

Text Books:

1. MarcoL.Napoli,BeginningFlutter:AHands-onGuidetoAppDevelopment.
2. RapPayne,BeginningAppDevelopmentwithFlutter:CreateCross-PlatformMobile Apps 1st Edition, Apres
3. RichardRose,Flutter&DartCookbook,DevelopingFullstackApplicationsforthe Cloud, Oreilly.

D.N.R. COLLEGE OF ENGINEERING & TECHNOLOGY(Autonomous)
BALUSUMUDI, BHIMAVARAM, W.G. Dist., A.P., PIN-534 202

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

IIIYear-IISemester	**Course Code:** BT24CS3201	**L**	**T**	**P**	**C**
		0	**1**	**2**	**2**

COMPILER DESIGN

Course Objectives:

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

Course Outcomes (COs)

After completing this course, students will be able to:

CO1:.	Explain the **structure and phases of a compiler** and design **lexical analyzers** using regular expressions, finite automata, and lexical analyzer generators.
CO2	Construct and analyze **context-free grammars** and perform **syntax analysis** by eliminating ambiguity, left recursion, and applying left factoring.
CO3:	Design and implement **top-down and bottom-up parsing techniques**, including LL(1), SLR, CLR, and LALR parsers with appropriate error recovery methods.
CO4:.	Apply **syntax-directed definitions and translation schemes** to generate **intermediate code**, including type checking, control flow, and procedure handling.
CO5: .	Analyze and apply **code optimization techniques** using basic blocks, flow graphs, loop optimization, data-flow analysis, and peephole optimization.
CO6:	Implement **runtime storage management** and **code generation techniques**, including register allocation and procedure calls.

CO \ PO	PO 1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	1	–	2	–	–	–	–	1	–	2
CO2	3	2	2	–	3	–	–	–	–	1	–	2
CO3	3	2	2	1	3	–	–	–	–	1	–	2
CO4	3	3	3	2	3	–	–	–	1	1	–	2
CO5	3	3	2	2	2	1	1	1	–	1	–	2
CO6	3	2	3	1	3	1	–	–	1	1	–	3

UNITI:

LexicalAnalysis: LanguageProcessors,StructureofaCompiler,LexicalAnalysis,TheRole 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,

UNITII:

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.

UNITIII:

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.

UNITIV:

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

UNITV:

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,TechniquesandTools,SecondEdition,AlfredV.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, CambridgeUniversity Press.
3. OptimizingCompilersforModernArchitectures,RandyAllen,KenKennedy, Morgan Kauffmann, 2001.
4. Levine, J.R., T. Mason and D. Brown, Lex andYacc, edition, O'Reilly & Associates,1990

D.N.R. COLLEGE OF ENGINEERING & TECHNOLOGY(Autonomous)
BALUSUMUDI, BHIMAVARAM, W.G. Dist., A.P., PIN-534 202

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

IIIYear-IISemester	Course Code: BT24CS3202	L	T	P	C
		0	1	2	2

CLOUD COMPUTING

Course Objectives:

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

CO1:.	Explain the fundamentals of cloud computing, including service models, deployment models, reference architecture, characteristics, benefits, and major cloud service providers.
CO2	Describe cloud-enabling technologies such as parallel and distributed computing, service-oriented architecture, web services, RPC, and virtualization.
CO3:	Analyze virtualization and containerization concepts and technologies, including hypervisors, containers, orchestration tools, and public cloud VM and container services.
CO4:.	Evaluate cloud computing challenges related to scalability, fault tolerance, interoperability, economics, energy efficiency, and cloud security models.
CO5: .	Apply advanced cloud computing concepts such as serverless computing, Function-as-a-Service, IoT integration, edge and fog computing.
CO6:	Analyze emerging trends and practices in cloud computing including DevOps, infrastructure-as-code, and quantum cloud computing for modern IT systems.

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	1									
CO2	3	2	1	1								
CO3	3	3	2	1	2							
CO4	3	3	2	2				1				
CO5	3	2	2	1	2							
CO6	3	2	2	1	2							3

UNIT-I: Introduction to Cloud Computing Fundamentals
Cloud computing provides an overview of modern computing paradigms and introduces the concept of a cloud. The course covers the **cloud computing reference model**, including the various types of services such as **IaaS, PaaS, and SaaS**, and deployment models like **public, private, and hybrid clouds**. Students will learn about **utility computing**, the characteristics and benefits of cloud computing, and major **cloud service providers** such as **Amazon Web Services, Microsoft Azure, and Google App Engine**.

UNIT-II: Cloud Enabling Technologies
This unit explores the enabling technologies that support cloud computing. It discusses the **ubiquitous Internet**, **parallel and distributed computing**, and the essential elements of parallel computing, including hardware architectures like **SISD, SIMD, MISD, and MIMD**. Students will gain an understanding of distributed computing, **inter-process communication**, technologies for distributed systems, and **remote procedure calls (RPC)**. Additionally, the unit introduces **Service-Oriented Architecture (SOA)**, **web services**, and the role of **virtualization** in cloud environments.

UNIT-III: Virtualization and Containers
Students will study the characteristics of **virtualized environments** and a taxonomy of virtualization techniques. The unit explains the advantages and disadvantages of virtualization and its integration with cloud computing, with technology examples such as **XEN and VMware**. The course also covers **containers**, including their building blocks, container platforms like **LXC and Docker**, and orchestration tools such as **Docker Swarm and Kubernetes**. Public cloud offerings, including **VMs (e.g., Amazon EC2)** and container services (e.g., **Amazon Elastic Container Service**), are also examined.

UNIT-IV: Cloud Computing Challenges
This unit focuses on the challenges in cloud computing. Students will study the **economics of the cloud**, interoperability and standards, **scalability**, **fault tolerance**, and **energy efficiency**. The course explores federated cloud systems and the fundamentals of **cloud computing security**, including security architecture and the **shared responsibility model**. Security considerations across different cloud deployment models are also discussed.

UNIT-V: Advanced Concepts in Cloud Computing
The final unit introduces advanced cloud computing concepts such as **serverless computing** and **Function-as-a-Service (FaaS)**. Students will learn about serverless architectures, public cloud platforms like **AWS Lambda**, and open-source alternatives such as **OpenFaaS**. The unit covers the **Internet of Things (IoT)**, cloud-centric IoT layers, and applications. Students will also explore **edge and fog computing**, **DevOps**, **infrastructure-as-code**, and emerging areas such as **quantum cloud computing**.

Text Books:

1. MasteringCloudComputing,2nd edition,RajkumarBuyya,ChristianVecchiola,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. EssentialsofcloudComputing,K.Chandrasekhran,CRCpress,2014.
3. Online documentation and tutorials from cloud service providers (e.g., AWS, Azure, GCP)

D.N.R. COLLEGE OF ENGINEERING & TECHNOLOGY(Autonomous)
BALUSUMUDI, BHIMAVARAM, W.G. Dist., A.P., PIN-534 202

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

IIIYear-IISemester	Course Code: BT24CS3203	L	T	P	C
		0	1	2	2
CRYPTOGRAPHY& NETWORK SECURITY					

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.

CO1:.	Explain the basic principles of information security, cryptographic services, attacks, and the mathematical foundations required for cryptography.
CO2	Analyze symmetric key cryptographic techniques, including DES and AES algorithms, their structure, security, and performance characteristics.
CO3:	Apply asymmetric key cryptographic algorithms such as RSA, ElGamal, Rabin, and Elliptic Curve Cryptography for secure communication.
CO4:.	Explain data integrity mechanisms, cryptographic hash functions, digital signature schemes, and key management techniques including Kerberos.
CO5: .	Analyze security mechanisms at application, transport, and network layers using protocols such as PGP, SSL/TLS, and IPSec.
CO6:	Evaluate system security threats and protection mechanisms including malware, buffer overflow attacks, intrusion detection systems, and firewalls.

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	1					1				
CO2	3	3	2	1								
CO3	3	3	2	1								
CO4	3	2	2	1				1				
CO5	3	3	2	2				1				
CO6	3	3	2	2				1				3

UNITI:

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

UNITII:

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.

UNITIII:

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

UNITIV:

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.

UNITV:

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.

TextBooks:

1. CryptographyandNetworkSecurity,3rdEditionBehrouzAForouzan,Debdeep Mukhopadhyay, McGraw Hill,2015
2. Cryptographyand NetworkSecurity,4thEdition,WilliamStallings, (6e) Pearson,2006
3. EverydayCryptography,1stEdition,KeithM.Martin,Oxford,2016

ReferenceBooks:

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

BALUSUMUDI, BHIMAVARAM, W.G. Dist., A.P., PIN-534 202

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

IIIYear-IISemester	Course Code: BT24CS32P2A	L	T	P	C
		0	1	2	2
SOFTWARE TESTING METHODOLOGIES					

Course Objectives

- Toprovideknowledgeoftheconceptsinsoftwaretestingsuchastestingprocess, criteria, strategies, and methodologies.
- Todevelopskillsinsoftwaretestautomationandmanagementusingthelatest tools.

CO1:.	Explain the purpose, principles, and models of software testing, including the impact of bugs and the taxonomy of software errors.
CO2	Apply path testing techniques, including flow graphs, path predicates, path sensitization, and instrumentation to identify test paths.
CO3:	Analyze data flow, transaction flow, and domain-based testing strategies and apply them for validating software correctness and interface behavior.
CO4:.	Apply logic-based testing techniques such as decision tables, KV charts, and path expressions for systematic test case generation.
CO5: .	Analyze state-based testing approaches including state graphs, transition testing, and testability considerations for software systems.
CO6:	Use graph matrices, node reduction algorithms, and software testing tools (e.g., JMeter, Selenium, SoapUI, Catalon) to design and execute practical test cases.

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	1									
CO2	3	3	2	1								
CO3	3	3	2	1								
CO4	3	3	2	2								
CO5	3	3	2	2								
CO6	3	2	3	2	2							3

UNIT-I

Introduction: Purpose of testing, Dichotomies, model for testing, consequences of bugs, taxonomy of bugs Flow graphs and Path testing: Basics concepts of path testing, predicates, path predicates and achievable paths, path sensitizing, path instrumentation, application of path testing.

UNIT-II

TransactionFlowTesting:transactionflows,transactionflowtesting techniques.
Data Flow testing: Basics of data flow testing, strategies in data flow testing, application ofdata flow testing.
DomainTesting:domainsandpaths,Nice&uglydomains,domaintesting,domainsand interfaces testing,

domain and interface testing, domains and testability.

UNIT-III

Paths,PathproductsandRegularexpressions:pathproducts&pathexpression,reduction procedure, applications, regular expressions & flow anomaly detection.
LogicBasedTesting:overview,decisiontables,pathexpressions,kvcharts,specifications.

UNIT-IV

State,StateGraphsandTransitiontesting:stategraphs,good&badstategraphs,statetesting, Testability tips.

UNIT-V

Graph Matrices and Application: Motivational overview, matrix of graph, relations, power of a matrix, node reduction algorithm, building tools. (Student should be given an exposure to a tool like Jmeter/selenium/soapUI/Catalon).

TextBooks:

1. SoftwareTestingtechniques-BarisBeizer,Dreamtech,second edition.
2. SoftwareTestingTools –Dr.K.V.K.K.Prasad,Dreamtech.

ReferenceBooks:

1. Thecraftofsoftwaretesting -BrianMarick,Pearson Education.
2. SoftwareTestingTechniques– SPD(Oreille)
3. SoftwareTestingintheRealWorld–Edward Kit,Pearson.
4. EffectivemethodsofSoftwareTesting,Perry,JohnWiley.
5. ArtofSoftwareTesting –Meyers,JohnWiley.

		0	1	2	2

CYBER SECURITY

Course Objectives:

Theaim of thecourseisto

- identifysecurityrisksandtakepreventive steps
- understandtheforensics fundamentals
- understandtheevidencecapturingprocess
- understandthepreservationofdigital evidence

CO1:.	Explain the fundamentals of cybercrime, including its origins, types, attack vectors, and security challenges posed by mobile and networked devices.
CO2	Analyze common cybercrime tools and attack methods, including phishing, malware, buffer overflows, DoS/DDoS attacks, SQL injection, spoofing, and social engineering.
CO3:	Apply techniques for cybercrime investigation including digital evidence collection, e-discovery, email/IP tracking, password recovery, and case-based forensic analysis.
CO4:.	Demonstrate knowledge of computer forensics, including software and hardware tools, system forensics (Windows/Linux), multimedia forensics, biometrics, and mobile device investigations.
CO5: .	Evaluate cybercrime legal frameworks, including global cyber laws, the Indian IT Act, amendments, digital signatures, and challenges in legal enforcement.
CO6:	Develop awareness of ethical, legal, and technological implications of cybercrime and apply best practices for prevention, detection, and mitigation of cyber threats.

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	1									
CO2	3	3	2	1								
CO3	3	3	2	2								
CO4	3	3	2	2	2			1				
CO5	3	2	1	1				1				2
CO6	3	2	2	1		1		2				3

UNIT I: Introduction to Cybercrime: Introduction, Cybercrime: Definition and Origins of the Word, Cybercrime and Information Security, Cyber criminals,Classificationsof Cybercrime, Cyberstalking, Cybercafe and Cybercrimes, Botnets. Attack Vector,Proliferation ofMobile and Wireless Devices, Security Challenges Posed by Mobile Devices, Attacks on Mobile/Cell Phones, Network and Computer Attacks.

UNIT II: Tools and Methods : Proxy Servers and Anonymizers, Phishing, Password Cracking, Keyloggers and Spywares, Virus and Worms, Trojan Horses and Backdoors, Steganography, Sniffers, Spoofing, Session Hijacking Buffer over flow, DoS and DDoS Attacks, SQL Injection, Buffer Overflow, Attacks on Wireless Networks, Identity Theft (ID Theft), Foot Printing and

Social Engineering, Port Scanning, Enumeration.

UNIT III: Cyber Crime Investigation: Introduction, Investigation Tools, eDiscovery, Digital Evidence Collection, Evidence Preservation, E-Mail Investigation, E-Mail Tracking, IP Tracking, E-Mail Recovery, Hands on Case Studies. Encryption and Decryption Methods, Search and Seizure of Computers, Recovering Deleted Evidences, Password Cracking.

UNIT IV: Computer Forensics and Investigations: Understanding Computer Forensics, Preparing for Computer Investigations. Current Computer Forensics Tools: Evaluating Computer Forensics Tools, Computer Forensics Software Tools, Computer Forensics Hardware Tools, Validating and Testing Forensics Software, Face, Iris and Fingerprint Recognition, Audio Video Analysis, Windows System Forensics, Linux System Forensics, Graphics and Network Forensics, E-mail Investigations, Cell Phone and Mobile Device Forensics.

UNIT V: Cyber Crime Legal Perspectives: Introduction, Cybercrime and the Legal Landscape around the World, The Indian IT Act, Challenges to Indian Law and Cybercrime Scenario in India, Consequences of Not Addressing the Weaknessin InformationTechnology Act,Digital SignaturesandtheIndianITAct,AmendmentstotheIndianIT Act,CybercrimeandPunishment, Cyberlaw, Technology and Students: Indian Scenario.

TextBooks:

1. SunitBelapureNinaGodbole"CyberSecurity:UnderstandingCyberCrimes,Computer Forensics and Legal Perspectives", WILEY, 2011.
2. NelsonPhillipsandEnfingerSteuart,"ComputerForensicsandInvestigations", CengageLearning,New Delhi,2009.

ReferenceBooks:

1. MichaelT.Simpson,KentBackmanandJamesE.Corley,"HandsonEthical Hacking and Network Defence", Cengage, 2019.
2. Computer Forensics,Computer Crime InvestigationbyJohnR.Vacca,Firewall Media, New Delhi.
3. AlfredBasta,NadineBasta,MaryBrownandRavinderKumar"CyberSecurity and Cyber Laws" , Cengage,2018.

E-Resources:

1. CERT-InGuidelines-http://www.cert-in.org.in/
2. https://www.coursera.org/learn/introduction-cybersecurity-cyber-attacks[Online Course]
3. https://computersecurity.stanford.edu/free-online-videos[FreeOnline Videos]
4. Nickolai Zeldovich. 6.858 Computer Systems Security. Fall 2014. Massachusetts Institute of Technology: MIT OpenCourseWare, https://ocw.mit.eduLicense: Creative Commons BY-NC-SA.

D.N.R. COLLEGE OF ENGINEERING & TECHNOLOGY(Autonomous)
BALUSUMUDI, BHIMAVARAM, W.G. Dist., A.P., PIN-534 202

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

IIIYear-IISemester	**Course Code:** BT24CS32P2C	**L**	**T**	**P**	**C**
		0	**1**	**2**	**2**
DEVOPS					

Course Objectives:

Themain objectivesof thiscourseareto:

- DescribetheagilerelationshipbetweendevelopmentandIT operations.
- Understand the skill sets and high-functioning teams involved in DevOps and related methods to reach a continuous delivery capability.
- ImplementautomatedsystemupdateandDevOps lifecycle.

CO1:.	Explain the fundamentals of DevOps, including SDLC, Agile model, DevOps principles, architecture, lifecycle, CI/CD concepts, workflow, and release management practices.
CO2	Demonstrate version control and source code management using Git, including workflows, branching, staging, collaboration, and code quality analysis with tools like SonarQube.
CO3:	Apply build automation and Continuous Integration (CI) concepts using Jenkins, including pipeline creation, master-slave architecture, user management, and automated build execution.
CO4:.	Implement Continuous Delivery (CD) processes, containerization using Docker, and testing automation with tools like Selenium and JavaScript testing frameworks.
CO5: .	Analyze configuration management using Ansible, Puppet, and Chef, including deployment tasks, roles, templating, and automation for infrastructure management.
CO6:	Deploy, manage, and orchestrate containerized applications on Kubernetes/OpenShift, understanding namespaces, resources, deployment strategies, and CI/CD integration in cloud environments.

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	1	1								
CO2	3	3	2	1	2							
CO3	3	3	3	1	2							
CO4	3	3	3	2	2							
CO5	3	2	2	1	2							
CO6	3	2	2	2	2							3

UNIT-I

Introduction to DevOps: Introduction to SDLC, Agile Model. Introduction to Devops. DevOps Features, DevOps Architecture, DevOps Lifecycle, Understanding Workflow and principles, Introduction to DevOps tools, Build Automation, Delivery Automation, Understanding Code Quality, Automation of CI/ CD. Release management, Scrum, Kanban, delivery pipeline, bottlenecks, examples

UNIT-II

Source Code Management (GIT): The need for source code control, The history of source code management, Roles and code, source code management system and migrations. What is Version Control and GIT, GIT Installation, GIT features, GIT workflow, working withremote repository, GIT commands, GIT branching, GIT staging and collaboration. UNIT TESTING - CODE COVERAGE: Junit, nUnit & Code Coverage with Sonar Qube, SonarQube - Code Quality Analysis.

UNIT-III

BuildAutomation-ContinuousIntegration(CI): BuildAutomation,WhatisCIWhy Clis Required, CI tools, Introduction to Jenkins (With Architecture), jenkins workflow, jenkins master slave architecture, Jenkins Pipelines, PIPELINE BASICS - Jenkins Master, Node, Agent, and Executor Freestyle Projects & Pipelines, Jenkins for Continuous Integration, Create and Manage Builds, User Management in Jenkins Schedule Builds, Launch Builds on Slave Nodes.

UNIT-IV

Continuous Delivery (CD): Importance of Continuous Delivery, CONTINUOUS DEPLOYMENT CD Flow, Containerization with Docker: Introduction to Docker, Docker installation, Docker commands, Images & Containers, DockerFile, Running containers, Working with containers and publish to Docker Hub.

TestingTools: IntroductiontoSeleniumanditsfeatures,JavaScripttesting.

UNIT-V

ConfigurationManagement-ANSIBLE: IntroductiontoAnsible,Ansibletasks,Roles, Jinja templating, Vaults, Deployments using Ansible.

CONTAINERIZATION USING KUBERNETES(OPENSHIFT): Introduction to Kubernetes Namespace & Resources, CI/CD - On OCP, BC, DC & ConfigMaps, Deploying Apps on Openshift Container Pods. Introduction to Puppet master and Chef.

TextBooks:

1. Joyner,Joseph.,DevopsforBeginners:DevopsSoftwareDevelopmentMethodGuide for Software Developers and It Professionals, 1st Edition Mihails Konoplows, 2015.
2. AlissonMachadodeMenezes.,Hands-onDevOpswithLinux,1st Edition,BPBPublications, India, 2021.

ReferenceBooks:

1. LenBass,IngoWeber,LimingZhu.DevOps:ASoftwareArchitect'sPerspective. Addison Wesley; ISBN-10
2. GeneKimJeHumble,PatrickDebois,JohnWillis.TheDevOpsHandbook,1st Edition, IT Revolution Press, 2016.
3. Verona,JoakimPracticalDevOps,1st Edition,PacktPublishing,2016.
4. JoakimVerona.PracticalDevops,Ingramshorttitle;2nd edition(2018).ISBN10: 1788392574
5. DeepakGaikwad,Viral Thakkar.DevOpsTools from Practitioner'sViewpoint.Wiley publications. ISBN: 9788126579952

D.N.R. COLLEGE OF ENGINEERING & TECHNOLOGY(Autonomous)
BALUSUMUDI, BHIMAVARAM, W.G. Dist., A.P., PIN-534 202

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

IIIYear-IISemester	**Course Code:** BT24CS32P2D	**L**	**T**	**P**	**C**
		0	**1**	**2**	**2**
MACHINE LEARNING					

Course Objectives:

Theobjectivesofthe courseisto

- Definemachinelearninganditsdifferenttypes(supervisedandunsupervised)and understand their applications.
- Applysupervisedlearningalgorithmsincludingdecisiontreesandk-nearest neighbours (k-NN).
- Implementunsupervised learningtechniques,such asK-meansclustering.

CO1:.	Explain the fundamentals of Machine Learning, including paradigms, data types, stages of ML, feature engineering, model selection, learning, evaluation, and prediction.
CO2	Apply nearest neighbor-based models, including K-Nearest Neighbor, radius-based methods, and regression techniques, to solve classification and regression problems.
CO3:	Analyze decision tree-based models, random forests, and Bayes classifiers for classification and regression, and understand bias-variance trade-offs.
CO4:.	Apply linear discriminant techniques, perceptrons, support vector machines (SVM),

	logistic regression, and multilayer perceptrons (MLPs) for supervised learning tasks.
CO5: .	Demonstrate clustering techniques including partitional, hierarchical, fuzzy, rough, and expectation-maximization-based clustering methods for unsupervised learning.
CO6:	Evaluate the performance of different machine learning models using appropriate metrics and select suitable models for real-world datasets.

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	1	1								
CO2	3	3	2	1								
CO3	3	3	2	2								
CO4	3	3	3	2								
CO5	3	3	2	1								
CO6	3	3	2	2								3

UNIT-I: Introduction to Machine Learning: Evolution of Machine Learning, Paradigms for ML, Learning by Rote, Learning by Induction, Reinforcement Learning, Types of Data, Matching, Stages in Machine Learning, Data Acquisition, Feature Engineering, Data Representation, Model Selection, Model Learning, Model Evaluation, Model Prediction, Search and Learning, Data Sets.

UNIT-II: Nearest Neighbor-Based Models: Introduction to Proximity Measures, Distance Measures, Non-Metric Similarity Functions, Proximity Between Binary Patterns, Different Classification Algorithms Based on the Distance Measures ,K-Nearest Neighbor Classifier, Radius Distance Nearest Neighbor Algorithm, KNN Regression, Performance of Classifiers, Performance of Regression Algorithms.

UNIT-III: Models Based on Decision Trees: Decision Trees for Classification, Impurity Measures, Properties, Regression Based on Decision Trees, Bias–Variance Trade-off, Random Forests for Classification and Regression. The Bayes Classifier: Introduction to the Bayes Classifier, Bayes' Rule and Inference, The Bayes Classifier and its Optimality, Multi- Class Classification, Class Conditional Independence and Naive Bayes Classifier (NBC)

UNIT-IV: Linear Discriminants for Machine Learning: Introduction to Linear Discriminants, Linear Discriminants for Classification, Perceptron Classifier, Perceptron Learning Algorithm, Support Vector Machines, Linearly Non-Separable Case, Non-linear SVM, Kernel Trick, Logistic Regression, Linear Regression, Multi-Layer Perceptrons (MLPs), Backpropagation for Training an MLP.

UNIT-V: Clustering : Introduction to Clustering, Partitioning of Data, Matrix Factorization, Clustering of Patterns, Divisive Clustering, Agglomerative Clustering, Partitional Clustering, K-

Means Clustering, Soft Partitioning, Soft Clustering, Fuzzy C-Means Clustering, Rough Clustering, Rough K-Means Clustering Algorithm, Expectation Maximization-Based Clustering, Spectral Clustering.

TextBooks:"MachineLearningTheoryandPractice",MNMurthy,VSAnanthanarayana, UniversitiesPress(India),2024

ReferenceBooks:

1. "MachineLearning", TomM.Mitchell,McGraw-HillPublication,2017
2. "MachineLearningin Action",PeterHarrington,DreamTech
3. "IntroductiontoDataMining",Pang-NingTan,MichelStenbach,VipinKumar,7th Edition, 2019.

D.N.R. COLLEGE OF ENGINEERING & TECHNOLOGY(Autonomous)
BALUSUMUDI, BHIMAVARAM, W.G. Dist., A.P., PIN-534 202

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

IIIYear-IISemester	Course Code: BT24CS32P3A	L	T	P	C
		0	1	2	2
SOFTWAREPROJECTMANAGEMENT					

Course Objectives:

Attheend ofthe course,the studentshall beable to:

- To describe and determine the purpose and importance of project management from the perspectives of planning, tracking and completion of project
- Tocompare anddifferentiateorganizationstructuresandprojectstructures
- To implement a project to manage project schedule, expenses and resources with the application of suitable project management tools

CO1:.	Explain conventional and modern software management principles, including the software lifecycle, software economics, and iterative process adoption.
CO2	Analyze software lifecycle phases and artifacts, including engineering, management, and programmatic artifacts, to support effective project planning.
CO3:	Apply model-based software architectures, process workflows, iterative process planning, milestones, and checkpoint assessment for project execution.
CO4:.	Demonstrate project organization structures, process automation techniques, and software metrics for project control, monitoring, and quality assurance.
CO5: .	Explain Agile methodology, Scrum adoption patterns, and iterating towards agility for software development projects.
CO6:	Analyze DevOps principles, architecture, delivery pipeline, tool stack, and adoption strategies for continuous integration, delivery, and deployment in projects.

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	1	1								
CO2	3	3	2	1								
CO3	3	3	3	2								
CO4	3	2	2	2	2							
CO5	3	2	1	1								2
CO6	3	3	2	2	2							3

UNIT-I:

Conventional Software Management: The waterfall model, conventional software Management performance.

Evolution of Software Economics: Software Economics, pragmatic software costestimation.

Improving Software Economics: Reducing Software product size, improving software processes, improving team effectiveness, improving automation, Achieving required quality, peer inspections.

The old way and the new: The principles of conventional software Engineering, principles of modern software management, transitioning to an iterative process.

UNIT-II:

Life cycle phases: Engineering and production stages, inception, Elaboration, construction, transition phases.

Artifacts of the process: The artifact sets, Management artifacts, Engineering artifacts, programmatic artifacts.

UNIT-III:

Modelbasedsoftwarearchitectures: AManagementperspectiveandtechnicalperspective.

Work Flows of the process: Software process workflows, Iteration workflows.**Checkpointsoftheprocess:Major**milestones,MinorMilestones,Periodicstatus assessments.

IterativeProcessPlanning: Workbreakdownstructures,planningguidelines,costand schedule estimating, Iteration planning process, Pragmatic planning.

UNIT-IV:

ProjectOrganizationsandResponsibilities: Line-of-BusinessOrganizations,Project Organizations, evolution of Organizations.

ProcessAutomation: AutomationBuildingblocks,TheProject Environment.

Project Control and Process instrumentation: The seven core Metrics, Management indicators, quality indicators, life cycle expectations, pragmatic Software Metrics, Metrics automation.

UNIT-V:

Agile Methodology, ADAPTing to Scrum, Patterns for Adopting Scrum, Iterating towards Agility. **Fundamentals of DevOps**: Architecture, Deployments, Orchestration, Need, Instance of applications, DevOps delivery pipeline, DevOps eco system. DevOps adoption in projects: Technology aspects, Agiling capabilities, Tool stack implementation, People aspect, processes

TextBooks:

1. SoftwareProjectManagement,WalkerRoyce,PEA,2005.
2. Succeeding with Agile: Software Development Using Scrum, Mike Cohn, Addison Wesley.
3. The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations, Gene Kim , John Willis , Patrick Debois , Jez Humb,1st Edition, O'Reilly publications, 2016.

ReferenceBooks:

1. SoftwareProjectManagement,BobHughes,3/e,MikeCotterell,TMH
2. SoftwareProjectManagement,JoelHenry,PEA
3. SoftwareProjectManagementinpractice,PankajJalote,PEA,2005,
4. EffectiveSoftwareProjectManagement,RobertK.Wysocki,Wiley,2006.
5. ProjectManagementinIT,KathySchwalbe,Cengage

D.N.R. COLLEGE OF ENGINEERING & TECHNOLOGY(Autonomous)
BALUSUMUDI, BHIMAVARAM, W.G. Dist., A.P., PIN-534 202

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

IIIYear-IISemester	Course Code: BT24CS32P3B	L	T	P	C
		0	1	2	2

MOBILEADHOCNETWORKS

Course Objectives:

Fromthe coursethe student will learn

- Architectsensornetworksforvariousapplicationsetups.
- Deviseappropriatedatadisseminationprotocolsandmodellinkscost.
- Understandingofthefundamentalconceptsofwirelesssensornetworksandhasa basic knowledge of the various protocols at various layers.
- Evaluatetheperformanceof sensornetworksandidentify bottlenecks.

CO1:.	Explain the fundamentals of Ad Hoc Wireless Networks, including characteristics, applications, challenges, and MAC protocol classifications.
CO2	Analyze routing protocols for MANETs, including topology-based, position-based approaches, and transport layer solutions for TCP and other protocols.
CO3:	Evaluate security requirements and protocols in Ad Hoc Wireless Networks, including key management, secure routing, cooperation, and intrusion detection systems.
CO4:.	Explain the architecture, design issues, communication, and energy considerations in Wireless Sensor Networks (WSNs), including sensor clustering and data retrieval mechanisms.
CO5: .	Apply network layer, transport layer, MAC layer, and application layer protocols to design efficient WSN communication and data aggregation systems.
CO6:	Analyze security mechanisms, key management, secure data aggregation, sensor network operating systems, programming languages (nesC, TinyGALS), and simulation tools (NS-2, TOSSIM) for WSNs.

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	1	1								
CO2	3	3	2	2								
CO3	3	3	2	2				1				3
CO4	3	2	1	1								
CO5	3	3	2	2	2							
CO6	3	3	2	2	2			1				3

UNIT I: Introduction to Ad Hoc Wireless Networks- Cellular and Ad Hoc Wireless Networks, Characteristics of MANETs, Applications of MANETs, Issues and Challenges of MANETs, Ad Hoc Wireless Internet, MAC protocols for Ad hoc Wireless Networks-Issues, Design Goals and Classifications of the MAC Protocols.

UNIT II: Routing Protocols for Ad Hoc Wireless Networks- Issues in Designing aRouting Protocol, Classifications of Routing Protocols, Topology-based versus Position- based Approaches, Issues and design goals of a Transport layer protocol, Classification of Transport layer solutions, TCP over Ad hoc Wireless Networks, Solutions for TCP over Ad Hoc Wireless Networks, Other Transport layer protocols.

UNIT III: Security protocols for Ad hoc Wireless Networks- Security in Ad hoc Wireless Networks, Network Security Requirements, Issues and Challenges in Security Provisioning, Network Security Attacks, Key Management, Secure Routing in Ad hoc Wireless Networks, Cooperation in MANETs, Intrusion Detection Systems.

UNIT IV: Basics of Wireless Sensors and Applications- The Mica Mote, Sensing and Communication Range, Design Issues, Energy Consumption, Clustering of Sensors, Applications, Data Retrieval in Sensor Networks-Classification of WSNs, MAC layer, Routing layer, Transport layer, High-level application layer support, Adapting to the inherent dynamic nature of WSNs.

UNIT V: Security in WSNs- Security in WSNs, Key Management in WSNs, Secure Data Aggregation in WSNs, Sensor Network Hardware-Components of Sensor Mote, Sensor Network Operating Systems–TinyOS, LA-TinyOS, SOS, RETOS, Imperative Language- nesC, **Dataflow Style Language-**TinyGALS, Node-Level Simulators, NS-2 and its sensor network extension, TOSSIM.

TextBooks:

1. AdHocWirelessNetworks–ArchitecturesandProtocols,1st edition,C.SivaRam Murthy, B. S. Murthy, Pearson Education, 2004
2. Ad Hoc and Sensor Networks – Theory and Applications, 2nd edition *Carlos Corderio Dharma P.Aggarwal,* World Scientific Publications / Cambridge University Press, March 2006

ReferenceBooks:

1. Wireless Sensor Networks: An Information Processing Approach, 1st edition, *Feng Zhao, Leonidas Guibas,*Elsevier Science imprint, Morgan Kauffman Publishers,2005, rp2009
2. Wireless Ad hoc Mobile Wireless Networks – Principles, Protocols and Applications, 1stedition,SubirKumarSarkar,etal.,AuerbachPublications,Taylor&FrancisGroup, 2008
3. AdhocNetworking,1stedition,*CharlesE.Perkins,*PearsonEducation,2001

4. Wireless Ad hoc Networking, 1st edition, *Shih-Lin Wu, Yu-Chee Tseng,* Auerbach Publications, Taylor & Francis Group, 2007
5. Wireless Sensor Networks – Principles and Practice, 1st edition, Fei Hu, Xiaojun Cao, An Auerbach book, CRC Press, Taylor & Francis Group, 2010

D.N.R. COLLEGE OF ENGINEERING & TECHNOLOGY(Autonomous)
BALUSUMUDI, BHIMAVARAM, W.G. Dist., A.P., PIN-534 202

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

IIIYear-IISemester	**Course Code:** BT24CS32P3C	**L**	**T**	**P**	**C**
		0	**1**	**2**	**2**
NATURALLANGUAGEPROCESSING					

Course Objectives:

Thiscourseintroducesthefundamentalconceptsandtechniquesofnaturallanguage processing (NLP).

- Students will gain an in-depth understanding of the computational properties of natural languages and the commonly used algorithms for processing linguistic information.
- The course examines NLP models and algorithms using both the traditional symbolic and the more recent statistical approaches.
- Enablestudentstobecapabletodescribetheapplicationbasedonnaturallanguage processing and to show the points of syntactic, semantic and pragmatic processing.

CO1:.	Explain the fundamentals of Natural Language Processing (NLP), including language modeling, regular expressions, finite-state automata, morphology, tokenization, and spelling correction.
CO2	Apply word-level analysis techniques, including N-gram models, smoothing methods, Part-of-Speech tagging, and statistical tagging approaches like Hidden Markov Models and Maximum Entropy models.
CO3:	Analyze syntactic structures using Context-Free Grammars, dependency grammars, parsing techniques, probabilistic CFGs, and feature-based unification approaches.
CO4:.	Demonstrate semantic and pragmatic analysis, including first-order logic, description logics, word sense disambiguation, thematic roles, and selectional restrictions.
CO5: .	Apply discourse analysis methods, including discourse segmentation, coherence, anaphora resolution, coreference resolution, and use of lexical resources such as WordNet, PropBank, and FrameNet.
CO6:	Evaluate NLP models and techniques using real-world datasets and resources, integrating multiple levels of language analysis for practical NLP applications.

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	1	1								
CO2	3	3	2	1								
CO3	3	3	3	2								
CO4	3	3	2	2								
CO5	3	3	2	2								
CO6	3	3	2	2								3

UNITI:

INTRODUCTION: Origins and challenges of NLP – Language Modeling: Grammar-based LM, Statistical LM – Regular Expressions, Finite-State Automata – English Morphology, Transducers for lexicon and rules, Tokenization, Detecting and Correcting Spelling Errors, Minimum Edit Distance.

UNITII:

WORD LEVEL ANALYSIS: Unsmoothed N-grams, Evaluating N-grams, Smoothing, Interpolation and Backoff – Word Classes, Part- of-Speech Tagging, Rule-based, Stochastic and Transformation-based tagging, Issues in PoS tagging – Hidden Markov and Maximum Entropy models.

UNITIII:

SYNTACTIC ANALYSIS: Context-Free Grammars, Grammar rules for English,Treebanks, Normal Forms for grammar – Dependency Grammar – Syntactic Parsing, Ambiguity, Dynamic Programming parsing – Shallow parsing Probabilistic CFG, Probabilistic CYK, Probabilistic Lexicalized CFGs – Feature structures, Unification offeature structures

UNITIV:

SEMANTICS AND PRAGMATICS: Requirements for representation, First-Order Logic, Description Logics – Syntax-Driven Semantic analysis, Semantic attachments – WordSenses, Relations between Senses, Thematic Roles, selectional restrictions – Word Sense Disambiguation, WSD using Supervised, Dictionary & Thesaurus, Bootstrapping methods – Word Similarity using Thesaurus and Distributional methods.

UNITV:

DISCOURSE ANALYSIS AND LEXICAL RESOURCES: Discourse segmentation, Coherence – Reference Phenomena, Anaphora Resolution using Hobbs and Centering Algorithm–CoreferenceResolution–Resources:PorterStemmer,Lemmatizer,Penn Treebank,Brill'sTagger,WordNet,PropBank,FrameNet,BrownCorpus,BritishNational Corpus(BNC).

TextBooks:

1. Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech, 2nd Edition, Daniel Jurafsky, James H. Martin - Pearson Publication,2014.
2. Natural Language Processing with Python, First Edition, Steven Bird, Ewan Klein and Edward Loper, OReilly Media,2009.

ReferenceBooks:

1. Language Processingwith Java and Ling Pipe Cookbook, 1st Edition, Breck Baldwin,Atlantic Publisher, 2015.
2. NaturalLanguageProcessingwithJava,2nd Edition,RichardMReese,OReilly Media,2015.
3. HandbookofNaturalLanguageProcessing,Second,NitinIndurkhyaandFredJ. Damerau, Chapman and Hall/CRC Press, 2010.Edition
4. NaturalLanguageProcessingandInformationRetrieval,3rd Edition,TanveerSiddiqui, U.S.Tiwary,Oxford UniversityPress,2008.

D.N.R. COLLEGE OF ENGINEERING & TECHNOLOGY(Autonomous)
BALUSUMUDI, BHIMAVARAM, W.G. Dist., A.P., PIN-534 202

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

IIIYear-IISemester	**Course Code:** BT24CS32P3D	**L**	**T**	**P**	**C**
		0	**1**	**2**	**2**

BIGDATA ANALYTICS

CourseObjectives: Thiscourseisaimedat enablingthestudentsto

- Toprovidean overview ofanexciting growingfieldof bigdata analytics.
- TointroducethetoolsrequiredtomanageandanalyzebigdatalikeHadoop,NoSQL, Map Reduce, HIVE, Cassandra, Spark.
- To teach the fundamental techniques and principles in achieving big data analytics withscalability and streaming capability.
- TooptimizebusinessdecisionsandcreateCompetitiveadvantagewithBigDataanalytics

CO1:.	Explain the fundamentals of big data, key trends, applications across industries, analytics techniques, and associated technologies including Hadoop, cloud, and mobile business intelligence.
CO2	Apply NoSQL database concepts, including aggregate data models, key-value, document, graph databases, sharding, replication, and consistency models, with practical experience in Cassandra.
CO3:	Analyze Hadoop architecture, including HDFS, MapReduce, fault tolerance, high availability, data locality, data integrity, and Hive for data processing, querying, and optimization.

CO4:.	Demonstrate Apache Spark concepts, including in-memory processing, RDDs, DataFrames, transformations, actions, Spark SQL, Catalyst optimizer, and cluster deployment strategies.
CO5: .	Implement performance tuning in Spark and utilize Spark features for large-scale data processing, debugging, and cluster management.
CO6:	Apply stream processing techniques using Spark Structured Streaming, including event-time processing, stateful operations, windowing, handling late data, and transformations on streams for real-time analytics.

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	1	1								
CO2	3	3	2	1	2							
CO3	3	3	3	2	2							
CO4	3	3	3	2	2							
CO5	3	3	2	2	3							
CO6	3	3	2	2	3							3

UNIT I: big data, convergence of key trends, unstructured data, industry examples of big data, web analytics, big data and marketing, fraud and big data, risk and big data, credit risk management, big data and algorithmic trading, big data and healthcare, big data in medicine, advertising and big data, big data technologies, introduction to Hadoop, open source technologies, cloud and big data, mobile business intelligence, Crowd sourcing analytics,inter and trans firewall analytics.

UNIT II: Introduction to NoSQL, aggregate data models, aggregates, key-value and document data models, relationships, graph databases, schema less databases, materialized views, distribution models, sharding, master-slave replication, peer- peer replication, shardingand replication, consistency, relaxing consistency, version stamps, Working with Cassandra ,Table creation, loading and reading data.

UNIT III: Data formats, analyzing data with Hadoop, scaling out, Architecture of Hadoop distributed file system (HDFS), fault tolerance ,with data replication, High availability, Data locality , Map ReduceArchitecture, Process flow, Javainterface, data flow, Hadoop I/O, data integrity, compression, serialization. Introduction to Hive, data types and file formats, HiveQL data definition, HiveQL data manipulation, Logical joins, Window functions, Optimization, Table partitioning, Bucketing, Indexing, Join strategies.

UNIT IV: Apache spark- Advantages over Hadoop, lazy evaluation, In memory processing, DAG, Spark context, Spark Session, RDD, Transformations- Narrow and Wide, Actions,Dataframes,RDDtoDataframes,Catalystoptimizer,DataFrameTransformations,Working with Dates and Timestamps, Working with Nulls in Data, Working with Complex Types,

Working with JSON, Grouping, Window Functions, Joins, Data Sources, Broadcast Variables, Accumulators, Deploying Spark- On-Premises Cluster Deployments, Cluster Managers-StandaloneMode, Sparkon YARN, SparkLogs, TheSparkUI-SparkUIHistory Server, Debugging and Spark First Aid

UNIT V: Spark-Performance Tuning, Stream Processing Fundamentals, Event-Time and StatefullProcessing -EventTime,StatefullProcessing,WindowsonEventTime-Tumbling Windows,HandlingLateDatawithWatermarks,DroppingDuplicatesinaStream,
Structured Streaming Basics - Core Concepts, Structured Streaming in Action, Transformations on Streams, Input and Output.

TextBooks:

1. BigData,BigAnalytics:Emerging,MichaelMinnelli,MichelleChambers,and AmbigaDhiraj, 1st edition ,2013
2. SPARK:TheDefinitiveGuide,BillChambers&MateiZaharia,O'Reilley,2018-first Edition.
3. Business Intelligence and Analytic Trends for Today's Businesses", Wiley, First edition-2013.
4. P. J. Sadalage and M. Fowler, "NoSQL Distilled: A Brief Guide to the Emerging World Polyglot Persistence", Addison-Wesley Professional, 2012
5. TomWhite, "Hadoop: TheDefinitiveGuide", Third Edition, O'Reilley, 2012

ReferenceBooks:

1. "HadoopOperations",O'Reilley,EricSammer,FirstEdition-2012.
2. "ProgrammingHive",O'Reilley,E. Capriolo,D.Wampler,andJ.Rutherglen,2012.
3. "HBase:TheDefinitiveGuide",O'Reilley,Lars George,September2011:First Edition..
4. "Cassandra: The Definitive Guide", O'Reilley, Eben Hewitt, 2010.

"ProgrammingPig",O'Reilley,AlanGates,October2011:FirstEdition

D.N.R. COLLEGE OF ENGINEERING & TECHNOLOGY(Autonomous)
BALUSUMUDI, BHIMAVARAM, W.G. Dist., A.P., PIN-534 202

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

IIIYear-IISemester	Course Code: BT24CS32P3E	L	T	P	C
		0	1	2	2

DISTRIBUTED OPERATING SYSTEM

Course Objectives:

The main objective of the course is to introduce design issues and different message passing techniques in DOS, distributed systems, RPC implementation and its performance in DOS, distributed shared memory and resource management, distributed file systems and evaluate the performance in terms of fault tolerance, file replication as major factors

CO1:.	Explain the fundamentals of distributed computing systems, including system models, distributed operating systems, design issues, and Distributed Computing Environment (DCE).
CO2	Apply message passing concepts, including synchronization, buffering, process addressing, failure handling, group communication, and IPC mechanisms in distributed systems.
CO3:	Demonstrate Remote Procedure Calls (RPCs), including RPC models, transparency, stub generation, parameter passing, client-server binding, exception handling, security, and performance optimization.
CO4:.	Analyze Distributed Shared Memory (DSM) systems, including architecture, consistency models, synchronization mechanisms, clock/event ordering, mutual exclusion, and election algorithms.
CO5: .	Evaluate resource management techniques in distributed systems, including global scheduling, task assignment, load balancing, load sharing, process migration, and threads management.
CO6:	Explain distributed file systems, including file models, access models, file-sharing semantics, caching, replication, fault tolerance, and atomic transactions.

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	1	1								
CO2	3	3	2	2								
CO3	3	3	3	2	2							
CO4	3	3	2	2								
CO5	3	3	2	2	2							
CO6	3	2	2	2	2							3

Unit I:

Fundamentals:

What is Distributed Computing Systems? Evolution of Distributed Computing System; Distributed Computing System Models; What is Distributed Operating System? Issues in Designing a Distributed Operating System; Introduction to Distributed Computing Environment(DCE).

Message Passing:

Introduction, Desirable features of a Good Message Passing System, Issues in PC byMessage Passing, Synchronization, Buffering, Multi-datagram Messages, Encoding and Decoding of Message Data, Process Addressing, Failure Handling, Group Communication, Case Study: 4.3 BSD UNIX IPC Mechanism.

UnitII:RemoteProcedureCalls:

Introduction, The RPC Model, Transparency of RPC, Implementing RPC Mechanism, Stub Generation, RPC Messages, Marshaling Arguments and Results, Server Management, Parameter-Passing Semantics, Call Semantics, Communication Protocols for RPCs, Complicated RPCs, Client-Server Binding, Exception Handling, Security, Some Special Types of RPCs, RPC in Heterogeneous Environments, Lightweight RPC, Optimization for Better Performance, Case Studies: Sun RPC

UnitIII:DistributedShared Memory:

Introduction, General Architecture of DSM systems, Design and Implementation Issues of DSM, Granularity, Structure of Shared Memory Space, Consistency Models, Replacement Strategy, Thrashing, Other approaches to DSM, Heterogeneous DSM, Advantages of DSM. Synchronization: Introduction, Clock Synchronization, Event Ordering, Mutual Exclusion, Dead Lock, Election Algorithms

UnitIV:Resource Management:

Introduction, Desirable Features of a Good Global Scheduling Algorithm, Task Assignment

Approach, Load – Balancing Approach, Load – Sharing Approach Process Management: Introduction, Process Migration, Threads.

UnitV:DistributedFile Systems:

Introduction, Desirable Features of a Good Distributed File System, File models, File–AccessingModels,File–SharingSemantics,File –CachingSchemes,FileReplication,Fault Tolerance, Atomic Transactions and Design Principles.

Textbooks

1. Pradeep.K.Sinha:DistributedOperatingSystems:ConceptsandDesign,PHI,2007.

ReferenceBooks:

1. AndrewS.Tanenbaum:DistributedOperatingSystems,PearsonEducation,2013.
2. AjayD. KshemkalyaniandMukeshSinghal,DistributedComputing:Principles,Algorithms and Systems, Cambridge University Press, 2008
3. SunitaMahajan,Seema Shan,"DistributedComputing",OxfordUniversityPress,2015

D.N.R. COLLEGE OF ENGINEERING & TECHNOLOGY(Autonomous)
BALUSUMUDI, BHIMAVARAM, W.G. Dist., A.P., PIN-534 202

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

IIIYear-IISemester	Course Code: BT24CS3204	L	T	P	C
		0	1	2	2
CLOUDCOMPUTING LAB					

Course Objectives:

- Tointroducethevariouslevelsofservicesofferedbycloud.
- Togivepracticalknowledgeaboutworkingwithvirtualizationand containers.
- Tointroducetheadvancedconceptssuchasserverlesscomputingandcloud simulation.

CourseOutcomes: At theendofthe course, thestudentshould beable to

- Demonstratevariousservicetypes,deliverymodelsandtechnologiesofacloud computing environment.
- Distinguishtheservicesbasedonvirtualmachinesandcontainersinthecloud offerings.
- Assessthechallengesassociated withacloud-based application.
- Discussadvancedcloudconceptssuch asserverlesscomputingandcloudsimulation.
- Examine various programming paradigms suitable to solve real world and scientificproblems using cloud services.

List ofExperiments:

1. Labonwebservices
2. Labon IPC,messaging,publish/subscribe
3. InstallVirtual Box/VMwareWorkstationwithdifferentflavoursof LinuxorwindowsOS on top of windows8 or above.
4. Install a C compiler in the virtual machine created using Virtual Box and execute Simple Programs.
5. Create an Amazon EC2 instance and set up a web-server on the instance and associate an IP address with the instance. In the process, create a security group allowing access toport 80 on the instance.

OR

6. Dothesamewith OpenStack
7. Install Google App Engine. Create a hello world app and other simple web applications using python/java.
8. Start a Docker container and set up a web-server (e.g. apache2 or Python based Flask micro web framework) on the instance. Map the host directory as a data volume for the container.
9. Find aprocedureto transferthe files from onevirtual machineto anothervirtual machine. Similarly, from one container to another container.
10. Find a procedure to launch virtual machine using trystack (Online Openstack Demo Version)
11. InstallHadoopsinglenodeclusterandrunsimpleapplicationslikewordcount.
12. Utilize OpenFaaS – Serverless computing framework and demonstrate basic event driven function invocation.
13. SimulateacloudscenariousingCloudSimandrunaschedulingalgorithmthatisnot present in CloudSim.

TextBooks:

1. MasteringCloudComputing,2nd edition,RajkumarBuyya,ChristianVecchiola,Thamarai Selvi, Shivananda Poojara, Satish N. Srirama, McGraw Hill, 2024.
2. Distributed and Cloud Computing, Kai Hwang,Geoffery C. Fox, Jack J. Dongarra,Elsevier, 2012.

ReferenceBooks:

1. Cloud Computing, Theory and Practice, Dan C Marinescu, 2nd edition, MK Elsevier, 2018.
2. Cloud Computing: Principles and Paradigms by Rajkumar Buyya, James Broberg and Andrzej M. Goscinski, Wiley, 2011.
3. Online documentation and tutorials from cloud service providers (e.g. AWS, Google App Engine)
4. Docker,Referencedocumentation, https://docs.docker.com/reference/
5. OpenFaaS,ServerlessFunctionsMadeSimple, https://docs.openfaas.com/

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

IIIYear-IISemester	Course Code: BT24CS3205	L	T	P	C
		0	1	2	2
CRYPTOGRAPHY & NETWORK SECURITY LAB					

Course Objectives:

- Tolearnbasicunderstandingofcryptography,howithasevolved,andsomekey encryption techniques used today.
- TounderstandandimplementencryptionanddecryptionusingCeaserCipher, Substitution Cipher, Hill Cipher.

List ofExperiments:

1. WriteaCprogramthatcontainsastring(charpointer)withavalue\HelloWorld'.The programshouldXOReachcharacterinthisstringwith0anddisplaysthe result.
2. Write a C program that contains a string (char pointer) with a value \Hello World'. The program should AND or and XOR each character in this string with 127 and display the result
3. Write a Java program to perform encryption and decryption using the following algorithms:
   a) CeaserCipher
   b) Substitution Cipher
   c) Hill Cipher
4. WriteaJavaprogram toimplement theDES algorithm logic
5. WriteaC/JAVAprogramtoimplementtheBlowFishalgorithmlogic
6. WriteaC/JAVAprogramtoimplementtheRijndaelalgorithmlogic.
7. UsingJavaCryptography,encryptthetext"Helloworld"usingBlowFish.Createyour ownkeyusingJavakey tool.
8. WriteaJavaprogramto implement RSAAlgorithm
9. Implement the Diffie-Hellman Key Exchange mechanism using HTML and JavaScript. Considertheend useras oneoftheparties(Alice)andtheJavaScriptapplicationasother party (bob).
10. Calculatethemessagedigest ofatext usingthe SHA-1algorithm in JAVA.

D.N.R. COLLEGE OF ENGINEERING & TECHNOLOGY(Autonomous)
BALUSUMUDI, BHIMAVARAM, W.G. Dist., A.P., PIN-534 202

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

IIIYear-IISemester	Course Code:	L	T	P	C
		0	1	2	2
SOFT SKILLS					

Course Objectives:

- Toequipthestudentswiththeskillstoeffectively communicateinEnglish
- Totrainthestudentsininterviewskills,groupdiscussionsandpresentationskills
- Tomotivatethe studentstodevelop confidence
- Toenhancethestudents' interpersonalskills
- Toimprovethe students'writing skills

UNIT–I

Analytical Thinking & Listening Skills: Self-Introduction, Shaping Young Minds -ATalk by Azim Premji (Listening Activity), Self – Analysis, Developing Positive Attitude, Perception.

Communication Skills: Verbal Communication; Non Verbal Communication (Body Language)

UNIT–II

Self-Management Skills: Anger Management, Stress Management, Time Management, Six Thinking Hats, Team Building, Leadership Qualities

Etiquette: SocialEtiquette,BusinessEtiquette,TelephoneEtiquette,DiningEtiquette

UNIT–III

Standard Operation Methods : Basic Grammars, Tenses, Prepositions, Pronunciation, Letter Writing; Note Making, Note Taking,Minutes Preparation, Email & Letter Writing

UNIT-IV

Job-Oriented Skills: Group Discussion, Mock Group Discussions, Resume Preparation, Interview Skills, Mock Interviews

UNIT-V

Interpersonal relationships: Introduction, Importance, Types, Uses,Factors affecting interpersonal relationships, Accommodating different styles, Consequences of interpersonal relationships

Textbooks:

1. Barun K. Mitra, Personality Development and Soft Skills, Oxford University Press,2011.
2. S.P.Dhanavel,EnglishandSoftSkills,OrientBlackswan,2010.

Referencebooks:

1. R.S.Aggarwal, A Modern Approachto Verbal&Non-VerbalReasoning,S.Chand& Company Ltd., 2018.
2. Raman,Meenakshi&Sharma,Sangeeta,TechnicalCommunicationPrinciplesand Practice, Oxford University Press, 2011.

E-resources:

1. https://swayam-plus.swayam2.ac.in/courses/course-details?id=P_CAMBR_01

D.N.R. COLLEGE OF ENGINEERING & TECHNOLOGY(Autonomous)
BALUSUMUDI, BHIMAVARAM, W.G. Dist., A.P., PIN-534 202

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

IIIYear-IISemester	Course Code: BT24CS3206	L	T	P	C
		0	1	2	2

TECHNICAL PAPER WRITING & IPR

Course Objective : The course will explain the basic related to writing the technical reports andunderstandingtheconceptsrelatedtoformattingandstructuringthereport.Thiswillhelp students to comprehend the concept of proofreading, proposals and practice

Unit I:

Introduction: An introduction to writing technical reports, technical sentences formation, using transitions to join sentences, Using tenses for technical writing.

PlanningandStructuring: Planning the report, identifying reader(s), Voice, Formatting and structuring the report, Sections of a technical report, Minutes of meeting writing.

Unit II:

Draftingreportanddesignissues: Theuseofdrafts, Illustrationsand graphics.

Final edits: Grammar, spelling, readability and writing in plain English: Writing in plain English, Jargon and final layout issues, Spelling, punctuation and Grammar, Padding, Paragraphs, Ambiguity.

Unit III:

Proofreading and summaries: Proofreading, summaries, Activities on summaries. **Presenting final reports:** Printed presentation, Verbal presentation skills, Introduction to proposals and practice.

UnitIV:Usingwordprocessor:

Adding a Table of Contents, Updating the Table of Contents, Deleting the Table of Contents, Adding an Index, Creating an Outline, Adding Comments, Tracking Changes, Viewing Changes, Additions, and Comments, Accepting and Rejecting Changes , Working with Footnotes and Endnotes, Inserting citations and Bibliography, Comparing Documents, Combining Documents, Mark documents final and make them read only., Password protect Microsoft Word documents., Using Macros,

Unit V:

Nature of Intellectual Property: Patents, Designs, Trade and Copyright. Process of **Patenting and Development:** technological research, innovation, patenting, development. International Scenario: International cooperation on Intellectual Property

TextBooks:

1. Kompal Bansal & Parshit Bansal, "Fundamentals of IPR for Beginner's", 1st Ed., BS Publications, 2016.
2. WilliamS.PfeifferandKayeA.Adkins,"TechnicalCommunication:APractical Approach", Pearson.
3. Ramappa,T.,"IntellectualPropertyRightsUnderWTO",2ndEd.,SChand,2015.

ReferenceBooks:

1. AdrianWallwork,EnglishforWritingResearchPapers,SpringerNewYork Dordrecht Heidelberg London, 2011.
2. DayR,HowtoWriteandPublishaScientificPaper,Cambridge
3. University Press(2006)

E-resources:

1. https://www.udemy.com/course/reportwriting/
2. https://www.udemy.com/course/professional-business-english-and-technical-report-writing/
3. https://www.udemy.com/course/betterbusinesswriting/