



NALLA NARASIMHA REDDY
Education Society's Group of Institutions - Integrated Campus
Near Narapally, Chowdariguda (Village), Korremula 'X' Road, Ghatkesar (Mandal), Medchal District, Hyderabad - 500 088, Telangana.



B.Tech. in INFORMATION TECHNOLOGY
COURSE STRUCTURE, III & IV-YEAR SYLLABUS (R22 Regulations)

Applicable from A.Y. 2022-23 Batch

I Year I Semester

S. No.	Course Code	Course Title	L	T	P	Credits
1.	22MA101BS	Matrices and Calculus	3	1	0	4
2.	22CH102BS	Engineering Chemistry	3	1	0	4
3.	22CS103ES	Programming for Problem Solving	3	0	0	3
4.	22EE104ES	Basic Electrical Engineering	2	0	0	2
5.	22ME105ES	Computer Aided Engineering Graphics	1	0	4	3
6.	22CS106ES	Elements of Computer Science & Engineering	0	0	2	1
7.	22CH107BS	Engineering Chemistry Laboratory	0	0	2	1
8.	22CS108ES	Programming for Problem Solving Laboratory	0	0	2	1
9.	22EE109ES	Basic Electrical Engineering Laboratory	0	0	2	1
Total			12	2	12	20

I Year II Semester

S. No.	Course Code	Course Title	L	T	P	Credits
1.	22MA201BS	Ordinary Differential Equations and Vector Calculus	3	1	0	4
2.	22AP202BS	Applied Physics	3	1	0	4
3.	22ME203ES	Engineering Workshop	0	1	3	2.5
4.	22EN205HS	English for Skill Enhancement	2	0	0	2
5.	22EC205PC	Electronic Devices and Circuits	2	0	0	2
6.	22AP205BS	Applied Physics Laboratory	0	0	3	1.5
7.	22CS207ES	Python Programming Laboratory	0	1	2	2
8.	22EN207HS	English Language and Communication Skills Laboratory	0	0	2	1
9.	22CS209PC	IT Workshop	0	0	2	1
Total			10	4	12	20

II YEAR I SEMESTER

S. No.	Course Code	Course Title	L	T	P	Credits
1	22EC303ES	Digital Logic Design	3	0	0	3
2	22CS302PC	Data Structures	3	0	0	3
3	22MA303BS	Computer Oriented Statistical Methods	3	1	0	4
4	22IT304PC	Computer Organization and Microprocessor	3	0	0	3
5	22IT305PC	Introduction to IoT	2	0	0	2
6	22EC306ES	Digital Logic Design Lab	0	0	2	1
7	22CS306PC	Data Structures Lab	0	0	3	1.5
8	22IT308PC	Internet of Things Lab	0	0	3	1.5
9	22*MC309GS	Gender Sensitization Lab	0	0	2	0
10	22CS309PC	Skill Development Course	0	0	2	1
Total			14	1	12	20

II YEAR II SEMESTER

S. No.	Course Code	Course Title	L	T	P	Credits
1	22CS401PC	Discrete Mathematics	3	0	0	3
2	22SM402MS	Business Economics & Financial Analysis	3	0	0	3
3	22CS403PC	Operating Systems	3	0	0	3
4	22CS404PC	Database Management Systems	3	0	0	3
5	22IT405PC	Java Programming	2	0	0	2
6	22CS406PC	Operating Systems Lab	0	0	2	1
7	22CS407PC	Database Management Systems Lab	0	0	2	1
8	22IT408PC	Java Programming Lab	0	0	2	1
9	22CS408PC	Real-time Research Project/ Societal Related Project	0	0	4	2
10	22MC410CI	Constitution of India	3	0	0	0
11	22CS410PC	Skill Development Course	0	0	2	1
Total			17	0	12	20

III YEAR I SEMESTER

S. No.	Course Code	Course Title	L	T	P	Credits
1	22IT501PC	Software Engineering	3	0	0	3
2	22IT502PC	Data Communications and Computer Networks	3	1	0	4
3	22DS501PC	Machine Learning	3	0	0	3
4		Professional Elective – I	3	0	0	3
5		Professional Elective – II	3	0	0	3
6	22IT503PC	Software Engineering & Computer Networks Lab	0	0	2	1
7	22DS503PC	Machine Learning Lab	0	0	2	1
8	22EN508HS	Advanced English Communication Skills Lab	0	0	2	1
9	22MC509IP	Intellectual Property Rights	3	0	0	0
10	22CS506PC	Skill Development Course	0	0	2	1
Total			18	1	8	20

III YEAR II SEMESTER

S. No.	Course Code	Course Title	L	T	P	Credits
1	22DS601PC	Automata Theory and Compiler Design	3	0	0	3
2	22DS602PC	Algorithm Design and Analysis	3	0	0	3
3	22EC610PC	Embedded Systems	3	0	0	3
4		Professional Elective –III	0	0	2	1
5		Open Elective-I	3	0	0	3
6	22IT603PC	Compiler Design Lab	3	0	0	3
7	22EC611PC	Embedded Systems Lab	0	0	2	1
8		Professional Elective-III Lab	0	0	2	1
9	22CS606PC	Industrial Oriented Mini Project/ Internship/ Skill Development Course	0	0	4	2
10	22MC609ES	Environmental Science	3	0	0	0
Total			18	0	10	20

Environmental Science in III Yr II Sem Should be Registered by Lateral Entry Students Only.

IV YEAR I SEMESTER

S. No.	Course Code	Course Title	L	T	P	Credits
1	22IT701PC	Information Security	3	0	0	3
2	22IT702PC	Cloud Computing	3	0	0	3
3		Professional Elective -IV	3	0	0	3
4		Professional Elective -V	3	0	0	3
5		Open Elective-II	3	0	0	3
6	22IT703PC	Information Security Lab	0	0	2	1
7	22IT704PC	Cloud Computing Lab	0	0	2	1
8	22IT705PC	Project Stage - I	0	0	6	3
Total			15	0	10	20

IV YEAR II SEMESTER

S. No.	Course Code	Course Title	L	T	P	Credits
1	22SM801MS	Organizational Behavior	3	0	0	3
2		Professional Elective – VI	3	0	0	3
3		Open Elective – III	3	0	0	3
4	22IT802PC	Project Stage – II including Seminar	0	0	22	9+2
Total			9	0	22	20

*MC – Satisfactory/Unsatisfactory

#Skill Course – 1 credit with 2 Practical Hours

Professional Elective – I

22CS511PE	Quantum Computing
22CS512PE	Advanced Computer Architecture
22CS513PE	Data Analytics
22CS514PE	Image Processing
22CS515PE	Principles of Programming Languages

Professional Elective – II

22CS521PE	Computer Graphics
22IT522PE	Pattern Recognition
22IT523PE	Advanced Operating Systems
22CS524PE	Distributed Databases
22IT525PE	Biometrics

Professional Elective – III

22CS631PE	Full Stack Development
22IT632PE	Data Mining
22CS633PE	Scripting Languages
22CS634PE	Mobile Application Development
22CS635PE	Software Testing Methodologies

Courses in PE – III and PE – III Lab must be in 1-1 correspondence.

Professional Elective – IV

22IT741PE	Human Computer Interaction
22IT742PE	High Performance Computing
22IT743PE	Artificial Intelligence
22IT744PE	Information Retrieval Systems
22CS745PE	Ad-hoc & Sensor Networks

Professional Elective – V

22IT751PE	Intrusion Detection Systems
22IT752PE	Real Time Systems
22IT753PE	Deep Learning
22CS754PE	Blockchain Technology
22CS755PE	Software Process & Project Management

Professional Elective – VI

22IT861PE	Natural Language Processing
22CS862PE	Distributed Systems
22IT863PE	Augmented Reality & Virtual Reality
22IT864PE	Web Security
22CS865PE	Cyber Forensics

Skill Development Course**II Year – I Semester**

SD31	Data Visualization – R Programming
SD32	Power BI
SD33	Node JS
SD34	React JS
SD35	Django
SD36	Prolog
SD37	Lisp
SD38	Pyswip

II Year – II Semester

SD41	Data Visualization – R Programming
SD42	Power BI
SD43	Node JS
SD44	React JS
SD45	Django
SD46	Prolog
SD47	Lisp
SD48	Pyswip

III Year – I Semester

SD51	UI design - Flutter
SD52	ETL - Kafka
SD53	Talend
SD54	Big data - Spark
SD55	DevOps

III Year – II Semester

SD61	UI design – Flutter
SD62	ETL – Kafka
SD63	Talend
SD64	Big data – Spark
SD65	DevOps

Open Elective -1:

22IT611OE	Java Programming
22IT612OE	Object Oriented Programming using C++

Open Elective -2:

22IT721OE	Full Stack Development
22IT722OE	Scripting Languages

Open Elective -3:

22IT831OE	Big Data Technologies
22IT832OE	DevOps

22IT501PC: SOFTWARE ENGINEERING**B.Tech. III Year I Sem.**

L	T	P	C
3	0	0	3

Course Objectives

- The aim of the course is to provide an understanding of the working knowledge of the techniques for estimation, design, testing and quality management of large software development projects.
- Topics include process models, software requirements, software design, software testing, software process/product metrics, risk management, quality management and UML diagrams

Course Outcomes

- Ability to translate end-user requirements into system and software requirements, using e.g. UML, and structure the requirements in a Software Requirements Document (SRD).
- Identify and apply appropriate software architectures and patterns to carry out high level design of a system and be able to critically compare alternative choices.
- Will have experience and/or awareness of testing problems and will be able to develop a simple testing report

UNIT - I

Introduction to Software Engineering: The evolving role of software, changing nature of software, software myths.

A Generic view of process: Software engineering- a layered technology, a process framework, the capability maturity model integration (CMMI).

Process models: The waterfall model, Spiral model and Agile methodology

UNIT - II

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

Requirements engineering process: Feasibility studies, requirements elicitation and analysis, requirements validation, requirements management.

UNIT - III

Design Engineering: Design process and design quality, design concepts, the design model.

Creating an architectural design: software architecture, data design, architectural styles and patterns, architectural design, conceptual model of UML, basic structural modeling, class diagrams, sequence diagrams, collaboration diagrams, use case diagrams, component diagrams.

UNIT - IV

Testing Strategies: A strategic approach to software testing, test strategies for conventional software, black-box and white-box testing, validation testing, system testing, the art of debugging.

Metrics for Process and Products: Software measurement, metrics for software quality.

UNIT - V

Risk management: Reactive Vs proactive risk strategies, software risks, risk identification, risk projection, risk refinement, RMMM.

Quality Management: Quality concepts, software quality assurance, software reviews, formal technical reviews, statistical software quality assurance, software reliability, the ISO 9000 quality standards.

TEXT BOOKS:

1. Software Engineering, A practitioner's Approach- Roger S. Pressman, 6th edition, McGraw Hill International Edition.

2. Software Engineering- Sommerville, 7th edition, Pearson Education.

REFERENCE BOOKS:

1. The unified modeling language user guide Grady Booch, James Rumbaugh, Ivar Jacobson, Pearson Education.
2. Software Engineering, an Engineering approach- James F. Peters, Witold Pedrycz, John Wiley.
3. Software Engineering principles and practice- Waman S Jawadekar, The McGraw-Hill Companies.
4. Fundamentals of object-oriented design using UML Meiler page-Jones: Pearson Education.

22IT502PC: DATA COMMUNICATIONS AND COMPUTER NETWORKS

B.Tech. III Year I Sem.

L	T	P	C
3	1	0	4

Course Objectives:

- Explore basic concepts of data communication
- Introduce students to TCP/IP and OSI models along with their merits and demerits.
- Explore in detail services offered by various layers of OSI Model.
- Understand link layer protocols, UDP, TCP and application layer protocols.

Course Outcomes:

- Understand and explore the basics of communication and computer networks
- Understand datalink, network and transport layers concepts of a computer network.
- Understand the working of application layer protocols.

UNIT – I

Data Communications: Components - Direction of Data flow - Networks - Components and Categories - Types of Connections - Topologies -Protocols and Standards - ISO / OSI model, Example Networks such as ATM, Frame Relay, ISDN

Physical layer: Transmission modes, Multiplexing, Transmission Media, Switching, Circuit Switched Networks, Datagram Networks, Virtual Circuit Networks.

UNIT – II

Data link layer: Introduction, Framing, and Error - Detection and Correction - Parity - LRC - CRC Hamming code, Flow and Error Control, Noiseless Channels, Noisy Channels, HDLC, Point to Point Protocols. 111 Medium Access sub layer: ALOHA, CSMA/CD, LAN - Ethernet IEEE 802.3, IEEE 802.5 - IEEE 802.11, Random access, Controlled access, Channelization.

UNIT – III

Network layer: Logical Addressing, Internetworking, Tunneling, Address mapping, ICMP, IGMP, Forwarding, Uni-Cast Routing Protocols, Multicast Routing Protocols.

UNIT – IV

Transport Layer: Process to Process Delivery, UDP and TCP protocols, Data Traffic, Congestion, Congestion Control, QoS, Integrated Services, Differentiated Services, QoS in Switched Networks.

UNIT – V

Application Layer: Domain name space, DNS in Internet, Electronic Mail, SMTP, FTP, WWW, HTTP, SNMP.

TEXT BOOKS:

1. Data Communications and Networking, Behrouz A. Forouzan, Fourth Edition TMH.

REFERENCE BOOKS:

1. Computer Networks, Andrew S Tanenbaum, 6th Edition. Pearson Education.
2. Computer Networking: A Top-Down Approach Featuring the Internet. James F. Kurose & Keith W. Ross, 3rd Edition, Pearson Education
3. Data communications and Computer Networks, P.C Gupta, PHI.
4. An Engineering Approach to Computer Networks, S. Keshav, 2nd Edition, Pearson Education.

22DS501PC: MACHINE LEARNING**(Common To CSE(DS), CSE(AIML) & IT)****B.Tech. III Year I Sem.****L T P C****3 0 0 3****Course Objectives:**

- To introduce students to the basic concepts and techniques of Machine Learning.
- To have a thorough understanding of the Supervised and Unsupervised learning techniques
- To study the various probability-based learning techniques

Course Outcomes:

- Distinguish between, supervised, unsupervised and semi-supervised learning
- Understand algorithms for building classifiers applied on datasets of non-linearly separable classes
- Understand the principles of evolutionary computing algorithms
- Design an ensembler to increase the classification accuracy

UNIT - I

Learning - Types of Machine Learning - Supervised Learning - The Brain and the Neuron - Design a Learning System - Perspectives and Issues in Machine Learning - Concept Learning Task - Concept Learning as Search - Finding a Maximally Specific Hypothesis - Version Spaces and the Candidate Elimination Algorithm - Linear Discriminants: - Perceptron - Linear Separability - Linear Regression.

UNIT - II

Multi-layer Perceptron- Going Forwards - Going Backwards: Back Propagation Error - Multi-layer Perceptron in Practice - Examples of using the MLP - Overview - Deriving Back-Propagation - Radial Basis Functions and Splines - Concepts - RBF Network - Curse of Dimensionality - Interpolations and Basis Functions - Support Vector Machines

UNIT - III

Learning with Trees - Decision Trees - Constructing Decision Trees - Classification and Regression Trees - Ensemble Learning - Boosting - Bagging - Different ways to Combine Classifiers - Basic Statistics - Gaussian Mixture Models - Nearest Neighbor Methods - Unsupervised Learning - K means Algorithms

UNIT - IV

Dimensionality Reduction - Linear Discriminant Analysis - Principal Component Analysis - Factor Analysis - Independent Component Analysis - Locally Linear Embedding - Isomap - Least Squares Optimization
Evolutionary Learning - Genetic algorithms - Genetic Offspring: - Genetic Operators - Using Genetic Algorithms

UNIT - V

Reinforcement Learning - Overview - Getting Lost Example
Markov Chain Monte Carlo Methods - Sampling - Proposal Distribution - Markov Chain Monte Carlo - Graphical Models - Bayesian Networks - Markov Random Fields - Hidden Markov Models - Tracking Methods

TEXT BOOKS:

1. Stephen Marsland, "Machine Learning – An Algorithmic Perspective, Second Edition, Chapman and Hall/CRC Machine Learning and Pattern Recognition Series, 2014.

REFERENCE BOOKS:

1. Tom M Mitchell, "Machine Learning, First Edition, McGraw Hill Education, 2013.
2. Peter Flach, "Machine Learning: The Art and Science of Algorithms that Make Sense of Data", First Edition, Cambridge University Press, 2012.
3. Jason Bell, "Machine learning - Hands on for Developers and Technical Professionals", First Edition, Wiley, 2014
4. Ethem Alpaydin, "Introduction to Machine Learning 3e (Adaptive Computation and Machine Learning Series), Third Edition, MIT Press, 2014

22CS511PE: QUANTUM COMPUTING (Professional Elective – I)
(Common To CSE & IT)

B.Tech. III Year I Sem.

L T P C
3 0 0 3

Course Objectives

- To introduce the fundamentals of quantum computing
- The problem-solving approach using finite dimensional mathematics

Course Outcomes

- Understand basics of quantum computing
- Understand physical implementation of Qubit
- Understand Quantum algorithms and their implementation
- Understand The Impact of Quantum Computing on Cryptography

UNIT - I

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

UNIT - II

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

UNIT - III

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

UNIT - IV

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

UNIT - V

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

TEXT BOOKS:

1. Nielsen M. A., Quantum Computation and Quantum Information, Cambridge

REFERENCE BOOKS:

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

22CS512PE: ADVANCED COMPUTER ARCHITECTURE**(Professional Elective – I)****(Common To CSE & IT)****B.Tech. III Year I Sem.****L T P C****3 0 0 3****Prerequisites:** Computer Organization**Course Objectives:**

- To impart the concepts and principles of parallel and advanced computer architectures.
- To develop the design techniques of Scalable and multithreaded Architectures.
- To Apply the concepts and techniques of parallel and advanced computer architectures to design modern computer systems

Course Outcomes:

- Computational models and Computer Architectures.
- Concepts of parallel computer models.
- Scalable Architectures, Pipelining, Superscalar processors

UNIT - I

Theory of Parallelism, Parallel computer models, The State of Computing, Multiprocessors and Multicomputers, Multivector and SIMD Computers, PRAM and VLSI models, Architectural development tracks, Program and network properties, Conditions of parallelism, Program partitioning and Scheduling, Program flow Mechanisms, System interconnect Architectures.

UNIT - II

Principles of Scalable performance, Performance metrics and measures, Parallel Processing applications, Speed up performance laws, Scalability Analysis and Approaches, Hardware Technologies, Processes and Memory Hierarchy, Advanced Processor Technology, Superscalar and Vector Processors

UNIT - III

Shared-Memory Organizations, Sequential and weak consistency models, Pipelining and superscalar techniques, Linear Pipeline Processors, Non-Linear Pipeline Processors, Instruction Pipeline design, Arithmetic pipeline design, superscalar pipeline design.

UNIT - IV

Parallel and Scalable Architectures, Multiprocessors and multicomputers, Multiprocessor system interconnects, cache coherence and synchronization mechanism, Three Generations of Multicomputers, Message-passing Mechanisms, Multivector and SIMD computers.

UNIT - V

Vector Processing Principles, Multivector Multiprocessors, Compound Vector processing, SIMD computer Organizations, The connection machine CM-5.

TEXT BOOK:

1. Advanced Computer Architecture, Kai Hwang, 2nd Edition, Tata McGraw Hill Publishers.

REFERENCE BOOKS:

1. Computer Architecture, J.L. Hennessy and D.A. Patterson, 4th Edition, ELSEVIER.
2. Advanced Computer Architectures, S.G.Shiva, Special Indian edition, CRC, Taylor &Francis.
3. Introduction to High Performance Computing for Scientists and Engineers, G. Hager and G. Wellein, CRC Press, Taylor & Francis Group.
4. Advanced Computer Architecture, D. Sima, T. Fountain, P. Kacsuk, Pearson education.
5. Computer Architecture, B. Parhami, Oxford Univ. Press.

22CS513PE: DATA ANALYTICS (Professional Elective – I)
(Common To CSE & IT)

B.Tech. III Year I Sem.

L T P C
3 0 0 3

Prerequisites

1. A course on “Database Management Systems”.
2. Knowledge of probability and statistics.

Course Objectives:

- To explore the fundamental concepts of data analytics.
- To learn the principles and methods of statistical analysis
- Discover interesting patterns, analyze supervised and unsupervised models and estimate the accuracy of the algorithms.
- To understand the various search methods and visualization techniques.

Course Outcomes: After completion of this course students will be able to

- Understand the impact of data analytics for business decisions and strategy
- Carry out data analysis/statistical analysis
- To carry out standard data visualization and formal inference procedures
- Design Data Architecture
- Understand various Data Sources

UNIT - I

Data Management: Design Data Architecture and manage the data for analysis, understand various sources of Data like Sensors/Signals/GPS etc. Data Management, Data Quality(noise, outliers, missing values, duplicate data) and Data Processing & Processing.

UNIT - II

Data Analytics: Introduction to Analytics, Introduction to Tools and Environment, Application of Modeling in Business, Databases & Types of Data and Variables, Data Modeling Techniques, Missing Imputations etc. Need for Business Modeling.

UNIT - III

Regression – Concepts, Blue property assumptions, Least Square Estimation, Variable Rationalization, and Model Building etc.

Logistic Regression: Model Theory, Model fit Statistics, Model Construction, Analytics applications to various Business Domains etc.

UNIT - IV

Object Segmentation: Regression Vs Segmentation - Supervised and Unsupervised Learning, Tree Building - Regression, Classification, Overfitting, Pruning and Complexity, Multiple Decision Trees etc.

Time Series Methods: Arima, Measures of Forecast Accuracy, STL approach, Extract features from generated model as Height, Average Energy etc and Analyze for prediction

UNIT - V

Data Visualization: Pixel-Oriented Visualization Techniques, Geometric Projection Visualization Techniques, Icon-Based Visualization Techniques, Hierarchical Visualization Techniques, Visualizing Complex Data and Relations.

TEXT BOOKS:

1. Student’s Handbook for Associate Analytics - II, III.
2. Data Mining Concepts and Techniques, Han, Kamber, 3rd Edition, Morgan Kaufmann Publishers.

REFERENCE BOOKS:

1. Introduction to Data Mining, Tan, Steinbach and Kumar, Addison Wisley, 2006.
2. Data Mining Analysis and Concepts, M. Zaki and W. Meira
3. Mining of Massive Datasets, Jure Leskovec Stanford Univ. Anand Rajaraman Millway Labs Jeffrey D Ullman Stanford Univ.

22CS514PE: IMAGE PROCESSING (Professional Elective – I)
(Common To CSE, CSE(DS), CSE(AIML) & IT)

B.Tech. III Year I Sem.

L T P C

3 0 0 3

Prerequisites

1. Students are expected to have knowledge in linear signals and systems, Fourier Transform, basic linear algebra, basic probability theory and basic programming techniques; knowledge of digital signal processing is desirable.
2. A course on “Computational Mathematics”
3. A course on “Computer Oriented Statistical Methods”

Course Objectives

- Provide a theoretical and mathematical foundation of fundamental Digital Image Processing concepts.
- The topics include image acquisition; sampling and quantization; preprocessing; enhancement; restoration; segmentation; and compression.

Course Outcomes

- Demonstrate the knowledge of the basic concepts of two-dimensional signal acquisition, sampling, and quantization.
- Demonstrate the knowledge of filtering techniques.
- Demonstrate the knowledge of 2D transformation techniques.
- Demonstrate the knowledge of image enhancement, segmentation, restoration and compression techniques.

UNIT - I

Digital Image Fundamentals: Digital Image through Scanner, Digital Camera. Concept of Gray Levels. Gray Level to Binary Image Conversion. Sampling and Quantization. Relationship between Pixels. Imaging Geometry. 2D Transformations-DFT, DCT, KLT and SVD.

UNIT - II

Image Enhancement in Spatial Domain Point Processing, Histogram Processing, Spatial Filtering, Enhancement in Frequency Domain, Image Smoothing, Image Sharpening.

UNIT - III

Image Restoration Degradation Model, Algebraic Approach to Restoration, Inverse Filtering, Least Mean Square Filters, Constrained Least Squares Restoration, Interactive Restoration.

UNIT - IV

Image Segmentation Detection of Discontinuities, Edge Linking and Boundary Detection, Thresholding, Region Oriented Segmentation.

UNIT - V

Image Compression Redundancies and their Removal Methods, Fidelity Criteria, Image Compression Models, Source Encoder and Decoder, Error Free Compression, Lossy Compression.

TEXT BOOK:

1. Digital Image Processing: R.C. Gonzalez & R. E. Woods, Addison Wesley/ Pearson Education, 2nd Ed, 2004.

REFERENCE BOOKS:

1. Fundamentals of Digital Image Processing: A. K. Jain, PHI.
2. Digital Image Processing using MATLAB: Rafael C. Gonzalez, Richard E. Woods, Steven L. Eddins: Pearson Education India, 2004.
3. Digital Image Processing: William K. Pratt, John Wiley, 3rd Edition, 20

22CS515PE: PRINCIPLES OF PROGRAMMING LANGUAGES**(Professional Elective – I)****(Common To CSE & IT)****B.Tech. III Year I Sem.****L T P C****3 0 0 3****Prerequisites**

1. A course on “Mathematical Foundations of Computer Science”.
2. A course on “Computer Programming and Data Structures”.

Course Objectives

- Introduce important paradigms of programming languages
- To provide conceptual understanding of high-level language design and implementation
- Topics include programming paradigms; syntax and semantics; data types, expressions and statements; subprograms and blocks; abstract data types; concurrency; functional and logic programming languages; and scripting languages

Course Outcomes

- Acquire the skills for expressing syntax and semantics in formal notation
- Identify and apply a suitable programming paradigm for a given computing application
- Gain knowledge of the features of various programming languages and their comparison

UNIT - I

Preliminary Concepts: Reasons for Studying Concepts of Programming Languages, Programming Domains, Language Evaluation Criteria, Influences on Language Design, Language Categories, Language Design Trade-Offs, Implementation Methods, Programming Environments Syntax and Semantics: General Problem of Describing Syntax and Semantics, Formal Methods of Describing Syntax, Attribute Grammars, Describing the Meanings of Programs

UNIT - II

Names, Bindings, and Scopes: Introduction, Names, Variables, Concept of Binding, Scope, Scope and Lifetime, Referencing Environments, Named Constants

Data Types: Introduction, Primitive Data Types, Character String Types, User Defined Ordinal Types, Array, Associative Arrays, Record, Union, Tuple Types, List Types, Pointer and Reference Types, Type Checking, Strong Typing, Type Equivalence Expressions and Statements, Arithmetic Expressions, Overloaded Operators, Type Conversions, Relational and Boolean Expressions, Short Circuit Evaluation, Assignment Statements, Mixed-Mode

Assignment Control Structures – Introduction, Selection Statements, Iterative Statements, Unconditional Branching, Guarded Commands.

UNIT - III

Subprograms and Blocks: Fundamentals of Sub-Programs, Design Issues for Subprograms, Local Referencing Environments, Parameter Passing Methods, Parameters that Are Subprograms, Calling Subprograms Indirectly, Overloaded Subprograms, Generic Subprograms, Design Issues for Functions, User Defined Overloaded Operators, Closures, **Coroutines Implementing Subprograms:** General Semantics of Calls and Returns, Implementing Simple Subprograms, Implementing Subprograms with Stack-Dynamic Local Variables, Nested Subprograms, Blocks, Implementing Dynamic Scoping

Abstract Data Types: The Concept of Abstraction, Introductions to Data Abstraction, Design Issues, Language Examples, Parameterized ADT, Encapsulation Constructs, Naming Encapsulations

UNIT - IV

Concurrency: Introduction, Introduction to Subprogram Level Concurrency, Semaphores, Monitors, Message Passing, Java Threads, Concurrency in Function Languages, Statement Level Concurrency.

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

UNIT - V

Functional Programming Languages: Introduction, Mathematical Functions, Fundamentals of Functional Programming Language, LISP, Support for Functional Programming in Primarily Imperative Languages, Comparison of Functional and Imperative Languages

Logic Programming Language: Introduction, an Overview of Logic Programming, Basic Elements of Prolog, Applications of Logic Programming.

Scripting Language: Pragmatics, Key Concepts, Case Study: Python - Values and Types, Variables, Storage and Control, Bindings and Scope, Procedural Abstraction, Data Abstraction, Separate Compilation, Module Library. (Text Book 2)

TEXT BOOKS:

1. Concepts of Programming Languages Robert. W. Sebesta 10/E, Pearson Education.
2. Programming Language Design Concepts, D. A. Watt, Wiley Dreamtech, 2007.

REFERENCE BOOKS:

1. Programming Languages, 2nd Edition, A.B. Tucker, R. E. Noonan, TMH.
2. Programming Languages, K. C. Loudon, 2nd Edition, Thomson, 2003.

22CS521PE: COMPUTER GRAPHICS (Professional Elective – II)
(Common To CSE & IT)

B.Tech. III Year I Sem.

L	T	P	C
3	0	0	3

Prerequisites

1. Programming for problem solving and Data Structures

Course Objectives

- Provide the basics of graphics systems including Points and lines, line drawing algorithms, 2D, 3D objective transformations

Course Outcomes

- Explore applications of computer graphics
- Understand 2D, 3D geometric transformations and clipping algorithms
- Understand 3D object representations, curves, surfaces, polygon rendering methods, color models
- Analyze animation sequence and visible surface detection methods

UNIT - I

Introduction: Application areas of Computer Graphics, overview of graphics systems, video-display devices, raster-scan systems, random-scan systems, graphics monitors and work stations and input devices

Output primitives: Points and lines, line drawing algorithms (DDA and Bresenham's Algorithm) circle-generating algorithms and ellipse - generating algorithms

Polygon Filling: Scan-line algorithm, boundary-fill and flood-fill algorithms

UNIT - II

2-D geometric transformations: Translation, scaling, rotation, reflection and shear transformations, matrix representations and homogeneous coordinates, composite transforms, transformations between coordinate systems

2-D viewing: The viewing pipeline, viewing coordinate reference frame, window to view-port coordinate transformation, viewing functions, clipping operations, point clipping, Line clipping-Cohen Sutherland algorithms, Polygon clipping-Sutherland Hodgeman polygon clipping algorithm.

UNIT - III

3-D object representation: Polygon surfaces, quadric surfaces, spline representation, Hermite curve, Bezier curve and B-Spline curves, Bezier and B-Spline surfaces, Polygon rendering methods, color models, and color applications.

UNIT - IV

3-D Geometric transformations: Translation, rotation, scaling, reflection and shear transformations, composite transformations.

Viewing: Viewing pipeline, viewing coordinates, projections, view volume and general projection transforms and clipping.

UNIT - V

Computer animation: Design of animation sequence, general computer animation functions, raster animations, computer animation languages, key frame systems, motion specifications.

Visible surface detection methods: Classification, back-face detection, depth-buffer method, BSP-tree method, area sub-division method, and octree method.

TEXT BOOKS:

1. "Computer Graphics C version", Donald Hearn and M. Pauline Baker, Pearson Education.

REFERENCE BOOKS:

1. Procedural elements for Computer Graphics, David F Rogers, Tata Mc Graw hill, 2nd edition.
2. Principles of Interactive Computer Graphics”, Neuman and Sproul, TMH.
3. Principles of Computer Graphics, Shalini Govil, Pai, 2005, Springer.
4. “Computer Graphics Principles & practice”, second edition in C, Foley, Van Dam, Feiner and Hughes, Pearson Education.
5. Computer Graphics, Steven Harrington, TMH.

22AM522PE: PATTERN RECOGNITION (Professional Elective – II)
(Common To CSE(AIML) & IT)

B.Tech. III Year I Sem.

L T P C
3 0 0 3

Prerequisites:

- Programming for problem solving
- Computer Oriented Statistical Methods

Course Objectives:

- Introducing fundamental concepts, theories, and algorithms for pattern recognition and machine learning.

Course Outcomes:

- Understand the importance of pattern recognition and its representation
- Analyze the variants of NN algorithm
- Understand the necessity of Hidden markov models, decision tree and SVM for classification
- Understand different types of clustering algorithms

UNIT - I

Introduction: Pattern Recognition, Data Sets for Pattern Recognition, Different Paradigms for Pattern Recognition. Representation: Data Structures for Pattern Representation, Representation of Clusters, Proximity Measures, Size of Patterns, Abstractions of the Data Set, Feature Extraction, Feature Selection, Evaluation of Classifier, Evaluation of Clustering.

UNIT - II

Nearest Neighbor Based Classifier: Nearest Neighbor Algorithm, Variants of the NN Algorithm, use of the Nearest Neighbor Algorithm for Transaction Databases, Efficient Algorithms, Data Reduction, Prototype Selection. Bayes Classifier: Bayes Theorem, Minimum Error Rate Classifier, Estimation of Probabilities, Comparison with the NNC, Naïve Bayes Classifier, Bayesian Belief Network.

UNIT - III

Hidden Markov Models: Markov Models for Classification, Hidden Markov Models, Classification using HMMs. Decision Trees: Introduction, Decision Tree for Pattern Classification, Construction of Decision Trees, Splitting at the Nodes, Overfitting and Pruning, Examples of Decision Tree Induction.

UNIT - IV

Support Vector Machines: Introduction, Learning the Linear Discriminant Functions, Neural Networks, SVM for Classification. Combination of Classifiers: Introduction, Methods for Constructing Ensembles of Classifiers, Methods for Combining Classifiers.

UNIT - V

Clustering: Importance of clustering, Hierarchical Algorithms, Partitional Clustering, Clustering Large Data Sets. An Application-Hand Written Digit Recognition: Description of the Digit Data, Preprocessing of Data, Classification Algorithms, Selection of Representative Patterns, Results.

TEXT BOOK:

1. Pattern Recognition: An Algorithmic Approach: Murty, M. Narasimha, Devi, V. Susheela, Springer Pub, 1st Ed.

REFERENCE BOOKS:

1. Machine Learning - Mc Graw Hill, Tom M. Mitchell.
2. Fundamentals of Speech Recognition: Lawrence Rabiner and Bing-Hwang Juang. Prentice Hall Pub.

22IT523PE: ADVANCED OPERATING SYSTEMS (Professional Elective – II)**B.Tech. III Year I Sem.****L T P C**
3 0 0 3**Course Objectives**

- To study, learn, and understand the main concepts of advanced operating systems (parallel processing systems, distributed systems, real time systems, network operating systems, and open-source operating systems), Hardware and software features that support these systems.

Course Outcomes

- Understand the design approaches of advanced operating systems
- Analyze the design issues of distributed operating systems.
- Evaluate design issues of multi-processor operating systems.
- Identify the requirements Distributed File System and Distributed Shared Memory.
- Formulate the solutions to schedule the real time applications.

UNIT - I

Architectures of Distributed Systems: System Architecture Types, Distributed Operating Systems, Issues in Distributed Operating Systems, Communication Primitives. Theoretical Foundations: Inherent Limitations of a Distributed System, Lamport's Logical Clocks, Vector Clocks, Causal Ordering of Messages, Termination Detection.

UNIT - II

Distributed Mutual Exclusion: The Classification of Mutual Exclusion Algorithms, Non-Token - Based Algorithms: Lamport's Algorithm, The Ricart-Agrawala Algorithm, Maekawa's Algorithm, Token-Based Algorithms: Suzuki-Kasami's Broadcast Algorithm, Singhal's Heuristic Algorithm, Raymond's Heuristic Algorithm.

UNIT - III

Distributed Deadlock Detection: Preliminaries, Deadlock Handling Strategies in Distributed Systems, Issues in Deadlock Detection and Resolution, Control Organizations for Distributed Deadlock Detection, Centralized- Deadlock - Detection Algorithms, Distributed Deadlock Detection Algorithms, Hierarchical Deadlock Detection Algorithms.

UNIT - IV

Multiprocessor System Architectures: Introduction, Motivation for multiprocessor Systems, Basic Multiprocessor System Architectures Multi Processor Operating Systems: Introduction, Structures of Multiprocessor Operating Systems, Operating Design Issues, Threads, Process Synchronization, Processor Scheduling.

Distributed File Systems: Architecture, Mechanisms for Building Distributed File Systems, Design Issues.

UNIT - V

Distributed Scheduling: Issues in Load Distributing, Components of a Load Distributed Algorithm, Stability, Load Distributing Algorithms, Requirements for Load Distributing, Task Migration, Issues in task Migration Distributed Shared Memory: Architecture and Motivation, Algorithms for Implementing DSM, Memory Coherence, Coherence Protocols, Design Issues.

TEXT BOOK

1. Advanced Concepts in Operating Systems, Mukesh Singhal, Niranjan G. Shivaratri, Tata McGraw-Hill Edition 2001

REFERENCE BOOK

1. Distributed Systems: Andrew S. Tanenbaum, Maarten Van Steen, Pearson Prentice Hall, Edition – 2, 2007.

22CS524PE: DISTRIBUTED DATABASES (Professional Elective – II)
(Common To CSE & IT)

B.Tech. III Year I Sem.

L T P C
3 0 0 3

Prerequisites:

1. A course on "Database Management Systems"

Course Objectives:

- The purpose of the course is to enrich the previous knowledge of database systems and expose the need for distributed database technology to confront the deficiencies of the centralized database systems.
- Introduce basic principles and implementation techniques of distributed database systems.
- Equip students with principles and knowledge of parallel and object-oriented databases.
- Topics include distributed DBMS architecture and design; query processing and optimization; distributed transaction management and reliability; parallel and object database management systems.

Course Outcomes:

- Understand theoretical and practical aspects of distributed database systems.
- Study and identify various issues related to the development of distributed database systems.
- Understand the design aspects of object-oriented database systems and related developments.

UNIT - I

Introduction; Distributed Data Processing, Distributed Database System, Promises of DDBSs, Problem areas.

Distributed DBMS Architecture: Architectural Models for Distributed DBMS, DDMBS Architecture.

Distributed Database Design: Alternative Design Strategies, Distribution Design issues, Fragmentation, Allocation.

UNIT - II

Query processing and decomposition: Query processing objectives, characterization of query processors, layers of query processing, query decomposition, localization of distributed data.

Distributed query Optimization: Query optimization, centralized query optimization, distributed query optimization algorithms.

UNIT - III

Transaction Management: Definition, properties of transaction, types of transactions, distributed concurrency control: serializability, concurrency control mechanisms & algorithms, time - stamped & optimistic concurrency control Algorithms, deadlock Management.

UNIT - IV

Distributed DBMS Reliability: Reliability concepts and measures, fault-tolerance in distributed systems, failures in Distributed DBMS, local & distributed reliability protocols, site failures and network partitioning.

Parallel Database Systems: Parallel database system architectures, parallel data placement, parallel query processing, load balancing, database clusters.

UNIT - V

Distributed object Database Management Systems: Fundamental object concepts and models, object distributed design, architectural issues, object management, distributed object storage, object query Processing.

Object Oriented Data Model: Inheritance, object identity, persistent programming languages, persistence of objects, comparison OODBMS and ORDBMS

TEXT BOOKS:

1. M. Tamer OZSU and Patuck Valduriez: Principles of Distributed Database Systems, Pearson Edn. Asia, 2001.
2. Stefano Ceri and Giuseppe Pelagatti: Distributed Databases, McGraw Hill.

REFERENCE BOOK:

1. Hector Garcia-Molina, Jeffrey D. Ullman, Jennifer Widom: "Database Systems: The Complete Book", Second Edition, Pearson International Edition.

22IT525PE: BIOMETRICS (Professional Elective – II)**B.Tech. III Year I Sem.****L T P C**
3 0 0 3**Prerequisites:**

1. Data structures

Course Objectives:

- Will learn the biometric technologies.
- Learn the computational methods involved in the biometric systems.
- Learn methods for evaluation of the reliability and quality of the biometric systems.

Course Outcomes: After completion of the course, students will be able to:

- Identify the various Biometric technologies.
- Design of face recognition systems for the organization.
- Design finger print, hand geometry applications.
- Develop simple applications for privacy.

UNIT - I

Introduction, history, type of Biometrics, General Architecture of Biometric Systems, Basic Working of biometric Matching, Biometric System Error and performance Measures, Design of Biometric Systems, Applications of Biometrics, Benefits of Biometrics Versus Traditional Authentication Methods

UNIT - II

Face Biometrics & Retina And Iris Biometrics Introduction, Background of Face Recognition, Design of Face Recognition System, Neural Network for Face Recognition, Face Detection in Video Sequences, Challenges in Face Biometrics, Face Recognition Methods, Advantages and Disadvantages, Performance of Biometrics.

UNIT -III

Design of Retina Biometrics, Iris Segmentation Method, Determination of Iris Region, Experimental Results of Iris Localization, Applications of Iris Biometrics, Advantages and Disadvantages.

UNIT - IV

Vein and Fingerprint Biometrics & Biometric Hand Gesture Recognition For Indian Sign Language. Biometrics Using Vein Pattern of Palm, Fingerprint Biometrics, Fingerprint Recognition System, Minutiae Extraction, Fingerprint Indexing, Experimental Results, Advantages and Disadvantages, Basics of Hand Geometry, Sign Language, Indian Sign Language, SIFT Algorithms- Practical Approach Advantages and Disadvantages.

UNIT - V

Privacy Enhancement Using Biometrics & Biometric Cryptography And Multimodal Biometrics: Introduction, Privacy Concerns Associated with Biometric Developments, Identity and Privacy, Privacy Concerns, Biometrics with Privacy Enhancement, Comparison of Various Biometrics in Terms of Privacy, Soft Biometrics - Introduction to Biometric Cryptography.

TEXT BOOKS:

1. G r Sinha and Sandeep B. Patil, Biometrics: concepts and applications, Wiley, 2013.
2. Paul Reid, Biometrics for Network Security, Pearson Education.

REFERENCE BOOKS:

1. Samir Nanavathi, Micheal Thieme and Raj Nanavathi, Biometrics, Identity verification in a networked world, Wiley, dream Tech.
2. John D. Woodward and Jr. Wiley Dreamtech, Biometrics, The Ultimate Reference.

Online websites / Materials:

1. <https://www.biometricsinstitute.org>
2. https://www.tutorialspoint.com/biometrics/biometrics_quick_guide.htm

22IT503PC: SOFTWARE ENGINEERING & COMPUTER NETWORKS LAB**B.Tech. III Year I Sem.****L T P C**
0 0 2 1**Course Objectives:**

- To have hands-on experience in developing a software project by using various software engineering principles and methods in each of the phases of software development.
- To understand the working principle of various communication protocols.

Course Outcomes:

- Ability to translate end-user requirements into system and software requirements
- Ability to generate a high-level design of the system from the software requirements
- Implement data link layer framing methods
- Analyze error detection and error correction codes.
- Implement and analyze routing and congestion issues in network design.

Software Engineering**List of Experiments:**

Do the following seven exercises for any two projects given in the list of sample projects or any other Projects:

1. Development of problem statements.
2. Preparation of Software Requirement Specification Document, Design Documents and Testing Phase related documents.
3. Preparation of Software Configuration Management and Risk Management related documents.
4. Study and usage of any Design phase CASE tool
5. Performing the Design by using any Design phase CASE tools.
6. Develop test cases for unit testing and integration testing
7. Develop test cases for various white box and black box testing techniques.

Sample Projects:

1. Passport automation System
2. Book Bank
3. Online Exam Registration

Computer Networks**List of Experiments:**

1. Implement the data link layer framing methods such as character, character-stuffing and bit stuffing.
2. Write a program to compute CRC code for the polynomials CRC-12, CRC-16 and CRC CCIP
3. Develop a simple data link layer that performs the flow control using the sliding window protocol, and loss recovery using the Go-Back-N mechanism.
4. Implement Dijkstra's algorithm to compute the shortest path through a network
5. Take an example subnet of hosts and obtain a broadcast tree for the subnet.
6. Implement distance vector routing algorithm for obtaining routing tables at each node.
7. Implement data encryption and data decryption

22DS503PC: MACHINE LEARNING LAB**(Common To CSE, CSE(DS), CSE(AIML) & IT)****B.Tech. III Year I Sem.****L T P C**
0 0 2 1**Course Objective:**

- The objective of this lab is to get an overview of the various machine learning techniques and can demonstrate them using python.

Course Outcomes:

- Understand modern notions in predictive data analysis
- Select data, model selection, model complexity and identify the trends
- Understand a range of machine learning algorithms along with their strengths and weaknesses
- Build predictive models from data and analyze their performance

List of Experiments

1. Write a python program to compute Central Tendency Measures: Mean, Median, Mode Measure of Dispersion: Variance, Standard Deviation
2. Study of Python Basic Libraries such as Statistics, Math, Numpy and Scipy
3. Study of Python Libraries for ML application such as Pandas and Matplotlib
4. Write a Python program to implement Simple Linear Regression
5. Implementation of Multiple Linear Regression for House Price Prediction using sklearn
6. Implementation of Decision tree using sklearn and its parameter tuning
7. Implementation of KNN using sklearn
8. Implementation of Logistic Regression using sklearn
9. Implementation of K-Means Clustering
10. Performance analysis of Classification Algorithms on a specific dataset (Mini Project)

TEXT BOOK:

1. Machine Learning - Tom M. Mitchell, - MGH.

REFERENCE BOOK:

1. Machine Learning: An Algorithmic Perspective, Stephen Marshland, Taylor & Francis.

22EN508HS: ADVANCED ENGLISH COMMUNICATION SKILLS LAB

(Common To CSE, CSE(AIML) & IT)

III Year B.Tech. I Sem

L	T	P	C
0	0	2	1

1. Introduction

The introduction of the Advanced English Communication Skills Lab is considered essential at the B.Tech 3rd year level. At this stage, the students need to prepare themselves for their career which may require them to listen to, read, speak and write in English both for their professional and interpersonal communication in the globalised context.

The proposed course should be a laboratory course to enable students to use appropriate English and perform the following:

1. Gathering ideas and information to organise ideas relevantly and coherently.
2. Making oral presentations.
3. Writing formal letters.
4. Transferring information from non-verbal to verbal texts and vice-versa.
5. Writing project/research reports/technical reports.
6. Participating in group discussions.
7. Engaging in debates.
8. Facing interviews.
9. Taking part in social and professional communication.

2. Objectives:

This Lab focuses on using multi-media instruction for language development to meet the following targets:

- To improve the students' fluency in English, with a focus on vocabulary
- To enable them to listen to English spoken at normal conversational speed by educated English speakers
- To respond appropriately in different socio-cultural and professional contexts
- To communicate their ideas relevantly and coherently in writing
- To prepare the students for placements.

3. Syllabus:

The following course content to conduct the activities is prescribed for the Advanced English Communication Skills (AECS) Lab:

1. **Activities on Listening and Reading Comprehension:** Active Listening – Development of Listening Skills Through Audio clips - Benefits of Reading - Methods and Techniques of Reading – Basic Steps to Effective Reading - Common Obstacles - Discourse Markers or Linkers - Sub-skills of reading - Reading for facts, negative facts and Specific Details- Guessing Meanings from Context, Inferring Meaning - Critical Reading – Reading Comprehension - Exercises for Practice.
2. **Activities on Writing Skills:** Vocabulary for Competitive Examinations - Planning for Writing - Improving Writing Skills - Structure and presentation of different types of writing - Free Writing and Structured Writing - Letter Writing -Writing a Letter of Application -Resume vs. Curriculum Vitae - Writing a Résumé – Styles of Résumé - e-Correspondence - Emails - Blog Writing - (N)etiquette - Report Writing - Importance of Reports - Types and Formats of Reports- Technical Report Writing- Exercises for Practice.
3. **Activities on Presentation Skills** - Starting a conversation - responding appropriately and relevantly - using the right language and body language - Role Play in different situations including Seeking Clarification, Making a Request, Asking for and Refusing Permission, Participating in a Small Talk - Oral presentations (individual and group) through JAM sessions- PPTs - Importance of Presentation Skills - Planning, Preparing, Rehearsing and Making a Presentation - Dealing with

- Glossophobia or Stage Fear – Understanding Nuances of Delivery - Presentations through Posters/Projects/Reports - Checklist for Making a Presentation and Rubrics of Evaluation
4. **Activities on Group Discussion (GD):** Types of GD and GD as a part of a Selection Procedure - Dynamics of Group Discussion- Myths of GD - Intervention, Summarizing - Modulation of Voice, Body Language, Relevance, Fluency and Organization of Ideas – Do's and Don'ts - GD Strategies – Exercises for Practice.
 5. **Interview Skills:** Concept and Process - Interview Preparation Techniques - Types of Interview Questions - Pre-interview Planning, Opening Strategies, Answering Strategies - Interview Through Tele-conference & Video-conference - Mock Interviews.

4. Minimum Requirement:

The Advanced English Communication Skills (AECS) Laboratory shall have the following infrastructural facilities to accommodate at least 35 students in the lab:

- Spacious room with appropriate acoustics
- Round Tables with movable chairs
- Audio-visual aids
- LCD Projector
- Public Address system
- One PC with latest configuration for the teacher
- T. V, a digital stereo & Camcorder
- Headphones of High quality

5. Suggested Software: The software consisting of the prescribed topics elaborated above should be procured and used.

- **TOEFL & GRE** (KAPLAN, AARCO & BARRONS, USA, Cracking GRE by CLIFFS)
- **Oxford Advanced Learner's Dictionary**, 10th Edition
- **Cambridge Advanced Learner's Dictionary**
- **DELTA's key to the Next Generation TOEFL Test: Advanced Skill Practice.**
- **Lingua TOEFL CBT Insider**, by Dreamtech

6. Books Recommended:

1. Rizvi, M. Ashraf (2018). *Effective Technical Communication*. (2nd ed.). McGraw Hill Education (India) Pvt. Ltd.
2. Suresh Kumar, E. (2015). *Engineering English*. Orient BlackSwan Pvt. Ltd.
3. Bailey, Stephen. (2018). *Academic Writing: A Handbook for International Students*. (5th Edition). Routledge.
4. Koneru, Aruna. (2016). *Professional Communication*. McGraw Hill Education (India) Pvt. Ltd.
5. Raman, Meenakshi & Sharma, Sangeeta. (2022). *Technical Communication, Principles and Practice*. (4th Edition) Oxford University Press.
6. Anderson, Paul V. (2007). *Technical Communication*. Cengage Learning Pvt. Ltd. New Delhi.
7. McCarthy, Michael; O'Dell, Felicity & Redman, Stuart. (2017). *English Vocabulary in Use Series*. Cambridge University Press
8. Sen, Leela. (2009). *Communication Skills*. PHI Learning Pvt Ltd., New Delhi.
9. Elbow, Peter. (1998). *Writing with Power*. Oxford University Press.
10. Goleman, Daniel. (2013). *Emotional Intelligence: Why it can matter more than IQ*. Bloomsbury Publishing.

22MC509IP: INTELLECTUAL PROPERTY RIGHTS**(Common To CSE, CSE(DS), CSE(AIML) & IT)****B.Tech. III Year I Sem.****L T P C**
3 0 0 0**Course Objectives:**

- Significance of intellectual property and its protection
- Introduce various forms of intellectual property

Course Outcomes:

- Distinguish and Explain various forms of IPRs.
- Identify criteria to fit one's own intellectual work in particular form of IPRs.
- Apply statutory provisions to protect particular form of IPRs.
- Appraise new developments in IPR laws at national and international level

UNIT – I

Introduction to Intellectual property: Introduction, types of intellectual property, international organizations, agencies and treaties, importance of intellectual property rights.

UNIT – II

Trade Marks: Purpose and function of trademarks, acquisition of trade mark rights, protectable matter, selecting, and evaluating trade mark, trade mark registration processes.

UNIT – III

Law of copyrights: Fundamental of copyright law, originality of material, rights of reproduction, rights to perform the work publicly, copyright ownership issues, copyright registration, notice of copyright, International copyright law.

Law of patents: Foundation of patent law, patent searching process, ownership rights and transfer

UNIT – IV

Trade Secrets: Trade secret law, determination of trade secret status, liability for misappropriations of trade secrets, protection for submission, trade secret litigation.

Unfair competition: Misappropriation right of publicity, false advertising.

UNIT – V

New development of intellectual property: new developments in trade mark law; copyright law, patent law, intellectual property audits.

International overview on intellectual property, international – trade mark law, copyright law, international patent law, and international development in trade secrets law.

TEXT BOOK:

1. Intellectual property right, Deborah. E. Bouchoux, Cengage learning.

REFERENCE BOOK:

1. Intellectual property right – Unleashing the knowledge economy, prabuddha ganguli, Tata McGraw Hill Publishing company ltd.

22CS506PC: SKILL DEVELOPMENT COURSE
(Common To CSE, CSE(AIML) & IT)

B.Tech. III Year I Sem.

L T P C
0 0 2 1

Course Objectives:

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

Course Outcomes:

- Implements Flutter Widgets and Layouts
- Responsive UI Design and with Navigation in Flutter
- Create custom widgets for specific UI elements and also Apply styling using themes and custom styles.
- Design a form with various input fields, along with validation and error handling
- Fetches data and write code for unit Test for UI components and also animation

List of Experiments: Students need to implement the following experiments

1. a) Install Flutter and Dart SDK.
b) Write a simple Dart program to understand the language basics.
2. a) Explore various Flutter widgets (Text, Image, Container, etc.).
b) Implement different layout structures using Row, Column, and Stack widgets.
3. a) Design a responsive UI that adapts to different screen sizes.
b) Implement media queries and breakpoints for responsiveness.
4. a) Set up navigation between different screens using Navigator.
b) Implement navigation with named routes.
5. a) Learn about stateful and stateless widgets.
b) Implement state management using set State and Provider.
6. a) Create custom widgets for specific UI elements.
b) Apply styling using themes and custom styles.
7. a) Design a form with various input fields.
b) Implement form validation and error handling.
8. a) Add animations to UI elements using Flutter's animation framework.
b) Experiment with different types of animations (fade, slide, etc.).
9. a) Fetch data from a REST API.
b) Display the fetched data in a meaningful way in the UI.
10. a) Write unit tests for UI components.
b) Use Flutter's debugging tools to identify and fix issues.

TEXT BOOK:

1. Marco L. Napoli, Beginning Flutter: A Hands-on Guide to App Development.

22DS601PC: AUTOMATA THEORY AND COMPILER DESIGN**(Common To CSE(DS) & IT)****B.Tech. III Year II Sem.****L T P C**
3 0 0 3**Course Objectives**

- To introduce the fundamental concepts of formal languages, grammars and automata theory.
- To understand deterministic and non-deterministic machines and the differences between decidability and undecidability.
- Introduce the major concepts of language translation and compiler design and impart the knowledge of practical skills necessary for constructing a compiler.
- Topics include phases of compiler, parsing, syntax directed translation, type checking use of symbol tables, intermediate code generation

Course Outcomes

- Able to employ finite state machines for modeling and solving computing problems.
- Able to design context free grammars for formal languages.
- Able to distinguish between decidability and undecidability.
- Demonstrate the knowledge of patterns, tokens & regular expressions for lexical analysis.
- Acquire skills in using lex tool and design LR parsers

UNIT - I

Introduction to Finite Automata: Structural Representations, Automata and Complexity, the Central Concepts of Automata Theory – Alphabets, Strings, Languages, Problems.

Nondeterministic Finite Automata: Formal Definition, an application, Text Search, Finite Automata with Epsilon-Transitions.

Deterministic Finite Automata: Definition of DFA, How A DFA Process Strings, The language of DFA, Conversion of NFA with ϵ -transitions to NFA without ϵ -transitions. Conversion of NFA to DFA

UNIT - II

Regular Expressions: Finite Automata and Regular Expressions, Applications of Regular Expressions, Algebraic Laws for Regular Expressions, Conversion of Finite Automata to Regular Expressions.

Pumping Lemma for Regular Languages:

Statement of the pumping lemma, Applications of the Pumping Lemma.

Context-Free Grammars: Definition of Context-Free Grammars, Derivations Using a Grammar, Leftmost and Rightmost Derivations, the Language of a Grammar, Parse Trees, Ambiguity in Grammars and Languages.

UNIT - III

Push Down Automata: Definition of the Pushdown Automaton, the Languages of a PDA, Equivalence of PDA's and CFG's, Acceptance by final state

Turing Machines:

Introduction to Turing Machine, Formal Description, Instantaneous description, The language of a Turing machine

Undecidability:

Undecidability, A Language that is Not Recursively Enumerable, An Undecidable Problem That is RE, Undecidable Problems about Turing Machines

UNIT - IV

Introduction: The structure of a compiler

Lexical Analysis: The Role of the Lexical Analyzer, Input Buffering, Recognition of Tokens, The Lexical- Analyzer Generator Lex

Syntax Analysis: Introduction, Context-Free Grammars, Writing a Grammar, Top-Down Parsing, Bottom-Up Parsing, Introduction to LR Parsing: Simple LR, More Powerful LR Parsers

UNIT - V

Syntax-Directed Translation: Syntax-Directed Definitions, Evaluation Orders for SDD's, Syntax-Directed Translation Schemes, Implementing L-Attributed SDD's.

Intermediate-Code Generation: Variants of Syntax Trees, Three-Address Code

Run-Time Environments: Stack Allocation of Space, Access to Nonlocal Data on the Stack, Heap Management

TEXT BOOKS:

1. Introduction to Automata Theory, Languages, and Computation, 3rd Edition, John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman, Pearson Education.
2. Theory of Computer Science- Automata languages and computation, Mishra and Chandrashekar, 2nd Edition, PHI.

REFERENCE BOOKS:

1. Compilers: Principles, Techniques and Tools, Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman, 2nd Edition, Pearson.
2. Introduction to Formal languages Automata Theory and Computation, Kamala Krithivasan, Rama R, Pearson.
3. Introduction to Languages and The Theory of Computation, John C Martin, TMH.
4. lex & yacc - John R. Levine, Tony Mason, Doug Brown, O'reilly
5. Compiler Construction, Kenneth C. Loudon, Thomson. Course Technology.

22DS602PC: ALGORITHM DESIGN AND ANALYSIS**(Common To CSE(DS) & IT)****B.Tech. III Year II Sem.**

L	T	P	C
3	0	0	3

Prerequisites: Programming for problem solving and Data Structures**Course Objectives:**

- Introduces the notations for analysis of the performance of algorithms.
- Describes major algorithmic techniques (divide-and-conquer, backtracking, dynamic programming, greedy, branch and bound methods) and mention problems for which each technique is appropriate;
- Describes how to evaluate and compare different algorithms using worst, average, and best-case analysis.
- Explains the difference between tractable and intractable problems, and introduces the problems that are P, NP and NP complete.

Course Outcomes:

- Analyze the performance of algorithms
- Choose appropriate data structures and algorithm design methods for a specified application
- Understand the choice of data structures and the algorithm design methods

UNIT - I**Introduction:** Algorithm, Performance Analysis-Space complexity, Time complexity, Asymptotic Notations- Big oh notation, Omega notation, Theta notation and Little oh notation.**Divide and conquer:** General method, applications-Binary search, Quick sort, Merge sort, Strassen's matrix multiplication.**UNIT - II****Disjoint Sets:** Disjoint set operations, union and find algorithms, Priority Queue- Heaps, Heapsort**Backtracking:** General method, applications, n-queen's problem, sum of subsets problem, graph Coloring, Hamiltonian cycles.**UNIT - III****Dynamic Programming:** General method, applications- Optimal binary search tree, 0/1 knapsack problem, All pairs shortest path problem, Traveling sales person problem, Reliability design.**UNIT - IV****Greedy method:** General method, applications- Job sequencing with deadlines, knapsack problem, Minimum cost spanning trees, Single source shortest path problem.**Basic Traversal and Search Techniques:** Techniques for Binary Trees, Techniques for Graphs, Connected components, Biconnected components.**UNIT - V****Branch and Bound:** General method, applications - Travelling sales person problem, 0/1 knapsack problem - LC Branch and Bound solution, FIFO Branch and Bound solution.**NP-Hard and NP-Complete problems:** Basic concepts, non-deterministic algorithms, NP - Hard and NP-Complete classes, Cook's theorem.**TEXT BOOK:**

1. Fundamentals of Computer Algorithms, Ellis Horowitz, Satraj Sahni and Rajasekharan, University Press.

REFERENCE BOOKS:

1. Design and Analysis of algorithms, Aho, Ullman and Hopcroft, Pearson education.
2. Introduction to Algorithms, second edition, T. H. Cormen, C.E. Leiserson, R. L. Rivest, and C. Stein, PHI Pvt. Ltd./ Pearson Education.
3. Algorithm Design: Foundations, Analysis and Internet Examples, M.T. Goodrich and R.Tamassia, John Wiley and sons.

22EC610PC: EMBEDDED SYSTEMS**B.Tech. III Year II Sem.****L T P C**
3 0 0 3**Pre-requisites:**

1. A course on "Digital Logic Design and Microprocessors".
2. A course on "Computer Organization and Architecture".

Course Objectives:

- To provide an overview of principles of Embedded System
- To provide a clear understanding of role of firmware, operating systems in correlation with hardware systems.

Course Outcomes:

- Expected to understand the selection procedure of processors in the embedded domain.
- Design procedure of embedded firm ware.
- Expected to visualize the role of realtime operating systems in embedded systems.
- Expected to evaluate the correlation between task synchronization and latency issues

UNIT - I

Introduction to Embedded Systems: Processor embedded into a system, Embedded Hardware units and devices in a system, Embedded software in a system, Design process of an embedded system, classification of embedded systems, characteristics and quality attributes of an embedded systems

UNIT - II

Introduction to processor/microcontroller architecture, Real world interfacing, processor and memory organization, memory types, memory maps and addresses, interrupt sources and interrupt service mechanism.

UNIT - III

On board Communication Basics: serial; communication devices, Parallel devices, Wireless devices, Real time clock, Serial bus communication Protocols-I2C, SPI; Parallel buss communication-ISA, PCI.

UNIT - IV

Embedded Firmware Development: Overview of programming concepts - in assembly language and in high level language 'C', C Program elements- Heads, Source files, Processor Directives, Macros, Functions, Data types and Data Structures

UNIT - V

OS Based Embedded Systems: OS services - Process/Task Management, Memory Management, I/O subsystem manager, Inter Process/Task communications - Tasks, Task states, Shared data, Signals, Message Queues, Mailbox, Pipes and concepts of Semaphores.

TEXT BOOK:

1. Embedded Systems, Raj Kamal, 2nd edition, Tata Mc Graw Hill
2. Shibu K V, "Introduction to Embedded Systems", Second Edition, Mc Graw Hill

REFERENCE BOOKS:

1. Rajkamal, Embedded Systems Architecture, Programming and Design, Tata McGraw-Hill
2. Frank Vahid and Tony Givargis, "Embedded Systems Design" - A Unified Hardware/Software Introduction, John Wiley
3. Lyla, "Embedded Systems" -Pearson
4. David E. Simon, An Embedded Software Primer, Pearson Education Asia, First Indian Reprint 2000.

22CS621PE: FULL STACK DEVELOPMENT (Professional Elective – III)
(Common To CSE & IT)

B.Tech. III Year II Sem.

L T P C
3 0 0 3

Pre-Requisites:

1. Object Oriented Programming
2. Web Technologies

Course Objectives:

- Students will become familiar to implement fast, efficient, interactive and scalable web applications using run time environment provided by the full stack components.

Course Outcomes:

1. Understand Full stack components for developing web application.
2. Apply packages of NodeJS to work with Data, Files, Http Requests and Responses.
3. Use MongoDB data base for storing and processing huge data and connects with NodeJS application.
4. Design faster and effective single page applications using Express and Angular.
5. Create interactive user interfaces with react components.

UNIT-I**Introduction to Full Stack Development:**

Understanding the Basic Web Development Framework- User, Browser, Webserver, Backend Services, Full Stack Components - Node.js, MongoDB, Express, React, Angular. Java Script Fundamentals, NodeJS- Understanding Node.js, Installing Node.js, Working with Node Packages, creating a Node.js Application, Understanding the Node.js Event Model, Adding Work to the Event Queue, Implementing Callbacks

UNIT-II**Node.js:**

Working with JSON, Using the Buffer Module to Buffer Data, Using the Stream Module to Stream Data, Accessing the File System from Node.js- Opening, Closing, Writing, Reading Files and other File System Tasks. Implementing HTTP Services in Node.js- Processing URLs, Processing Query Strings and Form Parameters, Understanding Request, Response, and Server Objects, Implementing HTTP Clients and Servers in Node.js, Implementing HTTPS Servers and Clients. Using Additional Node.js Modules-Using the os Module, Using the util Module, Using the dns Module, Using the crypto Module.

UNIT-III**MongoDB:**

Need of NoSQL, Understanding MongoDB, MongoDB Data Types, Planning Your Data Model, Building the MongoDB Environment, Administering User Accounts, Configuring Access Control, Administering Databases, Managing Collections, Adding the MongoDB Driver to Node.js, Connecting to MongoDB from Node.js, Understanding the Objects Used in the MongoDB Node.js Driver, Accessing and Manipulating Databases, Accessing and Manipulating Collections

UNIT-IV**Express and Angular:**

Getting Started with Express, Configuring Routes, Using Requests Objects, Using Response Objects. Angular: importance of Angular, Understanding Angular, creating a Basic Angular Application, Angular Components, Expressions, Data Binding, Built-in Directives, Custom Directives, Implementing Angular Services in Web Applications.

UNIT-V**React:**

Need of React, Simple React Structure, The Virtual DOM, React Components, Introducing React Components, Creating Components in React, Data and Data Flow in React, Rendering and Life Cycle Methods in React, Working with forms in React, integrating third party libraries, Routing in React.

TEXT BOOKS:

1. Brad Dayley, Brendan Dayley, Caleb Dayley., Node.js, MongoDB and Angular Web Development, 2nd Edition, Addison-Wesley, 2019.
2. Mark Tielens Thomas, React in Action, 1st Edition, Manning Publications.

REFERENCE BOOKS:

1. Vasan Subramanian, Pro MERN Stack, Full Stack Web App Development with Mongo, Express, React, and Node, 2nd Edition, Apress, 2019.
2. Chris Northwood, 'The Full Stack Developer: Your Essential Guide to the Everyday Skills Expected of a Modern Full Stack Web Developer', 1st edition, Apress, 2018.
3. Kirupa Chinnathambi, Learning React: A Hands-On Guide to Building Web Applications Using React and Redux, 2nd edition, Addison-Wesley Professional, 2018.

22IT632PE: DATA MINING (Professional Elective – III)**B.Tech. III Year II Sem.****L T P C**
3 0 0 3**Pre-Requisites:**

1. Database Management System
2. Probability and Statistics

Course Objectives:

- Students will become acquainted with both the strengths and limitations of various data mining techniques like Association, Classification, Cluster and Outlier analysis.

Course Outcomes:

- Understand the need of data mining and pre-processing techniques.
- Perform market basket analysis using association rule mining.
- Utilize classification techniques for analysis and interpretation of data.
- Identify appropriate clustering and outlier detection techniques to handle complex data.
- Understand the mining of data from web, text and time series data.

UNIT - I**Introduction to Data Mining:**

What Data mining? Kinds of Data, Knowledge Discovery process, Data Mining Functionalities, Kinds of Patterns, Major Issues in Data Mining. Data Objects and Attribute Types, Basic Statistical Descriptions of Data, Data Visualization, Measuring Data Similarity and Dissimilarity, Data Pre-processing: Major Tasks in Data Pre-processing, Data Cleaning, Data Integration, Data Reduction, Data Transformation and Data Discretization.

UNIT - II

Association Analysis: Basic Concepts, Market Basket Analysis, Apriori Algorithm, FP-growth, From Association Analysis to Correlation Analysis, Pattern Mining in Multilevel Associations and Multidimensional Associations.

UNIT - III

Classification: Basic Concepts, Decision Tree Induction, Bayes Classification Methods, Rule-Based Classification, Metrics for Evaluating Classifier Performance, Ensemble Methods, Multilayer Feed-Forward Neural Network, Support Vector Machines, k-Nearest-Neighbor Classifiers.

UNIT - IV

Cluster Analysis: Requirements for Cluster Analysis, Overview of Basic Clustering Methods, Partitioning Methods-k-Means, k-Medoids, Hierarchical Methods-AGENES, DIANA, BIRCH, Density-Based Method-DBSCAN, Outlier Analysis: Types of Outliers, Challenges of Outlier Detection, and Overview of Outlier Detection Methods

UNIT - V

Advanced Concepts: Web Mining- Web Content Mining, Web Structure Mining, Web Usage Mining, Spatial Mining- Spatial Data Overview, Spatial Data Mining Primitives, Spatial Rules, Spatial Classification Algorithm, Spatial Clustering Algorithms, Temporal Mining- Modeling Temporal Events, Time Series, Pattern Detection, Sequences, Temporal Association Rules.

TEXT BOOKS:

1. Jiawei Han, Micheline Kamber, Jian Pei., Data Mining: Concepts and Techniques, 3rd Edition, Morgan Kaufmann/Elsevier, 2012.
2. Margaret H Dunham, Data Mining Introductory and Advanced Topics, 2nd Edition, Pearson Education, India, 2006.

REFERENCE BOOKS:

1. Data Mining Techniques, Arun K Pujari, 3rd Edition, Universities Press.
2. Pang-Ning Tan, Michael Steinbach, Anuj Karpatne and Vipin Kumar, Introduction to Data Mining, 2nd Edition, Pearson Education India, 2021.
3. Amitesh Sinha, Data Warehousing, Thomson Learning, India, 2007.

22CS633PE: SCRIPTING LANGUAGES (Professional Elective – III)
(Common To CSE, CSE(DS) & IT)

B.Tech. III Year II Sem.

L T P C
3 0 0 3

Prerequisites:

1. A course on “Computer Programming and Data Structures”.
2. A course on “Object Oriented Programming Concepts”.

Course Objectives:

- This course introduces the script programming paradigm
- Introduces scripting languages such as Perl, Ruby and TCL.
- Learning TCL

Course Outcomes:

- Comprehend the differences between typical scripting languages and typical system and application programming languages.
- Gain knowledge of the strengths and weakness of Perl, TCL and Ruby; and select an appropriate language for solving a given problem.
- Acquire programming skills in scripting language

UNIT - I

Introduction: Ruby, Rails, The structure and Execution of Ruby Programs, Package Management with RUBYGEMS, Ruby and web: Writing CGI scripts, cookies, Choice of Webservers, SOAP and web services

RubyTk - Simple Tk Application, widgets, Binding events, Canvas, scrolling

UNIT - II

Extending Ruby: Ruby Objects in C, the Jukebox extension, Memory allocation, Ruby Type System, Embedding Ruby to Other Languages, Embedding a Ruby Interpreter

UNIT - III

Introduction to PERL and Scripting

Scripts and Programs, Origin of Scripting, Scripting Today, Characteristics of Scripting Languages, Uses for Scripting Languages, Web Scripting, and the universe of Scripting Languages. PERL- Names and Values, Variables, Scalar Expressions, Control Structures, arrays, list, hashes, strings, pattern and regular expressions, subroutines.

UNIT - IV

Advanced perl

Finer points of looping, pack and unpack, filesystem, eval, data structures, packages, modules, objects, interfacing to the operating system, Creating Internet ware applications, Dirty Hands Internet Programming, security Issues.

UNIT - V

TCL

TCL Structure, syntax, Variables and Data in TCL, Control Flow, Data Structures, input/output, procedures, strings, patterns, files, Advance TCL- eval, source, exec and uplevel commands, Name spaces, trapping errors, event driven programs, making applications internet aware, Nuts and Bolts Internet Programming, Security Issues, C Interface.

Tk

Tk-Visual Tool Kits, Fundamental Concepts of Tk, Tk by example, Events and Binding, Perl-Tk.

TEXT BOOKS:

1. The World of Scripting Languages, David Barron, Wiley Publications.
2. Ruby Programming language by David Flanagan and Yukihiro Matsumoto O'Reilly
3. "Programming Ruby" The Pragmatic Programmers guide by Dabve Thomas Second edition

REFERENCE BOOKS:

1. Open Source Web Development with LAMP using Linux Apache, MySQL, Perl and PHP, J.Lee and B. Ware (Addison Wesley) Pearson Education.
2. Perl by Example, E. Quigley, Pearson Education.
3. Programming Perl, Larry Wall, T. Christiansen and J. Orwant, O'Reilly, SPD.
4. Tcl and the Tk Tool kit, Ousterhout, Pearson Education.
5. Perl Power, J. P. Flynt, Cengage Learning.

22CS634PE: MOBILE APPLICATION DEVELOPMENT (Professional Elective – III)
(Common To CSE, CSE(DS) & IT)

B.Tech. III Year II Sem.

L T P C
3 0 0 3

Prerequisites

1. Acquaintance with JAVA programming
2. A Course on DBMS

Course Objectives

- To demonstrate their understanding of the fundamentals of Android operating systems
- To improve their skills of using Android software development tools
- To demonstrate their ability to develop software with reasonable complexity on mobile platform
- To demonstrate their ability to deploy software to mobile devices
- To demonstrate their ability to debug programs running on mobile devices

Course Outcomes

- Understand the working of Android OS Practically.
- Develop Android user interfaces
- Develop, deploy and maintain the Android Applications.

UNIT - I

Introduction to Android Operating System: Android OS design and Features - Android development framework, SDK features, Installing and running applications on Android Studio, Creating AVDs, Types of Android applications, Best practices in Android programming, Android tools Android application components - Android Manifest file, Externalizing resources like values, themes, layouts, Menus etc, Resources for different devices and languages, Runtime Configuration Changes
 Android Application Lifecycle - Activities, Activity lifecycle, activity states, monitoring state changes

UNIT - II

Android User Interface: Measurements - Device and pixel density independent measuring unit - s
 Layouts - Linear, Relative, Grid and Table Layouts
 User Interface (UI) Components -Editable and non-editable TextViews, Buttons, Radio and Toggle Buttons, Checkboxes, Spinners, Dialog and pickers
 Event Handling - Handling clicks or changes of various UI components
 Fragments - Creating fragments, Lifecycle of fragments, Fragment states, Adding fragments to Activity, adding, removing and replacing fragments with fragment transactions, interfacing between fragments and Activities, Multi-screen Activities

UNIT - III

Intents and Broadcasts: Intent - Using intents to launch Activities, Explicitly starting new Activity, Implicit Intents, Passing data to Intents, Getting results from Activities, Native Actions, using Intent to dial a number or to send SMS
 Broadcast Receivers - Using Intent filters to service implicit Intents, Resolving Intent filters, finding and using Intents received within an Activity
 Notifications - Creating and Displaying notifications, Displaying Toasts

UNIT - IV

Persistent Storage: Files - Using application specific folders and files, creating files, reading data from files, listing contents of a directory Shared Preferences - Creating shared preferences, saving and retrieving data using Shared Preference

UNIT - V

Database - Introduction to SQLite database, creating and opening a database, creating tables, inserting retrieving and etindelg data, Registering Content Providers, Using content Providers (insert, delete, retrieve and update)

TEXT BOOK:

1. Professional Android 4 Application Development, Reto Meier, Wiley India, (Wrox), 2012

REFERENCE BOOKS:

1. Android Application Development for Java Programmers, James C Sheusi, Cengage Learning, 2013
2. Beginning Android 4 Application Development, Wei-Meng Lee, Wiley India (Wrox), 2013

22CS635PE: SOFTWARE TESTING METHODOLOGIES**(Professional Elective – III)****(Common To CSE, CSE(DS) & IT)****B.Tech. III Year II Sem.****L T P C****3 0 0 3****Prerequisites**

1. Software Engineering

Course Objectives

- To provide knowledge of the concepts in software testing such as testing process, criteria, strategies, and methodologies.
- To develop skills in software test automation and management using the latest tools.

Course Outcomes:

- Understand purpose of testing and path testing
- Understand strategies in data flow testing and domain testing
- Develop logic-based test strategies
- Understand graph matrices and its applications
- Implement test cases using any testing automation tool

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

Transaction Flow Testing: transaction flows, transaction flow testing techniques.

Data Flow testing: Basics of data flow testing, strategies in data flow testing, application of data flow testing.

Domain Testing: domains and paths, Nice & ugly domains, domain testing, domains and interfaces testing, domain and interface testing, domains and testability.

UNIT - III

Paths, Path products and Regular expressions: path products & path expression, reduction procedure, applications, regular expressions & flow anomaly detection.

Logic Based Testing: overview, decision tables, path expressions, kv charts, specifications.

UNIT - IV

State, State Graphs and Transition testing: state graphs, good & bad state graphs, state testing, 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).

TEXT BOOKS:

1. Software Testing techniques - Baris Beizer, Dreamtech, second edition.
2. Software Testing Tools - Dr. K. V. K. K. Prasad, Dreamtech.

REFERENCE BOOKS:

1. The craft of software testing - Brian Marick, Pearson Education.
2. Software Testing Techniques - SPD(Oreille)
3. Software Testing in the Real World - Edward Kit, Pearson.
4. Effective methods of Software Testing, Perry, John Wiley.
5. Art of Software Testing - Meyers, John Wiley.

22IT611OE: JAVA PROGRAMMING (Open Elective – I)**B.Tech. III Year II Sem.****L T P C**
3 0 0 3**Course Objectives:**

- To introduce object-oriented programming principles and apply them in solving problems.
- To introduce the implementation of packages and interfaces.
- To introduce the concepts of exception handling and multithreading.
- To introduce the design of Graphical User Interface using swing controls.

Course Outcomes:

- Able to solve real world problems using OOP techniques.
- Able to solve problems using java collection framework and I/O classes.
- Able to develop multithreaded applications with synchronization.
- Able to design GUI based applications.

UNIT - I

Foundations of Java: History of Java, Java Features, Variables, Data Types, Operators, Expressions, Control Statements. Elements of Java - Class, Object, Methods, Constructors and Access Modifiers, Generics, Inner classes, String class and Annotations.

OOP Principles: Encapsulation - concept, setter and getter method usage, this keyword. Inheritance - concept, Inheritance Types, super keyword. Polymorphism - concept, Method Overriding usage and Type Casting. Abstraction - concept, abstract keyword and Interface.

UNIT – II

Exception Handling: Exception and Error, Exception Types, Exception Handler, Exception Handling Clauses – try, catch, finally, throws and the throw statement, Built-in-Exceptions and Custom Exceptions.

Files and I/O Streams: The file class, Streams, The Byte Streams, Filtered Byte Streams, The Random Access File class.

UNIT – III

Packages- Defining a Package, CLASSPATH, Access Specifiers, importing packages. Few Utility Classes - String Tokenizer, BitSet, Date, Calendar, Random, Formatter, Scanner.

Collections: Collections overview, Collection Interfaces, Collections Implementation Classes, Sorting in Collections, Comparable and Comparator Interfaces.

UNIT – IV

Multithreading: Process and Thread, Differences between thread-based multitasking and process-based multitasking, Java thread life cycle, creating threads, thread priorities, synchronizing threads, inter thread communication.

Java Database Connectivity: Types of Drivers, JDBC architecture, JDBC Classes and Interfaces, Basic steps in Developing JDBC Application, Creating a New Database and Table with JDBC.

UNIT - V

GUI Programming with Swing – Introduction, limitations of AWT, MVC architecture, components, containers, Layout Manager Classes, Simple Applications using AWT and Swing.

Event Handling- The Delegation event model- Events, Event sources, Event Listeners, Event classes, Handling mouse and keyboard events, Adapter classes.

TEXT BOOKS:

1. Java The complete reference, 9th edition, Herbert Schildt, McGraw Hill Education (India) Pvt. Ltd.
2. Understanding Object-Oriented Programming with Java, updated edition, T. Budd, Pearson Education.

REFERENCE BOOKS:

1. An Introduction to programming and OO design using Java, J. Nino and F.A. Hosch, John Wiley & sons
2. Introduction to Java programming, Y. Daniel Liang, Pearson Education.
3. Object Oriented Programming through Java, P. Radha Krishna, University Press.
4. Programming in Java, S. Malhotra, S. Chudhary, 2nd edition, Oxford Univ. Press.
5. Java Programming and Object-oriented Application Development, R. A. Johnson, Cengage Learning.

22IT612OE: OBJECT ORIENTED PROGRAMMING USING C++ (Open Elective – I)**B.Tech. III Year II Sem.****L T P C**
3 0 0 3**Course Objectives:**

- Introduces Object Oriented Programming concepts using the C++ language.
- Understand the principles of data abstraction, inheritance and polymorphism;
- Implementation of the principles of virtual functions and polymorphism
- Handling formatted I/O, unformatted I/O in C++ and implementation of exception handling

Course Outcomes:

- Develop programs with reusability and understand OO functions
- Develop programs for file handling, data abstraction, data hiding
- Develop inheritance, overloading and exceptions in programming
- Implement I/O operations and file handling
- Develop applications for a range of problems using object-oriented programming techniques

UNIT - I

Object-Oriented Thinking: Different paradigms for problem solving, need for OOP paradigm, differences between OOP and Procedure oriented programming, Overview of OOP concepts Abstraction, Encapsulation, Inheritance and Polymorphism.

C++ Basics: Structure of a C++ program, Data types, Declaration of variables, Expressions, Operators, Operator Precedence, Evaluation of expressions, Type conversions, Pointers, Arrays, Pointers and Arrays, Strings, Structures, References. Flow control statement- if, switch, while, for, do, break, continue, goto statements. Functions - Scope of variables, Parameter passing, Default arguments, inline functions, Recursive functions, Pointers to functions. Dynamic memory allocation and deallocation operators-new and delete, Preprocessor directives.

UNIT - II

C++ Classes and Data Abstraction: Class definition, Class structure, Class objects, Class scope, this pointer, Friends to a class, Static class members, Constant member functions, Constructors and Destructors, Dynamic creation and destruction of objects, Data abstraction, ADT and information hiding.

UNIT - III

Inheritance: Defining a class hierarchy, Different forms of inheritance, Defining the Base and Derived classes, Access to the base class members, Base and Derived class construction, Destructors, Virtual base class.

Virtual Functions and Polymorphism: Static and Dynamic binding, virtual functions, Dynamic binding through virtual functions, Virtual function call mechanism, Pure virtual functions, Abstract classes, Implications of polymorphic use of classes, Virtual destructors.

UNIT - IV

C++ I/O: I/O using C functions, Stream classes hierarchy, Stream I/O, File streams and String streams, Overloading operators, Error handling during file operations, Formatted I/O.

UNIT - V

Exception Handling: Benefits of exception handling, Throwing an exception, The try block, Catching an exception, Exception objects, Exception specifications, Stack unwinding, Rethrowing an exception, Catching all exceptions.

TEXT BOOKS:

1. The Complete Reference C++, 4th Edition, Herbert Schildt, Tata McGraw Hill.

2. Problem solving with C++: The Object of Programming, 4th Edition, Walter Savitch, Pearson Education.

REFERENCE BOOKS:

1. The C++ Programming Language, 3rd Edition, B. Stroustrup, Pearson Education.
2. OOP in C++, 3rd Edition, T. Gaddis, J. Walters and G. Muganda, Wiley DreamTech Press.
3. Object Oriented Programming in C++, 3rd Edition, R. Lafore, Galgotia Publications Pvt Ltd.

22IT603PC: COMPILER DESIGN LAB**B.Tech. III Year II Sem.**

L	T	P	C
0	0	2	1

Prerequisites

1. A Course on "Object Oriented Programming through Java".

Co-requisites:

1. A course on "Web Technologies".

Course Objectives:

- To understand the various phases in the design of a compiler.
- To understand the design of top-down and bottom-up parsers.
- To understand syntax directed translation schemes.
- To introduce lex and yacc tools.

Course Outcomes:

- Design, develop, and implement a compiler for any language.
- Use lex and yacc tools for developing a scanner and a parser.
- Design and implement LL and LR parsers.

List of Experiments

1. Implementation of symbol table.
2. Develop a lexical analyzer to recognize a few patterns inc (ex. Identifiers, constants, comments, operators etc.)
3. Implementation of lexical analyzer using lex tool.
4. Generate yacc specification for a few syntactic categories.
 - a) Program to recognize a valid arithmetic expression that uses operator +, -, * and /.
 - b) Program to recognize a valid variable which starts with a letter followed by any number of letter or digits.
 - c) Implementation of calculator using lex and yacc.
5. Convert the bnf rules into yacc form and write code to generate abstract syntax tree.
6. Implement type checking
7. Implement any one storage allocation strategies (heap, stack, static)
8. Write a lex program to count the number of words and number of lines in a given file or program.
9. Write a 'C' program to implement lexical analyzer using c program.
10. write recursive descent parser for the grammar $E \rightarrow E+T$ $E \rightarrow T$ $T \rightarrow T^*F$ $T \rightarrow F$
 $F \rightarrow (E)/id$.
11. write recursive descent parser for the grammar $S \rightarrow (L)$ $S \rightarrow a$
 $L \rightarrow L,S$ $L \rightarrow S$
12. Write a C program to calculate first function for the grammar
 $E \rightarrow E+T$ $E \rightarrow T$ $T \rightarrow T^*F$ $T \rightarrow F$ $F \rightarrow (E)/id$
13. Write a YACC program to implement a top down parser for the given grammar.
14. Write a YACC program to evaluate algebraic expression.

TEXT BOOK:

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

REFERENCE BOOKS:

1. Lex & Yacc - John R. Levine, Tony Mason, Doug Brown, O'reilly
2. Compiler Construction, Loudon, Thomson.

22EC611PC: EMBEDDED SYSTEMS LAB**B.Tech. III Year II Sem.****L T P C**
0 0 2 1**Pre-requisites:**

1. A course on "Digital Logic Design and Microprocessors".
2. A course on "Computer Organization and Architecture".

Course Objectives:

- To provide an overview of principles of Embedded System
- To provide a clear understanding of role of firmware, operating systems in correlation with hardware systems.

Course Outcomes:

- Expected to understand the selection procedure of processors in the embedded domain.
- Design procedure of embedded firm ware.
- Expected to visualize the role of real-time operating systems in embedded systems.
- Expected to evaluate the correlation between task synchronization and latency issues

The following experiments have to be executed using

- i) Microcontroller Kits (8051/ Raspberry Pi /Arduino)
- (ii) Use embedded C/ Python/ assembly language.

List of Experiments:

1. Programs to perform arithmetic, logical, branching, and loop operations by a microcontroller.
2. Generate time delay using timers in a microcontroller.
3. Write a C program to count the number of times the switch is pressed and released.
4. Illustrate the use of a port header file (Port M) using an interface consisting of a keypad and LCD.
5. write a program to display "Hello world" on display of the receiving microcontroller using RS232.
6. Flash the operating System onto the device into a stable functional state by porting desktop environment with naway Packages necessary packages
7. Program available GPIO Plus of the corresponding device using native programming language, interface LEDs and interface LED / Switches and test it's functionality
8. Using the light sensor, monitor the light intensity and automatically turn ON/OFF LED.(for Predefined threshold light intensity value)
9. Dice game simulation- generate a random value Similar to dice value and display the same using 16x2 LCD
10. Export display to other system using available desktop display as display for the device using SSH client and X11 display seven.
11. Hosting a website on Board- Build and host a simple website (static) on the device and make it accessible online. (Need to install Sewn (e.g., Apache).)
12. Interface a regular USB webcam to the device and turn it into fully functional IP webcam

Note: Devices include Arduino, Raspberry Pi, and BeagleBour.

TEXT BOOK:

1. Embedded Systems, Raj Kamal, 2nd edition, Tata Mc Graw Hill
2. Shibu K V, "Introduction to Embedded Systems", Second Edition, Mc Graw Hill

REFERENCE BOOKS:

1. Rajkamal, Embedded Systems Architecture, Programming and Design, Tata McGraw-Hill
2. Frank Vahid and Tony Givargis, "Embedded Systems Design" - A Unified Hardware/Software Introduction, John Wiley
3. Lyla, "Embedded Systems" -Pearson
4. David E. Simon, An Embedded Software Primer, Pearson Education Asia, First Indian Reprint 2000.

22IT611PE: FULL STACK DEVELOPMENT LAB (Professional Elective – III)
(Common To CSE, CSE(DS) & IT)

B.Tech. III Year II Sem.

L T P C
0 0 2 1

Pre-Requisites:

1. Object Oriented Programming
2. Web Technologies

Course Objectives:

- Introduce fast, efficient, interactive and scalable web applications using run time environment provided by the full stack components.

Course Outcomes:

- Design flexible and responsive Web applications using Node JS, React, Express and Angular.
- Perform CRUD operations with MongoDB on huge amount of data.
- Develop real time applications using react components.
- Use various full stack modules to handle http requests and responses.

List of Experiments

1. Create an application to setup node JS environment and display "Hello World".
2. Create a Node JS application for user login system.
3. Write a Node JS program to perform read, write and other operations on a file.
4. Write a Node JS program to read form data from query string and generate response using NodeJS
5. Create a food delivery website where users can order food from a particular restaurant listed in the website for handling http requests and responses using NodeJS.
6. Implement a program with basic commands on databases and collections using MongoDB.
7. Implement CRUD operations on the given dataset using MongoDB.
8. Perform Count, Limit, Sort, and Skip operations on the given collections using MongoDB.
9. Develop an angular JS form to apply CSS and Events.
10. Develop a Job Registration form and validate it using angular JS.
11. Write an angular JS application to access JSON file data of an employee from a server using \$http service.
12. Develop a web application to manage student information using Express and Angular JS.
13. Write a program to create a simple calculator Application using React JS.
14. Write a program to create a voting application using React JS
15. Develop a leave management system for an organization where users can apply different types of leaves such as casual leave and medical leave. They also can view the available number of days using react application.
16. Build a music store application using react components and provide routing among the web pages.
17. Create a react application for an online store which consist of registration, login, product information pages and implement routing to navigate through these pages.

TEXT BOOKS:

1. Brad Dayley, Brendan Dayley, Caleb Dayley., Node.js, MongoDB and Angular Web Development, 2nd Edition, Addison-Wesley, 2019.
2. Mark Tielens Thomas., React in Action, 1st Edition, Manning Publications.

REFERENCE BOOKS:

1. Vasan Subramanian, Pro MERN Stack, Full Stack Web App Development with Mongo, Express, React, and Node, 2nd Edition, Apress, 2019.
2. Chris Northwood, The Full Stack Developer: Your Essential Guide to the Everyday Skills Expected of a Modern Full Stack Web Developer', 1st edition, Apress, 2018.
3. Brad Green & Seshadri. Angular JS. 1st Edition. O'Reilly Media, 2013.
4. Kirupa Chinnathambi, Learning React: A Hands-On Guide to Building Web Applications Using React and Redux, 2nd edition, Addison-Wesley Professional, 2018.

22IT612PE: DATA MINING LAB (Professional Elective – III)**B.Tech. III Year II Sem.****L T P C**
0 0 2 1**Prerequisites**

- A course on “Database Management System

Course Objectives:

- The course is intended to obtain hands-on experience using data mining software.
- Intended to provide practical exposure of the concepts in data mining algorithms

Course Outcomes:

- Apply preprocessing statistical methods for any given raw data.
- Gain practical experience of constructing a data warehouse.
- Implement various algorithms for data mining in order to discover interesting patterns from large amounts of data.
- Apply OLAP operations on data cube construction

LIST OF EXPERIMENTS: Experiments using Weka/ Pentaho/Python

1. Data Processing Techniques:
(i) Data cleaning (ii) Data transformation - Normalization (iii) Data integration
2. Partitioning - Horizontal, Vertical, Round Robin, Hash based
3. Data Warehouse schemas - star, snowflake, fact constellation
4. Data cube construction - OLAP operations
5. Data Extraction, Transformations & Loading operations
6. Implementation of Attribute oriented induction algorithm
7. Implementation of apriori algorithm
8. Implementation of FP - Growth algorithm
9. Implementation of Decision Tree Induction
10. Calculating Information gain measures
11. Classification of data using Bayesian approach
12. Classification of data using K - nearest neighbour approach
13. Implementation of K - means algorithm
14. Implementation of BIRCH algorithm
15. Implementation of PAM algorithm
16. Implementation of DBSCAN algorithm

TEXT BOOKS:

1. Data Mining - Concepts and Techniques - JIAWEI HAN &MICHELINE KAMBER, Elsevier.
2. Data Warehousing, Data Mining &OLAP- Alex Berson and Stephen J. Smith- Tata McGraw-Hill Edition, Tenth reprint 2007

REFERENCE BOOK:

1. Pang-Ning Tan, Michael Steinbach, Vipin Kumar, Anuj Karpatne, Introduction to Data Mining, Pearson Education

22CS613PE: SCRIPTING LANGUAGES LAB**(Professional Elective – III)****(Common To CSE, CSE(DS) & IT)****B.Tech. III Year II Sem.**

L	T	P	C
0	0	2	1

Prerequisites: Any High level programming language (C, C++)**Course Objectives**

- To Understand the concepts of scripting languages for developing web based projects
- To understand the applications the of Ruby, TCL, Perl scripting languages

Course Outcomes

- Ability to understand the differences between Scripting languages and programming languages
- Gain some fluency programming in Ruby, Perl, TCL

LIST OF EXPERIMENTS

1. Write a Ruby script to create a new string which is n copies of a given string where n is a non-negative integer
2. Write a Ruby script which accept the radius of a circle from the user and compute the parameter and area.
3. Write a Ruby script which accept the users first and last name and print them in reverse order with a space between them
4. Write a Ruby script to accept a filename from the user print the extension of that
5. Write a Ruby script to find the greatest of three numbers
6. Write a Ruby script to print odd numbers from 10 to 1
7. Write a Ruby script to check two integers and return true if one of them is 20 otherwise return their sum
8. Write a Ruby script to check two temperatures and return true if one is less than 0 and the other is greater than 100
9. Write a Ruby script to print the elements of a given array
10. Write a Ruby program to retrieve the total marks where subject name and marks of a student stored in a hash
11. Write a TCL script to find the factorial of a number
12. Write a TCL script that multiplies the numbers from 1 to 10
13. Write a TCL script for sorting a list using a comparison function
14. Write a TCL script to (i) create a list (ii) append elements to the list (iii) Traverse the list (iv) Concatenate the list
15. Write a TCL script to comparing the file modified times.
16. Write a TCL script to Copy a file and translate to native format.
17.
 - a) Write a Perl script to find the largest number among three numbers.
 - b) Write a Perl script to print the multiplication tables from 1-10 using subroutines.
18. Write a Perl program to implement the following list of manipulating functions
 - a) Shift
 - b) Unshift
 - c) Push
19.
 - a) Write a Perl script to substitute a word, with another word in a string.
 - b) Write a Perl script to validate IP address and email address.
20. Write a Perl script to print the file in reverse order using command line arguments

TEXT BOOKS:

1. The World of Scripting Languages, David Barron,Wiley Publications.
2. Ruby Programming language by David Flanagan and Yukihiro Matsumoto O'Reilly
3. "Programming Ruby" The Pratomic Progammmers guide by Dabve Thomas Second edition

REFERENCE BOOKS:

1. Open Source Web Development with LAMP using Linux Apache, MySQL, Perl and PHP, J.Lee and B. Ware (Addison Wesley) Pearson Education.
2. Perl by Example, E. Quigley, Pearson Education.
3. Programming Perl, Larry Wall, T. Christiansen and J. Orwant, O'Reilly, SPD.
4. Tcl and the Tk Tool kit, Ousterhout, Pearson Education.
5. Perl Power, J. P. Flynt, Cengage Learning.

22CS614PE: MOBILE APPLICATION DEVELOPMENT LAB**(Professional Elective – III)****(Common To CSE, CSE(DS) & IT)****B.Tech. III Year II Sem.**

L	T	P	C
0	0	2	1

Prerequisites: --- NIL---**Course Objectives:**

- To learn how to develop Applications in an android environment.
- To learn how to develop user interface applications.
- To learn how to develop URL related applications.

Course Outcomes:

- Understand the working of Android OS Practically.
- Develop user interfaces.
- Develop, deploy and maintain the Android Applications.

LIST OF EXPERIMENTS:

1. Create an Android application that shows Hello + name of the user and run it on an emulator.
- (b) Create an application that takes the name from a text box and shows hello message along with the name entered in the text box, when the user clicks the OK button.
2. Create a screen that has input boxes for User Name, Password, Address, Gender (radio buttons for male and female), Age (numeric), Date of Birth (Datepicker), State (Spinner) and a Submit button. On clicking the submit button, print all the data below the Submit Button. Use (a) Linear Layout (b) Relative Layout and (c) Grid Layout or Table Layout.
3. Develop an application that shows names as a list and on selecting a name it should show the details of the candidate on the next screen with a "Back" button. If the screen is rotated to landscape mode (width greater than height), then the screen should show list on left fragment and details on the right fragment instead of the second screen with the back button. Use Fragment transactions and Rotation event listeners.
4. Develop an application that uses a menu with 3 options for dialing a number, opening a website and to send an SMS. On selecting an option, the appropriate action should be invoked using intents.
5. Develop an application that inserts some notifications into Notification area and whenever a notification is inserted, it should show a toast with details of the notification.
6. Create an application that uses a text file to store usernames and passwords (tab separated fields and one record per line). When the user submits a login name and password through a screen, the details should be verified with the text file data and if they match, show a dialog saying that login is successful. Otherwise, show the dialog with a Login Failed message.
7. Create a user registration application that stores the user details in a database table.
8. Create a database and a user table where the details of login names and passwords are stored. Insert some names and passwords initially. Now the login details entered by the user should be verified with the database and an appropriate dialog should be shown to the user.
9. Create an admin application for the user table, which shows all records as a list and the admin can select any record for edit or modify. The results should be reflected in the table.
10. Develop an application that shows all contacts of the phone along with details like name, phone number, mobile number etc.
11. Create an application that saves user information like name, age, gender etc. in shared preference and retrieves them when the program restarts.
12. Create an alarm that rings every Sunday at 8:00 AM. Modify it to use a time picker to set alarm time.

TEXT BOOKS:

1. Professional Android 4 Application Development, Reto Meier, Wiley India, (Wrox), 2012.
2. Android Application Development for Java Programmers, James C Sheusi, Cengage, 2013.

REFERENCE BOOK:

1. Beginning Android 4 Application Development, Wei-Meng Lee, Wiley India (Wrox), 2013.

22CS615PE: SOFTWARE TESTING METHODOLOGIES LAB**(Professional Elective – III)****(Common To CSE, CSE(DS) & IT)****B.Tech. III Year II Sem.****L T P C**
0 0 2 1**Prerequisites**

- A basic knowledge of programming.

Course Objectives

- To provide knowledge of software testing methods.
- To develop skills in automation of software testing and software test automation management using the latest tools.

Course Outcomes

0. Design and develop the best test strategies in accordance with the development model.
1. Design and develop GUI, Bitmap and database checkpoints
2. Develop database checkpoints for different checks
3. Perform batch testing with and without parameter passing

List of Experiments

1. Recording in context sensitive mode and analog mode
2. GUI checkpoint for single property
3. GUI checkpoint for single object/window
4. GUI checkpoint for multiple objects
5.
 - a. Bitmap checkpoint for object/window
 - b. Bitmap checkpoint for screen area
6. Database checkpoint for Default check
7. Database checkpoint for custom check
8. Database checkpoint for runtime record check
9.
 - a. Data driven test for dynamic test data submission
 - b. Data driven test through flat files
 - c. Data driven test through front grids
 - d. Data driven test through excel test
10.
 - a. Batch testing without parameter passing
 - b. Batch testing with parameter passing
11. Data driven batch
12. Silent mode test execution without any interruption
13. Test case for calculator in windows application

TEXT BOOKS

1. Software Testing techniques, Baris Beizer, 2nd Edition, Dreamtech.
2. Software Testing Tools, Dr. K.V.K.K.Prasad, Dreamtech.

REFERENCE BOOKS

1. The craft of software testing, Brian Marick, Pearson Education.
2. Software Testing Techniques - SPD(Oreille)
3. Software Testing in the Real World, Edward Kit, Pearson.
4. Effective methods of Software Testing, Perry, John Wiley.
5. Art of Software Testing, Meyers, John Wiley.

22CS606PC: SKILL DEVELOPMENT COURSE
(Common to CSE & IT)

B.Tech. III Year II Sem.

L T P C
0 0 4 2

Course Objectives:

- The main objective of the course is to process Big Data with advance architecture like spark and streaming data in Spark

Course Outcomes:

0. Develop MapReduce Programs to analyze large dataset Using Hadoop and Spark
1. Write Hive queries to analyze large dataset Outline the Spark Ecosystem and its components
2. Perform the filter, count, distinct, map, flatMap RDD Operations in Spark.
3. Build Queries using Spark SQL
4. Apply Spark joins on Sample Data Sets
5. Make use of sqoop to import and export data from hadoop to database and vice-versa

List of Experiments:

1. To Study of Big Data Analytics and Hadoop Architecture
 - (i) know the concept of big data architecture
 - (ii) know the concept of Hadoop architecture
2. Loading DataSet in to HDFS for Spark Analysis
Installation of Hadoop and cluster management
 - (i) Installing Hadoop single node cluster in ubuntu environment
 - (ii) Knowing the differencing between single node clusters and multi-node clusters
 - (iii) Accessing WEB-UI and the port number
 - (iv) Installing and accessing the environments such as hive and sqoop
3. File management tasks & Basic linux commands
 - (i) Creating a directory in HDFS
 - (ii) Moving forth and back to directories
 - (iii) Listing directory contents
 - (iv) Uploading and downloading a file in HDFS
 - (v) Checking the contents of the file
 - (vi) Copying and moving files
 - (vii) Copying and moving files between local to HDFS environment
 - (viii) Removing files and paths
 - (ix) Displaying few lines of a file
 - (x) Display the aggregate length of a file
 - (xi) Checking the permissions of a file
 - (xii) Zipping and unzipping the files with & without permission pasting it to a location
 - (xiii) Copy, Paste commands
4. Map-reducing
 - (i) Definition of Map-reduce
 - (ii) Its stages and terminologies
 - (iii) Word-count program to understand map-reduce (Mapper phase, Reducer phase, Driver code)
5. Implementing Matrix-Multiplication with Hadoop Map-reduce
6. Compute Average Salary and Total Salary by Gender for an Enterprise.

7. (i) Creating hive tables (External and internal)
 (ii) Loading data to external hive tables from sql tables(or)Structured c.s.v using scoop
 (iii) Performing operations like filterations and updations
 (iv) Performing Join (inner, outer etc)
 (v) Writing User defined function on hive tables
8. Create a sql table of employees Employee table with id,designation Salary table (salary ,dept id) Create external table in hive with similar schema of above tables,Move data to hive using scoop and load the contents into tables,filter a new table and write a UDF to encrypt the table with AES-algorithm, Decrypt it with key to show contents
9. (i) Pyspark Definition(Apache Pyspark) and difference between Pyspark, Scala, pandas
 (ii) Pyspark files and class methods
 (iii) get(file name)
 (iv) get root directory()
10. Pyspark -RDD'S
 (i) what is RDD's?
 (ii) ways to Create RDD
 (iii) parallelized collections
 (iv) external dataset
 (v) existing RDD's
 (vi) Spark RDD's operations (Count, foreach(), Collect, join,Cache())
11. Perform pyspark transformations
 (i) map and flatMap
 (ii) to remove the words, which are not necessary to analyze this text.
 (iii) groupBy
 (iv) What if we want to calculate how many times each word is coming in corpus ?
 (v) How do I perform a task (say count the words 'spark' and 'apache' in rdd3) separatly on each partition and get the output of the task performed in these partition ?
 (vi) unions of RDD
 (vii) join two pairs of RDD Based upon their key
12. Pyspark sparkconf-Attributes and applications
 (i) What is Pyspark spark conf ()
 (ii) Using spark conf create a spark session to write a dataframe to read details in a c.s.v and later move that c.s.v to another location

TEXT BOOKS:

1. Spark in Action, Marko Bonaci and Petar Zecevic, Manning.
2. PySpark SQL Recipes: With HiveQL, Dataframe and Graphframes, Raju Kumar Mishra and Sundar Rajan Raman, Apress Media.

WEB LINKS:

1. https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_0133015058445189122518_2_shared/overview
2. https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01258388119638835242_shared/overview
3. https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_0126052684230082561692_shared/overview

22MC609ES: ENVIRONMENTAL SCIENCE**B.Tech. III Year II Sem.**

L	T	P	C
3	0	0	0

Course Objectives:

- Understanding the importance of ecological balance for sustainable development.
- Understanding the impacts of developmental activities and mitigation measures
- Understanding the environmental policies and regulations

Course Outcomes: Based on this course, the Engineering graduate will understand /evaluate / develop technologies on the basis of ecological principles and environmental regulations which in turn helps in sustainable development

UNIT - I

Ecosystems: Definition, Scope and Importance of ecosystem. Classification, structure, and function of an ecosystem, Food chains, food webs, and ecological pyramids. Flow of energy, Biogeochemical cycles, Bioaccumulation, Biomagnification, ecosystem value, services and carrying capacity, Field visits.

UNIT - II

Natural Resources: Classification of Resources: Living and Non-Living resources, **water resources:** use and over utilization of surface and ground water, floods and droughts, Dams: benefits and problems. **Mineral resources:** use and exploitation, environmental effects of extracting and using mineral resources, **Land resources:** Forest resources, **Energy resources:** growing energy needs, renewable and non renewable energy sources, use of alternate energy source, case studies.

UNIT - III

Biodiversity And Biotic Resources: Introduction, Definition, genetic, species and ecosystem diversity. Value of biodiversity; consumptive use, productive use, social, ethical, aesthetic and optional values. India as a mega diversity nation, Hot spots of biodiversity. Field visit. Threats to biodiversity: habitat loss, poaching of wildlife, man-wildlife conflicts; conservation of biodiversity: In-Situ and Ex-situ conservation. National Biodiversity act.

UNIT - IV

Environmental Pollution and Control Technologies: Environmental Pollution: Classification of pollution, **Air Pollution:** Primary and secondary pollutants, Automobile and Industrial pollution, Ambient air quality standards. **Water pollution:** Sources and types of pollution, drinking water quality standards. **Soil Pollution:** Sources and types, Impacts of modern agriculture, degradation of soil. **Noise Pollution:** Sources and Health hazards, standards, **Solid waste:** Municipal Solid Waste management, composition and characteristics of e-Waste and its management. **Pollution control technologies:** Wastewater Treatment methods: Primary, secondary and Tertiary. Overview of air pollution control technologies, Concepts of bioremediation. **Global Environmental Problems and Global Efforts:** Climate change and impacts on human environment. Ozone depletion and Ozone depleting substances (ODS). Deforestation and desertification. International conventions / Protocols: Earth summit, Kyoto protocol, and Montréal Protocol.

UNIT - V

Environmental Policy, Legislation & EIA: Environmental Protection act, Legal aspects Air Act- 1981, Water Act, Forest Act, Wild life Act, Municipal solid waste management and handling rules, biomedical waste management and handling rules, hazardous waste management and handling rules. EIA: EIA structure, methods of baseline data acquisition. Overview on Impacts of air, water, biological and Socio-economical aspects. Strategies for risk assessment, Concepts of Environmental Management Plan

(EMP). **Towards Sustainable Future:** Concept of Sustainable Development, Population and its explosion, Crazy Consumerism, Environmental Education, Urban Sprawl, Human health, Environmental Ethics, Concept of Green Building, Ecological Foot Print, Life Cycle assessment (LCA), Low carbon life style.

TEXT BOOKS:

1. Textbook of Environmental Studies for Undergraduate Courses by Erach Bharucha for University Grants Commission.
2. Environmental Studies by R. Rajagopalan, Oxford University Press.

REFERENCE BOOKS:

1. Environmental Science: towards a sustainable future by Richard T. Wright. 2008 PHL Learning Private Ltd. New Delhi.
2. Environmental Engineering and science by Gilbert M. Masters and Wendell P. Ela. 2008 PHI Learning Pvt. Ltd.
3. Environmental Science by Daniel B. Botkin & Edward A. Keller, Wiley INDIA edition.
4. Environmental Studies by Anubha Kaushik, 4th Edition, New age international publishers.
5. Text book of Environmental Science and Technology - Dr. M. Anji Reddy 2007, BS Publications.

22IT701PC: INFORMATION SECURITY**B.Tech. IV Year I Sem.****L T P C**
0 0 2 1**Prerequisites**

1. A Course on "Computer Networks and a course on Mathematics

Course Objectives

- To understand the fundamentals of Cryptography
- To understand various key distribution and management schemes
- To understand how to deploy encryption techniques to secure data in transit across data networks
- To apply algorithms used for secure transactions in real world applications

Course Outcomes

- Demonstrate the knowledge of cryptography, network security concepts and applications.
- Ability to apply security principles in system design.
- Ability to identify and investigate vulnerabilities and security threats and mechanisms to counter them.

UNIT - I

Security Attacks (Interruption, Interception, Modification and Fabrication), Security Services (Confidentiality, Authentication, Integrity, Non-repudiation, access Control and Availability) and Mechanisms, A model for Internetwork security

Classical Encryption Techniques: DES, Strength of DES, Differential and Linear Cryptanalysis, Block Cipher Design Principles and Modes of operation, Blowfish, Placement of Encryption Function, Traffic Confidentiality, key Distribution, Random Number Generation.

UNIT - II

Public key Cryptography Principles, RSA algorithm, Key Management, Diffie-Hellman Key Exchange, Elliptic Curve Cryptography.

Message authentication and Hash Functions: Authentication Requirements and Functions, Message Authentication, Hash Functions and MACs Hash and MAC Algorithms SHA-512, HMAC.

UNIT - III

Digital Signatures: Authentication Protocols, Digital signature Standard, Authentication Applications, Kerberos, X.509 Directory Authentication Service.

Email Security: Pretty Good Privacy (PGP) and S/MIME.

UNIT - IV

IP Security: Overview, IP Security Architecture, Authentication Header, Encapsulating Security Payload, Combining Security Associations and Key Management.

Web Security: Web Security Requirements, Secure Socket Layer (SSL) and Transport Layer Security (TLS), Secure Electronic Transaction (SET).

UNIT - V

Intruders, Viruses and Worms Intruders, Viruses and related threats Firewalls: Firewall Design Principles, Trusted Systems, Intrusion Detection Systems.

TEXT BOOK:

1. Cryptography and Network Security (principles and approaches) by William Stallings Pearson Education, 4th Edition.

REFERENCE BOOKS:

1. Network Security Essentials (Applications and Standards) by William Stallings Pearson Education.
2. Principles of Information Security, Whitman, Thomson.

22IT702PC: CLOUD COMPUTING**B.Tech. IV Year I Sem.****L T P C**
3 0 0 3**Pre-requisites:**

1. A course on "Computer Networks".
2. A course on "Operating System".

Course Objectives:

- This course provides an insight into cloud computing
- Topics covered include- Cloud Computing Architecture, Deployment Models, Service Models, Technological Drivers for Cloud Computing, Networking for Cloud Computing and Security in Cloud Computing

Course Outcomes:

- Understand different computing paradigms and potential of the paradigms and specifically cloud computing
- Understand cloud service types, cloud deployment models and technologies supporting and driving the cloud
- Acquire the knowledge of programming models for cloud and development of software application that runs the cloud and various services available from major cloud providers
- Understand the security concerns and issues in cloud computing
- Acquire the knowledge of advances in cloud computing.

UNIT - I

Computing Paradigms, Cloud Computing Fundamentals, Cloud Computing Architecture and Management

UNIT - II

Cloud Deployment Models, Cloud Service Models, Technological Drivers for Cloud Computing: SOA and Cloud, Multicore Technology, Web 2.0 and Web 3.0, Pervasive Computing, Operating System, Application Environment

UNIT - III

Virtualization, Programming Models for Cloud Computing: MapReduce, Cloud Haskell, Software Development in Cloud

UNIT - IV

Networking for Cloud Computing: Introduction, Overview of Data Center Environment, Networking Issues in Data Centers, Transport Layer Issues in DCNs, Cloud Service Providers

UNIT - V

Security in Cloud Computing, and Advanced Concepts in Cloud Computing

TEXT BOOK:

1. Chandrasekaran, K. *Essentials of cloud computing*. CRC Press, 2014

REFERENCE BOOKS:

1. Cloud Computing: Principles and Paradigms, Editors: Rajkumar Buyya, James Broberg, Andrzej M. Goscinski, Wiley, 2011
2. Enterprise Cloud Computing - Technology, Architecture, Applications, Gautam Shroff, Cambridge University Press, 2010
3. Cloud Computing Bible, Barrie Sosinsky, Wiley-India, 2010

22IT741PE: HUMAN COMPUTER INTERACTION
(Professional Elective - IV)

B.Tech. IV Year I Sem.

L T P C
3 0 0 3

Course Objectives:

- To gain an overview of Human-Computer Interaction (HCI)
- Understanding the alternatives to traditional "keyboard and mouse" computing.
- Getting familiarity with the vocabulary associated with sensory and cognitive systems
- Be able to apply models from cognitive psychology to predicting user performance
- Working in small groups on a product design with invaluable team-work experience.

Course Outcomes:

0. Apply HCI and principles to interaction design.
1. Design certain tools for blind or PH people
2. Understand the social implications of technology and ethical responsibilities as engineers.
3. Understand the importance of a design and evaluation methodology

UNIT - I

Introduction: Importance of user Interface - definition, importance of good design. Benefits of good design, A brief history of Screen design.

The graphical user interface - popularity of graphics, the concept of direct manipulation, graphical system, Characteristics, Web user - Interface popularity, characteristics- Principles of user interface.

UNIT - II

Design process - Human interaction with computers, importance of human characteristics, human consideration, Human interaction speeds, understanding business junctions.

Screen Designing: Design goals - Screen planning and purpose, organizing screen elements, ordering of screen data and content - screen navigation and flow - Visually pleasing composition - amount of information - focus and emphasis - presentation information simply and meaningfully - information retrieval on web - statistical graphics - Technological consideration in interface design.

UNIT- III

Windows - New and Navigation schemes selection of window, selection of devices based and screen-based controls. Components - text and messages, Icons and increases - Multimedia, colors, uses problems, choosing colors.

UNIT- IV

HCI in the software process- The software life cycle, Usability engineering, Iterative design and prototyping, Design Focus: Prototyping in practice, Design rationale, Design rules, Principles to support usability Standards, Golden rules and heuristics, HCI patterns, Evaluation techniques, Goals of evaluation, Evaluation through expert analysis, Evaluation through user participation, Choosing an evaluation method, Universal design, Universal design principles Multimodal interaction

UNIT- V

Cognitive models Goal and task hierarchies Design Focus: GOMS saves money, Linguistic models, The challenge of display-based systems, Physical and device models, Cognitive architectures, Ubiquitous computing and augmented realities, Ubiquitous computing applications research, Design Focus: Ambient Wood - augmenting the physical, Virtual and augmented reality, Design Focus: Shared experience Design Focus: Applications of augmented reality Information and data visualization

TEXT BOOKS:

1. The essential guide to user interface design, Wilbert O Galitz, Wiley Dream Tech.

2. Human - Computer Interaction. Alan Dix, Janet Finckay, Gregory's, Abowd, Russell Bealg, Pearson Education.

REFERENCE BOOKS:

1. Designing the user interface. 3rd Edition Ben Shneidermann, Pearson Education Asia.
2. Interaction Design Prece, Rogers, Sharps. Wiley Dreamtech.
3. User Interface Design, Soren Lauesen, Pearson Education.
4. Human -Computer Interaction, D. R. Olsen, Cengage Learning.
5. Human -Computer Interaction, Smith - Atakan, Cengage Learning.

22IT742PE: HIGH PERFORMANCE COMPUTING
(Professional Elective - IV)

B.Tech. IV Year I Sem.

L T P C
3 0 0 3

Prerequisites

- Computer Organization & Architecture
- Operating Systems
- Algorithms and Data Structures
- Programming Language(C/C++)

Course Objectives

- To teach students to become good at parallel computing algorithm design
- To teach students to become good at modeling and solving problems using different types of parallel computing architectures
- To teach students the ability to measure the performance of parallel algorithms and arrive at reasonable estimates of cost tradeoffs
- To teach students the various paradigms in algorithm design for computationally intensive applications
- To teach students to become good at understanding and using modern multi-processor and multi-core architectures

Course Outcomes:

- Understand different parallel computing architectures and networks
- Design parallel algorithms and measure their performance
- Understand vector processing, memory bottlenecks, data and thread-level parallelism
- Understand the various programming frameworks like MPI, OpenMP and CUDA
- Gain knowledge of writing efficient parallel programs

UNIT - I

Modern Processors: Stored-Program Computer Architecture, General-Purpose cache-based Microprocessor Architecture, Memory Hierarchies, Multicore processors, Multithreaded processors, Vector processors.

Basic optimization techniques for serial code: Scalar profiling, Common sense optimizations, Simple measures, large impact, The role of compilers, Data access optimization.

UNIT - II

Parallel computers: Taxonomy of parallel computing paradigms, Shared-memory computers, Distributed-memory computers, Hierarchical (hybrid) systems, Networks.

Basics of parallelization: Need for Parallelism, Parallel scalability

UNIT - III

Shared-memory parallel programming with OpenMP: Introduction to OpenMP, Profiling OpenMP programs, Performance pitfalls, Case study: OpenMP-parallel Jacobi algorithm.

UNIT - IV

Distributed-memory parallel programming with MPI: Message passing, Introduction to MPI, MPI performance tools, Communication parameters, Synchronization, serialization, contention, Reducing communication overhead, Case study: Parallel sparse matrix-vector multiply.

UNIT - V

CUDA: Understanding the CUDA computing model and the API using nvcc compiler, Introduction to

TEXT BOOK:

1. Introduction to Parallel Computing, Second Edition, Ananth Grama, George Karypis, Vipin Kumar, Anshul Gupta, Addison-Wesley, 2003, ISBN: 0201648652
2. Georg Hager, Gerhard Wellein, Introduction to High Performance Computing for Scientists and Engineers, Chapman & Hall / CRC Computational Science series, 2011.

REFERENCE BOOKS:

1. CUDA Programming A Developer's Guide to Parallel Computing with GPUs by Shane Cook, Morgan Kaufman Publishers
2. Parallel Computing - Theory and Practice, Second Edition, Michael J. Quinn, Tata McGraw-Hill Edition.
3. Parallel Computers - Architectures and Programming, V. Rajaraman, C. Siva Ram Murthy, PHI.
4. Parallel Programming in C with MPI and OpenMP by Michael Quinn, McGraw-Hill Publisher
5. Computer Architecture A Quantitative Approach by John Hennessey and David Patterson, Morgan Kaufman Publishers

22IT743PE: ARTIFICIAL INTELLIGENCE
(Professional Elective –IV)

B.Tech. IV Year I Sem.

L T P C
3 0 0 3

Prerequisites:

1. Programming for problem solving, Data Structures.

Course Objectives:

- To learn the distinction between optimal reasoning Vs. human like reasoning
- To understand the concepts of state space representation, exhaustive search, heuristic search together with the time and space complexities.
- To learn different knowledge representation techniques.
- To understand the applications of AI, namely game playing, theorem proving, and machine learning.

Course Outcomes:

- Understand search strategies and intelligent agents
- Understand different adversarial search techniques
- Apply propositional logic, predicate logic for knowledge representation
- Apply AI techniques to solve problems of game playing, and machine learning.

UNIT - I

Introduction to AI, Intelligent Agents, problem-Solving Agents, Searching for Solutions, Uninformed Search Strategies: Breadth-first search, Uniform cost search, Depth-first search, Iterative deepening Depth-first search, Bidirectional search, Informed (Heuristic) Search Strategies: Greedy best-first search, A* search, Heuristic Functions, Beyond Classical Search: Hill-climbing search, Simulated annealing search, Local Search in Continuous Spaces

UNIT - II

Problem Solving by Search-II and Propositional Logic

Adversarial Search: Games, Optimal Decisions in Games, Alpha-Beta Pruning, Imperfect Real-Time Decisions. Constraint Satisfaction Problems: Defining Constraint Satisfaction Problems, Constraint Propagation, Backtracking Search for CSPs, Local Search for CSPs, The Structure of Problems. Propositional Logic: Knowledge-Based Agents, The Wumpus World, Logic, Propositional Logic, Propositional Theorem Proving: Inference and proofs, Proof by resolution, Horn clauses and definite clauses, Forward and backward chaining, Effective Propositional Model Checking, Agents Based on Propositional Logic.

UNIT - III

Logic and Knowledge Representation

First-Order Logic: Representation, Syntax and Semantics of First-Order Logic, Using First-Order Logic, Knowledge Engineering in First-Order Logic.

Inference in First-Order Logic: Propositional vs. First-Order Inference, Unification and Lifting, Forward Chaining, Backward Chaining, Resolution.

UNIT - IV

Knowledge Representation: Ontological Engineering, Categories and Objects, Events. Mental Events and Mental Objects, Reasoning Systems for Categories, Reasoning with Default Information.

Classical Planning: Definition of Classical Planning, Algorithms for Planning with State-Space Search, Planning Graphs, other Classical Planning Approaches, Analysis of Planning approaches.

UNIT - V

Uncertain knowledge and Learning Uncertainty: Acting under Uncertainty, Basic Probability Notation, Inference Using Full Joint Distributions, Independence, Bayes' Rule and Its Use

Probabilistic Reasoning: Representing Knowledge in an Uncertain Domain, The Semantics of Bayesian Networks, Efficient Representation of Conditional Distributions, Approximate Inference in Bayesian Networks, Relational and First-Order Probability, Other Approaches to Uncertain Reasoning; Dempster-Shafer theory.

TEXT BOOK:

1. Artificial Intelligence: A Modern Approach, Third Edition, Stuart Russell and Peter Norvig, Pearson Education.

REFERENCE BOOKS:

1. Artificial Intelligence, 3rd Edn, E. Rich and K. Knight (TMH)
2. Artificial Intelligence, 3rd Edn., Patrick Henry Winston, Pearson Education.
3. Artificial Intelligence, Shivani Goel, Pearson Education.
4. Artificial Intelligence and Expert systems - Patterson, Pearson Education

22IT744PE: INFORMATION RETRIEVAL SYSTEMS (Professional Elective – IV)**B.Tech. IV Year I Sem.****L T P C**
3 0 0 3**Prerequisites:**

1. Data Structures

Course Objectives:

- To learn the concepts and algorithms in Information Retrieval Systems
- To understand the data/file structures that are necessary to design, and implement information retrieval (IR) systems.

Course Outcomes:

- Ability to apply IR principles to locate relevant information large collections of data
- Ability to design different document clustering algorithms
- Implement retrieval systems for web search tasks.
- Design an Information Retrieval System for web search tasks.

UNIT - I

Introduction to Information Retrieval Systems: Definition of Information Retrieval System, Objectives of Information Retrieval Systems, Functional Overview, Relationship to Database Management Systems, Digital Libraries and Data Warehouses Information Retrieval System Capabilities: Search Capabilities, Browse Capabilities, Miscellaneous Capabilities

UNIT - II

Cataloging and Indexing: History and Objectives of Indexing, Indexing Process, Automatic Indexing, Information Extraction Data Structure: Introduction to Data Structure, Stemming Algorithms, Inverted File Structure, N-Gram Data Structures, PAT Data Structure, Signature File Structure, Hypertext and XML Data Structures, Hidden Markov Models.

UNIT - III

Automatic Indexing: Classes of Automatic Indexing, Statistical Indexing, Natural Language, Concept Indexing, Hypertext Linkages
Document and Term Clustering: Introduction to Clustering, Thesaurus Generation, Item Clustering, Hierarchy of Clusters

UNIT - IV

User Search Techniques: Search Statements and Binding, Similarity Measures and Ranking, Relevance Feedback, Selective Dissemination of Information Search, Weighted Searches of Boolean Systems, Searching the INTERNET and Hypertext
Information Visualization: Introduction to Information Visualization, Cognition and Perception, Information Visualization Technologies

UNIT - V

Text Search Algorithms: Introduction to Text Search Techniques, Software Text Search Algorithms, Hardware Text Search Systems
Multimedia Information Retrieval: Spoken Language Audio Retrieval, Non-Speech Audio Retrieval, Graph Retrieval, Imagery Retrieval, Video Retrieval

TEXT BOOK:

1. Information Storage and Retrieval Systems – Theory and Implementation, Second Edition, Gerald J. Kowalski, Mark T. Maybury, Springer

REFERENCE BOOKS:

1. Frakes, W.B., Ricardo Baeza-Yates: Information Retrieval Data Structures and Algorithms, Prentice Hall, 1992.
2. Information Storage & Retrieval by Robert Korfhage - John Wiley & Sons.
3. Modern Information Retrieval by Yates and Neto Pearson Education.

22CS745PE: AD HOC & SENSOR NETWORKS (Professional Elective – IV)
(Common To CSE & IT)

B.Tech. IV Year I Sem.

L T P C
3 0 0 3

Prerequisites

- Computer Networks
- Distributed Systems
- Mobile Computing

Course Objectives

- To understand the challenges of routing in ad-hoc and sensor networks
- To understand various broadcast, multicast and geocasting protocols in ad hoc and sensor networks
- To understand basics of Wireless sensors, and Lower Layer Issues and Upper Layer Issues of WSN

Course Outcomes

- Understand the concepts of sensor networks and applications
- Understand and compare the MAC and routing protocols for adhoc networks
- Understand the transport protocols of sensor networks

UNIT - I

Introduction to Ad Hoc Networks

Characteristics of MANETs, Applications of MANETs and Challenges of MANETs.

Routing in MANETs

Criteria for classification, Taxonomy of MANET routing algorithms, *Topology-based* routing algorithms- Proactive: DSDV, WRP; Reactive: DSR, AODV, TORA; Hybrid: ZRP; *Position-based* routing algorithms- Location Services-DREAM, Quorum-based, GLS; Forwarding Strategies, Greedy Packet, Restricted Directional Flooding-DREAM, LAR; Other routing algorithms-QoS Routing, CEDAR.

UNIT - II

Data Transmission

Broadcast Storm Problem, Rebroadcasting Schemes-Simple-flooding, Probability-based Methods, Area-based Methods, Neighbour Knowledge-based: SBA, Multipoint Relaying, AHBP. Multicasting: Tree-based: AMRIS, MAODV; Mesh-based: ODMRP, CAMP; Hybrid: AMRoute, MCEDAR.

UNIT - III

Geocasting

Data-transmission Oriented-LBM; Route Creation Oriented-GeoTORA, MGR.

TCP over Ad Hoc TCP protocol overview, TCP and MANETs, Solutions for TCP over Ad hoc

UNIT - IV

Basics of Wireless Sensors and Lower Layer Issues-Applications, Classification of sensor networks, Architecture of sensor network, Physical layer, MAC layer, Link layer, Routing Layer.

UNIT - V

Upper Layer Issues of WSN

Transport layer, High-level application layer support, Adapting to the inherent dynamic nature of WSNs.

TEXT BOOKS

1. Ad Hoc and Sensor Networks - Theory and Applications, *Carlos Corderio Dharma P. Aggarwal*, World Scientific Publications, March 2006, ISBN - 981-256-681-3
2. Wireless Sensor Networks: An Information Processing Approach, Feng Zhao, Leonidas Guibas, Elsevier Science, ISBN - 978-1-55860-914-3 (Morgan Kauffman)

REFERENCE BOOKS:

1. C. Siva Ram Murthy, B.S. Manoj Ad Hoc Wireless Networks: Architectures and Protocols.
2. Taieb Znati Kazem Sohraby, Daniel Minoli, Wireless Sensor Networks: Technology, Protocols and Applications, Wiley.

22IT751PE: INTRUSION DETECTION SYSTEMS (Professional Elective – V)**B.Tech. IV Year I Sem.****L T P C****3 0 0 3****Prerequisites:** Computer Networks, Computer Programming**Course Objectives:**

- Compare alternative tools and approaches for Intrusion Detection through quantitative analysis to determine the best tool or approach to reduce risk from intrusion.
- Identify and describe the parts of all intrusion detection systems and characterize new and emerging IDS technologies according to the basic capabilities all intrusion detection systems share.

Course Outcomes:

- Understand fundamental knowledge of intrusion detection and prevention
- Understand different types of attacks in network layer and code injection human layer
- Analyze different anomaly detection algorithms

UNIT - I

The state of threats against computers, and networked systems-Overview of computer security solutions and why they fail-Vulnerability assessment, firewalls, VPN's -Overview of Intrusion Detection and Intrusion Prevention, Network and Host-based IDS

UNIT - II

Classes of attacks - Network layer: scans, denial of service, penetration Application layer: software exploits, code injection-Human layer: identity theft, root access-Classes of attackers-Kids/hackers/sop Hesitated groups-Automated: Drones, Worms, Viruses

UNIT - III

A General IDS model and taxonomy, Signature-based Solutions, Snort, Snort rules, Evaluation of IDS, Cost sensitive IDS

UNIT - IV

Anomaly Detection Systems and Algorithms-Network Behavior Based Anomaly Detectors (rate based)-Host-based Anomaly Detectors-Software Vulnerabilities-State transition, Immunology, Payload Anomaly Detection

UNIT - V

Attack trees and Correlation of alerts- Autopsy of Worms and Botnets-Malware detection -Obfuscation, polymorphism- Document vectors, Email/IM security issues-Viruses/Spam-From signatures to thumbprints to zero day detection-Insider, Threat issues-Taxonomy-Masquerade and Impersonation Traitors, Decoys and Deception-Future: Collaborative Security

TEXT BOOKS:

1. Peter Szor, The Art of Computer Virus Research and Defense, Symantec Press ISBN 0-321-30545-3.
2. Markus Jakobsson and Zulfikar Ramzan, Crimeware, Understanding New Attacks and Defenses.

REFERENCE BOOKS:

1. Saiful Hasan, Intrusion Detection System, Kindle Edition.
2. Ankit Fadia, Intrusion Alert: An Ethical Hacking Guide to Intrusion Detection.

ONLINE WEBSITES/MATERIALS:

1. <https://www.intechopen.com/books/intrusion-detection-systems/>

ONLINE COURSES:

1. <https://www.sans.org/course/intrusion-detection-in-depth>
2. <https://www.cybrary.it/skill-certification-course/ids-ips-certification-training-course>

22IT752PE: REAL TIME SYSTEMS (Professional Elective – V)**B.Tech. IV Year I Sem.****L T P C**
3 0 0 3**Prerequisite:** Basic Programming/C/C++ Programming, Computer Organization and Operating System**Course Objectives**

- To provide a broad understanding of the requirements of Real Time Operating Systems.
- To make the student understand, applications of these Real Time features using case studies.

Course Outcomes:

- Understand the key concepts of Real-Time systems.
- To facilitate task scheduling and designing concurrency within an application using Semaphores, Message queues.
- Explore other kernel objects common to embedded system development.
- Attain knowledge of exception and interrupt handling in real time systems
- Understand real time operating systems like RT Linux, VxWorks, MicroC /OSII, TinyOs

UNIT – I**Introduction:** Introduction to UNIX/LINUX, Overview of Commands, File I/O,(open, create, close, lseek, read, write), Process Control (fork, vfork, exit, wait, waitpid, exec).**UNIT - II****Real Time Operating Systems:** Brief History of OS, Defining RTOS, The Scheduler, Objects, Services, Characteristics of RTOS, Defining a Task, asks States and Scheduling, Task Operations, Structure, Synchronization, Communication and Concurrency. Defining Semaphores, Operations and Use, Defining Message Queue, States, Content, Storage, Operations and Use**UNIT - III**

Objects, Services and I/O: Pipes, Event Registers, Signals, Other Building Blocks, Component Configuration, Basic I/O Concepts, I/O Subsystem

UNIT - IV

Exceptions, Interrupts and Timers: Exceptions, Interrupts, Applications, Processing of Exceptions and Spurious Interrupts, Real Time Clocks, Programmable Timers, Timer Interrupt Service Routines (ISR), Soft Timers, Operations.

UNIT - V**Case Studies of RTOS:** RT Linux, MicroC/OS-II, VxWorks, Embedded Linux, and Tiny OS.**TEXT BOOK:**

1. Real Time Concepts for Embedded Systems - Qing Li, Elsevier, 2011
2. Embedded Systems- Architecture, Programming and Design by Rajkamal, 2007, TMH.

REFERENCE BOOKS:

1. Advanced UNIX Programming, Richard Stevens
2. Embedded Linux: Hardware, Software and Interfacing - Dr. Craig Hollabaugh

22IT753PE: DEEP LEARNING (Professional Elective –V)**B.Tech. IV Year I Sem.**

L	T	P	C
3	0	0	3

Course Objectives:

- To understand deep Learning algorithms and their applications in real-world data

Course Outcomes:

- Understand machine learning basics and neural networks
- Understand optimal usage of data for training deep models
- Apply CNN and RNN models for real-world data
- Evaluate deep models
- Develop deep models for real-world problems

UNIT -I**Machine Learning Basics**

Learning Algorithms, Capacity, Overfitting and Underfitting, Hyperparameters and Validation Sets, Estimators, Bias and Variance, Maximum Likelihood Estimation, Bayesian Statistics, Supervised Learning Algorithms, Unsupervised Learning Algorithms, Stochastic Gradient Descent, Building a Machine Learning Algorithm, Challenges Motivating Deep Learning

Deep Feedforward Networks Learning XOR, Gradient-Based Learning, Hidden Units, Architecture Design, Back-Propagation and Other Differentiation Algorithms

UNIT -II**Regularization for Deep Learning**

Parameter Norm Penalties, Norm Penalties as Constrained Optimization, Regularization and Under-Constrained Problems, Dataset Augmentation, Noise Robustness, Semi-Supervised Learning, Multi-Task Learning, Early Stopping, Parameter Tying and Parameter Sharing, Sparse Representations, Bagging and Other Ensemble Methods, Dropout, Adversarial Training, Tangent Distance, Tangent Prop, and Manifold Tangent Classifier, Optimization for Training Deep Models, Learning vs Pure Optimization, Challenges in Neural Network Optimization, Basic Algorithms, Parameter Initialization Strategies, Algorithms with Adaptive Learning Rates

UNIT-III**Convolutional Networks**

The Convolution Operation, Motivation, Pooling, Convolution and Pooling as an Infinitely Strong Prior, Variants of the Basic Convolution Function, Structured Outputs, Data Types, Efficient Convolution Algorithms, Random or Unsupervised Features

UNIT -IV**Recurrent and Recursive Nets**

Unfolding Computational Graphs, Recurrent Neural Networks, Bidirectional RNNs, Encoder-Decoder Sequence-to-Sequence Architectures, Deep Recurrent Networks, Recursive Neural Networks, The Challenge of Long-Term Dependencies, Echo State Networks, Leaky Units and Other Strategies for Multiple Time Scales, The Long Short-Term Memory and Other Gated RNNs, Optimization for Long-Term Dependencies, Explicit Memory

UNIT -V

Practical Methodology: Performance Metrics, Default Baseline Models, Determining Whether to Gather More Data, Selecting Hyperparameters, Debugging Strategies, Example: Multi-Digit Number Recognition

Applications: Large-Scale Deep Learning, Computer Vision, Speech Recognition, Natural Language Processing, Other Applications.

TEXT BOOK:

1. Deep Learning by Ian Goodfellow, Yoshua Bengio and Aaron Courville, MIT Press.

REFERENCE BOOKS:

1. The Elements of Statistical Learning. Hastie, R. Tibshirani, and J. Friedman, Springer.
2. Probabilistic Graphical Models. Koller, and N. Friedman, MIT Press.
3. Bishop, C., M., Pattern Recognition and Machine Learning, Springer, 2006.
4. Yegnanarayana, B., Artificial Neural Networks PHI Learning Pvt. Ltd, 2009.
5. Golub, G., H., and Van Loan, C.,F., Matrix Computations, JHU Press, 2013.
6. Satish Kumar, Neural Networks: A Classroom Approach, Tata McGraw-Hill Education, 2004.

22IT754PE: BLOCKCHAIN TECHNOLOGY (Professional Elective – V)**B.Tech. IV Year I Sem.****L T P C**
3 0 0 3**Prerequisites:**

1. Knowledge in information security and applied cryptography.
2. Knowledge in Computer Networks

Course Objectives:

- To learn the fundamentals of Blockchain and various types of block chain and consensus mechanisms.
- To understand the public block chain system, Private block chain system and consortium blockchain.
- Able to know the security issues of blockchain technology.

Course Outcomes:

- Understanding concepts behind crypto currency
- Applications of smart contracts in decentralized application development
- Understand frameworks related to public, private and hybrid blockchain
- Create blockchain for different application case studies

UNIT-I

Fundamentals of Blockchain: Introduction, Origin of Blockchain, Blockchain Solution, Components of Blockchain, Block in a Blockchain, The Technology and the Future.

Blockchain Types and Consensus Mechanism: Introduction, Decentralization and Distribution, Types of Blockchain, Consensus Protocol.

Cryptocurrency – Bitcoin, Altcoin and Token: Introduction, Bitcoin and the Cryptocurrency, Cryptocurrency Basics, Types of Cryptocurrencies, Cryptocurrency Usage.

UNIT-II

Public Blockchain System: Introduction, Public Blockchain, Popular Public Blockchains, The Bitcoin Blockchain, Ethereum Blockchain.

Smart Contracts: Introduction, Smart Contract, Characteristics of a Smart Contract, Types of Smart Contracts, Types of Oracles, Smart Contracts in Ethereum, Smart Contracts in Industry.

UNIT-III

Private Blockchain System: Introduction, Key Characteristics of Private Blockchain, Need of Private Blockchain, Private Blockchain Examples, Private Blockchain and Open Source, E-commerce Site Example, Various Commands (Instructions) in E-commerce Blockchain, Smart Contract in Private Environment, State Machine, Different Algorithms of Permissioned Blockchain, Byzantine Fault, Multichain.

Consortium Blockchain: Introduction, Key Characteristics of Consortium Blockchain, Need of Consortium Blockchain, Hyperledger Platform, Overview of Ripple, Overview of Corda.

Initial Coin Offering: Introduction, Blockchain Fundraising Methods, Launching an ICO, Investing in an ICO, Pros and Cons of Initial Coin Offering, Successful Initial Coin Offerings, Evolution of ICO, ICO Platforms.

UNIT-IV

Security in Blockchain: Introduction, Security Aspects in Bitcoin, Security and Privacy Challenges of Blockchain in General, Performance and Scalability, Identity Management and Authentication, Regulatory Compliance and Assurance, Safeguarding Blockchain Smart Contract (DApp), Security Aspects in Hyperledger Fabric.

Applications of Blockchain: Introduction, Blockchain in Banking and Finance, Blockchain in Education, Blockchain in Energy, Blockchain in Healthcare, Blockchain in Real-estate, Blockchain In Supply Chain, The Blockchain and IoT. Limitations and Challenges of Blockchain.

UNIT-V

Blockchain Case Studies: Case Study 1 - Retail, Case Study 2 - Banking and Financial Services, Case Study 3 - Healthcare, Case Study 4 - Energy and Utilities.

Blockchain Platform using Python: Introduction, Learn How to Use Python Online Editor, Basic Programming Using Python, Python Packages for Blockchain.

Blockchain platform using Hyperledger Fabric: Introduction, Components of Hyper ledger Fabric Network, Chain codes from Developer.ibm.com, Blockchain Application Using Fabric Java SDK.

TEXT BOOK:

1. "Blockchain Technology", Chandramouli Subramanian, Asha A. George, Abhilasj K A and Meena Karthikeyan, Universities Press.

REFERENCE BOOKS:

1. Michael Juntao Yuan, Building Blockchain Apps, Pearson, India.
2. Blockchain Blueprint for Economy, Melanie Swan, SPD O'reilly.
3. Blockchain for Business, Jai Singh Arun, Jerry Cuomo, Nitin Gaur, Pearson.

22CS755PE: SOFTWARE PROCESS & PROJECT MANAGEMENT
(Professional Elective – V)
(Common To CSE & IT)

B.Tech. IV Year I Sem.

L T P C
3 0 0 3

Course Objectives:

- To acquire knowledge on software process management.
- To acquire managerial skills for software project development.
- To understand software economics.

Course Outcomes:

- Understand the software process change, assessment, project plans and Quality Standards.
- Examine the life cycle phases, artifacts, workflows and checkpoints of a process.
- Design and develop software products using conventional and modern principles of software project management.
- Identify the new project management process and practices.

UNIT - I

Software Process Maturity

Software maturity Framework, Principles of Software Process Change, Software Process Assessment, The Initial Process, The Repeatable Process, The Defined Process, The Managed Process, The Optimizing Process, Process Reference Models Capability Maturity Model (CMM), CMMI, PCMM, PSP, TSP).

UNIT - II

Software Project Management Renaissance

Conventional Software Management, Evolution of Software Economics, Improving Software Economics, Life-Cycle Phases and Process artifacts

Engineering and Production stages, inception phase, elaboration phase, construction phase, transition phase, artifact sets, management artifacts, engineering artifacts and pragmatic artifacts, model-based software architectures.

UNIT - III

Workflows and Checkpoints of process

Software process workflows, Iteration workflows, Major milestones, minor milestones, periodic status assessments, Process Planning Work breakdown structures, Planning guidelines, cost and schedule estimating process, iteration planning process, Pragmatic planning.

UNIT - IV

Project Organizations

Line-of- business organizations, project organizations, evolution of organizations, process automation. Project Control and process instrumentation, The seven-core metrics, management indicators, quality indicators, life-cycle expectations, Pragmatic software metrics, metrics automation.

UNIT - V

CCPDS-R Case Study and Future Software Project Management Practices, Modern Project Profiles, Next-Generation software Economics, Modern Process Transitions.

TEXT BOOKS:

1. Managing the Software Process, Watts S. Humphrey, Pearson Education
2. Software Project Management, Walker Royce, Pearson Education

REFERENCE BOOKS:

1. An Introduction to the Team Software Process, Watts S. Humphrey, Pearson Education, 2000
2. Process Improvement essentials, James R. Persse, O'Reilly, 2006
3. Software Project Management, Bob Hughes & Mike Cotterell, fourth edition, TMH, 2006
4. Applied Software Project Management, Andrew Stellman & Jennifer Greene, O'Reilly, 2006.
5. Software Engineering Project Management, Richard H. Thayer & Edward Yourdon, 2nd edition, Wiley India, 2004.
6. Agile Project Management, Jim Highsmith, Pearson education, 2004.

22IT7210E: FULL STACK DEVELOPMENT (Open Elective – II)**B.Tech. IV Year I Sem.****L T P C**
3 0 0 3**Pre-Requisites:**

1. Object Oriented Programming
2. Web Technologies

Course Objectives:

- Students will become familiar to implement fast, efficient, interactive and scalable web applications using run time environment provided by the full stack components.

Course Outcomes:

- Understand Full stack components for developing web application.
- Apply packages of NodeJS to work with Data, Files, Http Requests and Responses.
- Use MongoDB data base for storing and processing huge data and connects with NodeJS application.
- Design faster and effective single page applications using Express and Angular.
- Create interactive user interfaces with react components.

UNIT-I**Introduction to Full Stack Development:**

Understanding the Basic Web Development Framework- User, Browser, Webserver, Backend Services, Full Stack Components - Node.js, MongoDB, Express, React, Angular. Java Script Fundamentals, NodeJS- Understanding Node.js, Installing Node.js, Working with Node Packages, creating a Node.js Application, Understanding the Node.js Event Model, Adding Work to the Event Queue, Implementing Callbacks

UNIT-II**Node.js:**

Working with JSON, Using the Buffer Module to Buffer Data, Using the Stream Module to Stream Data, Accessing the File System from Node.js- Opening, Closing, Writing, Reading Files and other File System Tasks. Implementing HTTP Services in Node.js- Processing URLs, Processing Query Strings and Form Parameters, Understanding Request, Response, and Server Objects, Implementing HTTP Clients and Servers in Node.js, Implementing HTTPS Servers and Clients. Using Additional Node.js Modules-Using the os Module, Using the util Module, Using the dns Module, Using the crypto Module.

UNIT-III**MongoDB:**

Need of NoSQL, Understanding MongoDB, MongoDB Data Types, Planning Your Data Model, Building the MongoDB Environment, Administering User Accounts, Configuring Access Control, Administering Databases, Managing Collections, Adding the MongoDB Driver to Node.js, Connecting to MongoDB from Node.js, Understanding the Objects Used in the MongoDB Node.js Driver, Accessing and Manipulating Databases, Accessing and Manipulating Collections

UNIT-IV**Express and Angular:**

Getting Started with Express, Configuring Routes, Using Requests Objects, Using Response Objects. Angular: importance of Angular, Understanding Angular, creating a Basic Angular Application, Angular Components, Expressions, Data Binding, Built-in Directives, Custom Directives, Implementing Angular Services in Web Applications.

UNIT-V

React:

Need of React, Simple React Structure, The Virtual DOM, React Components, Introducing React Components, Creating Components in React, Data and Data Flow in React, Rendering and Life Cycle Methods in React, Working with forms in React, integrating third party libraries, Routing in React.

TEXT BOOKS:

1. Brad Dayley, Brendan Dayley, Caleb Dayley., Node.js, MongoDB and Angular Web Development, 2nd Edition, Addison-Wesley, 2019.
2. Mark Tielens Thomas, React in Action, 1st Edition, Manning Publications.

REFERENCE BOOKS:

1. Vasan Subramanian, Pro MERN Stack, Full Stack Web App Development with Mongo, Express, React, and Node, 2nd Edition, Apress, 2019.
2. Chris Northwood, 'The Full Stack Developer: Your Essential Guide to the Everyday Skills Expected of a Modern Full Stack Web Developer', 1st edition, Apress, 2018.
3. Kirupa Chinnathambi, Learning React: A Hands-On Guide to Building Web Applications Using React and Redux, 2nd edition, Addison-Wesley Professional, 2018.

22IT722OE: SCRIPTING LANGUAGES (Open Elective – II)**B.Tech. IV Year I Sem.****L T P C**
3 0 0 3**Prerequisites:**

1. A course on “Computer Programming and Data Structures”.
2. A course on “Object Oriented Programming Concepts”.

Course Objectives:

- This course introduces the script programming paradigm
- Introduces scripting languages such as Perl, Ruby and TCL.
- Learning TCL

Course Outcomes:

- Comprehend the differences between typical scripting languages and typical system and application programming languages.
- Gain knowledge of the strengths and weakness of Perl, TCL and Ruby; and select an appropriate language for solving a given problem.
- Acquire programming skills in scripting language

UNIT - I

Introduction: Ruby, Rails, The structure and Execution of Ruby Programs, Package Management with RUBYGEMS, Ruby and web: Writing CGI scripts, cookies, Choice of Webservers, SOAP and web services

RubyTk - Simple Tk Application, widgets, Binding events, Canvas, scrolling

UNIT - II

Extending Ruby: Ruby Objects in C, the Jukebox extension, Memory allocation, Ruby Type System, Embedding Ruby to Other Languages, Embedding a Ruby Interpreter

UNIT - III**Introduction to PERL and Scripting**

Scripts and Programs, Origin of Scripting, Scripting Today, Characteristics of Scripting Languages, Uses for Scripting Languages, Web Scripting, and the universe of Scripting Languages. PERL- Names and Values, Variables, Scalar Expressions, Control Structures, arrays, list, hashes, strings, pattern and regular expressions, subroutines.

UNIT - IV**Advanced perl**

Finer points of looping, pack and unpack, filesystem, eval, data structures, packages, modules, objects, interfacing to the operating system, Creating Internet ware applications, Dirty Hands Internet Programming, security Issues.

UNIT - V**TCL**

TCL Structure, syntax, Variables and Data in TCL, Control Flow, Data Structures, input/output, procedures, strings, patterns, files, Advance TCL- eval, source, exec and uplevel commands, Name spaces, trapping errors, event driven programs, making applications internet aware, Nuts and Bolts Internet Programming, Security Issues, C Interface.

Tk

Tk-Visual Tool Kits, Fundamental Concepts of Tk, Tk by example, Events and Binding, Perl-Tk.

TEXT BOOKS:

1. The World of Scripting Languages, David Barron, Wiley Publications.

2. Ruby Programming language by David Flanagan and Yukihiro Matsumoto O'Reilly
3. "Programming Ruby" The Pragmatic Programmers guide by Dabve Thomas Second edition

REFERENCE BOOKS:

1. Open Source Web Development with LAMP using Linux Apache, MySQL, Perl and PHP, J.Lee and B. Ware (Addison Wesley) Pearson Education.
2. Perl by Example, E. Quigley, Pearson Education.
3. Programming Perl, Larry Wall, T. Christiansen and J. Orwant, O'Reilly, SPD.
4. Tcl and the Tk Tool kit, Ousterhout, Pearson Education.
5. Perl Power, J. P. Flynt, Cengage Learning.

22IT703PC: INFORMATION SECURITY LAB**B.Tech. IV Year I Sem.****L T P C**
0 0 2 1**Course Objectives**

- To understand the fundamentals of Cryptography
- To understand various key distribution and management schemes
- To understand how to deploy encryption techniques to secure data in transit across data networks
- To apply algorithms used for secure transactions in real world applications

Course Outcomes

- Demonstrate the knowledge of cryptography, network security concepts and applications.
- Ability to apply security principles in system design.
- Ability to identify and investigate vulnerabilities and security threats and mechanisms to counter them.

List of Experiments:

1. Implementation of symmetric cipher algorithm (AES and RC4)
2. Random number generation using a subset of digits and alphabets.
3. Implementation of RSA based signature system
4. Implementation of Subset sum
5. Authenticating the given signature using the MD5 hash algorithm.
6. Implementation of Diffie-Hellman algorithm
7. Implementation of the ELGAMAL cryptosystem.
8. Implementation of Goldwasser-Micali probabilistic public key system
9. Implementation of Rabin Cryptosystem. (Optional).
10. Implementation of Kerberos cryptosystem
11. Implementation of a trusted secure web transaction.
12. Digital Certificates and Hybrid (ASSY/SY) encryption, PKI.
13. Message Authentication Codes.
14. Elliptic Curve cryptosystems (Optional)

TEXT BOOK:

1. Cryptography and Network Security (principles and approaches) by William Stallings Pearson Education, 4th Edition.

REFERENCE BOOKS:

1. Network Security Essentials (Applications and Standards) by William Stallings Pearson Education.
2. Principles of Information Security, Whitman, Thomson.

22IT704PC: CLOUD COMPUTING LAB**B.Tech. IV Year I Sem.****L T P C**
0 0 2 1**Course Objectives:**

- This course provides an insight into cloud computing
- Topics covered include- distributed system models, different cloud service models, service oriented architectures, cloud programming and software environments, resource management.

Course Outcomes:

- Understand various service types, delivery models and technologies of a cloud computing environment.
- Understand the ways in which the cloud can be programmed and deployed.
- Understand cloud service providers like Cloudsim, Globus Toolkit etc.
- Examine various programming paradigms suitable to solve real world and scientific problems using cloud services.

List of Experiments:

1. Install Virtualbox/VMware Workstation with different flavors of Linux or windows OS on top of windows7 or 8.
2. Install a C compiler in the virtual machine created using virtual box and execute Simple Programs
3. Create an Amazon EC2 instance and set up a web-server on the instance and associate an IP address with the instance.
4. Install Google App Engine. Create a hello world app and other simple web applications using python/java.
5. Simulate a cloud scenario using CloudSim and run a scheduling algorithm that is not present in CloudSim.
6. Find a procedure to transfer the files from one virtual machine to another virtual machine.
7. Find a procedure to launch virtual machine using trystack (Online Openstack Demo Version)
8. Install Hadoop single node cluster and run simple applications like word count.
9. Create a database instance in the cloud using Amazon RDS.
10. Create a database instance in the cloud using Google Cloud SQL

TEXT BOOK:

1. Essentials of cloud Computing: K. Chandrasekhran, CRC press, 2014

REFERENCE BOOKS:

1. Cloud Computing: Principles and Paradigms by Rajkumar Buyya, James Broberg and Andrzej M. Goscinski, Wiley, 2011.
2. Distributed and Cloud Computing, Kai Hwang, Geoffery C. Fox, Jack J. Dongarra, Elsevier, 2012.
3. Cloud Computing Bible, Barrie Sosinsky, Wiley-India, 2010

22SM801MS: ORGANIZATIONAL BEHAVIOUR**B.Tech. IV Year II Sem.**

L	T	P	C
3	0	0	3

Course Objectives:

- This course demonstrates individual, group behavior aspects: The dynamics of organizational climate, structure and its impact on Organizations.

Course Outcomes:

- Students understand their personality, perception and attitudes for overall development and further learn the importance of group behavior in the organizations.

UNIT - I Organizational Behaviour

Definition, need and importance of organizational behaviour - Nature and scope - Frame work - Organizational behaviour models.

UNIT - II Individual Behaviour

Personality - types - Factors influencing personality - Theories - Learning - Types of learners - The learning process - Learning theories - Organizational behaviour modification, Misbehaviour - Types - Management Intervention. Emotions - Emotional Labour - Emotional Intelligence - Theories. Attitudes - Characteristics - Components - Formation - Measurement- Values. Perceptions - Importance - Factors influencing perception - Interpersonal perception- Impression Management. Motivation - importance - Types - Effects on work behavior.

UNIT - III Group Behaviour

Organization structure - Formation - Groups in organizations - Influence - Group dynamics - Emergence of informal leaders and working norms - Group decision making techniques - Team building - Interpersonal relations - Communication - Control.

UNIT - IV Leadership and Power

Meaning - Importance - Leadership styles - Theories of leadership - Leaders Vs Managers - Sources of power - Power centers - Power and Politics.

UNIT - V Dynamics of Organizational Behaviour

Organizational culture and climate - Factors affecting organizational climate - Importance. Job satisfaction - Determinants - Measurements - Influence on behavior. Organizational change - Importance - Stability Vs Change - Proactive Vs Reaction change - the change process - Resistance to change - Managing change. Stress - Work Stressors - Prevention and Management of stress - Balancing work and Life. Organizational development - Characteristics - objectives -. Organizational effectiveness

TEXT BOOKS:

1. Stephen P. Robins, Organisational Behavior, PHI Learning / Pearson Education, 11th edition, 2008.
2. Fred Luthans, Organisational Behavior, McGraw Hill, 11th Edition, 2001.

REFERENCE BOOKS:

1. Schermerhorn, Hunt and Osborn, Organisational behavior, John Wiley, 9th Edition, 2008.
2. Udai Pareek, Understanding Organisational Behaviour, 2nd Edition, Oxford Higher Education, 2004.

22IT861PE: NATURAL LANGUAGE PROCESSING (Professional Elective – VI)**B.Tech. IV Year II Sem.****L T P C**
3 0 0 3**Prerequisites:**

1. Data structures and compiler design

Course Objectives:

- Introduction to some of the problems and solutions of NLP and their relation to linguistics and statistics.

Course Outcomes:

- Show sensitivity to linguistic phenomena and an ability to model them with formal grammars.
- Understand and carry out proper experimental methodology for training and evaluating empirical NLP systems
- Manipulate probabilities, construct statistical models over strings and trees, and estimate parameters using supervised and unsupervised training methods.
- Design, implement, and analyze NLP algorithms; and design different language modeling Techniques.

UNIT - I

Finding the Structure of Words: Words and Their Components, Issues and Challenges, Morphological Models

Finding the Structure of Documents: Introduction, Methods, Complexity of the Approaches, Performances of the Approaches, Features

UNIT - II

Syntax I: Parsing Natural Language, Treebanks: A Data-Driven Approach to Syntax, Representation of Syntactic Structure, Parsing Algorithms

UNIT – III

Syntax II: Models for Ambiguity Resolution in Parsing, Multilingual Issues

Semantic Parsing I: Introduction, Semantic Interpretation, System Paradigms, Word Sense

UNIT - IV

Semantic Parsing II: Predicate-Argument Structure, Meaning Representation Systems

UNIT - V

Language Modeling: Introduction, N-Gram Models, Language Model Evaluation, Bayesian parameter estimation, Language Model Adaptation, Language Models- class based, variable length, Bayesian topic based, Multilingual and Cross Lingual Language Modeling

TEXT BOOKS:

1. Multilingual natural Language Processing Applications: From Theory to Practice - Daniel M. Bikel and Imed Zitouni, Pearson Publication.

REFERENCE BOOK:

1. Speech and Natural Language Processing - Daniel Jurafsky & James H Martin, Pearson Publications.
2. Natural Language Processing and Information Retrieval: Tanvier Siddiqui, U.S. Tiwary.

22CS862PE: DISTRIBUTED SYSTEMS**(Professional Elective –VI)****(Common To CSE & IT)****B.Tech. IV Year II Sem.****L T P C****3 0 0 3****Prerequisites:**

1. A course on "Operating Systems".
2. A course on "Computer Organization & Architecture".

Course Objectives:

- To provide an insight into Distributed systems.
- To introduce concepts related to Peer to Peer Systems, Transactions and Concurrency control, Security and Distributed shared memory.

Course Outcomes:

- Understand Transactions and Concurrency control.
- Understand distributed shared memory.
- Design a protocol for a given distributed application.

UNIT - I

Characterization of Distributed Systems: Examples of Distributed systems, Resource sharing and web, challenges

System models: Architectural and Fundamental models, Networking and Internetworking, Interprocess Communication

Distributed objects and Remote Invocation: Communication between distributed objects, RPC, Events and notifications, Case study-Java RMI.

UNIT - II

Operating System Support- OS layer, Protection, Processes and Threads, Communication and Invocation, Operating system architecture.

Distributed File Systems-Introduction, File Service architecture.

UNIT - III

Peer to Peer Systems- Napster and its legacy, Peer to Peer middleware

Time and Global States-Introduction, Clocks, events and Process states, Synchronizing physical clocks, logical time and logical clocks, global states, distributed debugging.

Coordination and Agreement- Distributed mutual exclusion, Elections, Multicast communication, consensus and related problems.

UNIT - IV

Transactions and Concurrency Control- Introduction, Transactions, Nested Transactions, Locks, Optimistic concurrency control, Timestamp ordering.

Distributed Transactions-Introduction, Flat and Nested Distributed Transactions, Atomic commit protocols, Concurrency control in distributed transactions

Distributed deadlocks: Transaction recovery.

UNIT - V

Replication: Introduction, System model and group communication, Fault tolerant services, Transactions with replicated data.

Distributed shared memory: Design and Implementation issues, Consistency models.

TEXT BOOKS:

1. Distributed Systems Concepts and Design, G Coulouris, J Dollimore and T Kindberg, Fourth Edition, Pearson Education.
2. Distributed Systems, S. Ghosh, Chapman & Hall/CRC, Taylor & Francis Group, 2010.

REFERENCE BOOKS:

1. Distributed Systems - Principles and Paradigms, A.S. Tanenbaum and M.V. Steen, Pearson Education.
2. Distributed Computing, Principles, Algorithms and Systems, Ajay D. Kshemakalyani and Mukesh Singhal, Cambridge, rp 2010.

22IT863PE: AUGMENTED REALITY & VIRTUAL REALITY
(Professional Elective –VI)

B.Tech. IV Year II Sem.

L T P C
3 0 0 3

Course Objectives:

- Provide a foundation to the fast growing field of AR and make the students aware of the various AR concepts.
- To give historical and modern overviews and perspectives on virtual reality. It describes the fundamentals of sensation, perception, technical and engineering aspects of virtual reality systems.

Course Outcomes:

- Describe how AR systems work and list the applications of AR.
- Understand the software architectures of AR.
- Understand the Visual perception and rendering in VR
- Understand the interaction, auditory perception and rendering in VR

UNIT - I

Introduction to Augmented Reality: Augmented Reality - Defining augmented reality, history of augmented reality, Examples, Related fields

Displays: Multimodal Displays, Visual Perception, Requirements and Characteristics, Spatial Display Model, Visual Displays

Tracking: Tracking, Calibration, and Registration, Coordinate Systems, Characteristics of Tracking Technology, Stationary Tracking Systems, Mobile Sensors

UNIT - II

Computer Vision for Augmented Reality: Marker Tracking, Multiple-Camera Infrared Tracking, Natural Feature Tracking by Detection, Outdoor Tracking.

Interaction: Output Modalities, Input Modalities, Tangible Interfaces, Virtual User Interfaces on Real Surfaces, Augmented Paper, Multi-view Interfaces, Haptic Interaction

Software Architectures: AR Application Requirements, Software Engineering Requirements, Distributed Object Systems, Dataflow, Scene Graphs

UNIT - III

Introduction to Virtual Reality: Defining Virtual Reality, History of VR, Human Physiology and Perception

The Geometry of Virtual Worlds: Geometric Models, Axis-Angle Representations of Rotation, Viewing Transformations

Light and Optics: Basic Behavior of Light, Lenses, Optical Aberrations, The Human Eye, Cameras, Displays

UNIT - IV

The Physiology of Human Vision: From the Cornea to Photoreceptors, From Photoreceptors to the Visual Cortex, Eye Movements, Implications for VR

Visual Perception: Visual Perception - Perception of Depth, Perception of Motion, Perception of Color

Visual Rendering: Visual Rendering -Ray Tracing and Shading Models, Rasterization, Correcting Optical Distortions, Improving Latency and Frame Rates, Immersive Photos and Videos

UNIT - V

Motion in Real and Virtual Worlds: Velocities and Accelerations, The Vestibular System, Physics in the Virtual World, Mismatched Motion and Vection

Interaction: Motor Programs and Remapping, Locomotion, Social Interaction

Audio: The Physics of Sound, The Physiology of Human Hearing, Auditory Perception, Auditory Rendering

TEXT BOOKS:

1. Augmented Reality: Principles & Practice by Schmalstieg / Hollerer, Pearson Education India; First edition (12 October 2016), ISBN-10: 9332578494
2. Virtual Reality, Steven M. LaValle, Cambridge University Press, 2016

REFERENCE BOOKS:

1. Allan Fowler-AR Game Development], 1st Edition, A press Publications, 2018, ISBN 978-1484236178
2. Understanding Virtual Reality: Interface, Application and Design, William R Sherman and Alan B Craig, (The Morgan Kaufmann Series in Computer Graphics)". Morgan Kaufmann Publishers, San Francisco, CA, 2002
3. Developing Virtual Reality Applications: Foundations of Effective Design, Alan B Craig, William R Sherman and Jeffrey D Will, Morgan Kaufmann, 2009
4. Designing for Mixed Reality, Kharis O'Connell Published by O'Reilly Media, Inc., 2016, ISBN: 9781491962381
5. Sanni Siltanen- Theory and applications of marker-based augmented reality. Julkaisija – Utgivare Publisher. 2012. ISBN 978-951-38-7449-0
6. Gerard Jounghyun Kim, "Designing Virtual Systems: The Structured Approach", 2005.

22IT864PE: WEB SECURITY (Professional Elective –VI)**B.Tech. IV Year II Sem.**

L	T	P	C
3	0	0	3

Course Objectives:

1. Give an Overview of information security
2. Give an overview of Access control of relational databases

Course Outcomes: Students should be able to

1. Understand the Web architecture and applications
2. Understand client side and service side programming
3. Understand how common mistakes can be bypassed and exploit the application
4. Identify common application vulnerabilities

UNIT - I

The Web Security, The Web Security Problem, Risk Analysis and Best Practices
 Cryptography and the Web: Cryptography and Web Security, Working Cryptographic Systems and Protocols, Legal Restrictions on Cryptography, Digital Identification

UNIT - II

The Web's War on Your Privacy, Privacy-Protecting Techniques, Backups and Antitheft, Web Server Security, Physical Security for Servers, Host Security for Servers, Securing Web Applications

UNIT - III

Database Security: Recent Advances in Access Control, Access Control Models for XML, Database Issues in Trust Management and Trust Negotiation, Security in Data Warehouses and OLAP Systems

UNIT - IV

Security Re-engineering for Databases: Concepts and Techniques, Database Watermarking for Copyright Protection, Trustworthy Records Retention, Damage Quarantine and Recovery in Data Processing Systems, Hippocratic Databases: Current Capabilities and

UNIT - V

Future Trends Privacy in Database Publishing: A Bayesian Perspective, Privacy-enhanced Location Based Access Control, Efficiently Enforcing the Security and Privacy Policies in a Mobile Environment

TEXT BOOKS:

1. Web Security, Privacy and Commerce Simson G Arfinkel, Gene Spafford, O'Reilly.
2. Handbook on Database security applications and trends Michael Gertz, Sushil Jajodia

22CS865PE: CYBER FORENSICS (Professional Elective –VI)
(Common To CSE & IT)

B.Tech. IV Year II Sem.

L	T	P	C
3	0	0	3

Prerequisites: Network Security.

Course Objectives:

- A brief explanation of the objective is to provide digital evidence which is obtained from digital media.
- In order to understand the objectives of computer forensics, first of all, people have to recognize the different roles computers play in a certain crime.
- According to a snippet from the United States Security Service, the computer functions in different kinds of crimes.

Course Outcomes:

- Students will understand the usage of computers in forensic, and how to use various forensic tools for a wide variety of investigations.
- It gives an opportunity to students to continue their zeal in research in computer forensics

UNIT- I

Introduction of Cybercrime: Types, The Internet spawns crime, Worms versus viruses, Computers' roles in crimes, Introduction to digital forensics, Introduction to Incident - Incident Response Methodology – Steps - Activities in Initial Response, Phase after detection of an incident

UNIT- II

Initial Response and forensic duplication, Initial Response & Volatile Data Collection from Windows system -Initial Response & Volatile Data Collection from Unix system – Forensic Duplication: Forensic duplication: Forensic Duplicates as Admissible Evidence, Forensic Duplication Tool Requirements, Creating a Forensic. Duplicate/Qualified Forensic Duplicate of a Hard Drive

UNIT- III

Forensics analysis and validation: Determining what data to collect and analyze, validating forensic data, addressing data-hiding techniques, performing remote acquisitions
 Network Forensics: Network forensics overview, performing live acquisitions, developing standard procedures for network forensics, using network tools, examining the honeynet project.

UNIT- IV

Current Forensic tools: evaluating computer forensic tool needs, computer forensics software tools, computer forensics hardware tools, validating and testing forensics software
E-Mail Investigations: Exploring the role of e-mail in investigation, exploring the roles of the client and server in e-mail, investigating e-mail crimes and violations, understanding e-mail servers, using specialized e-mail forensic tools.

Cell phone and mobile device forensics: Understanding mobile device forensics, understanding acquisition procedures for cell phones and mobile devices.

UNIT- V

Working with Windows and DOS Systems: understanding file systems, exploring Microsoft File Structures, Examining NTFS disks, Understanding whole disk encryption, windows registry, Microsoft startup tasks, MS-DOS startup tasks, virtual machines.

TEXT BOOKS:

1. Kevin Mandia, Chris Prosise, "Incident Response and computer forensics", Tata McGraw Hill, 2006.
2. Computer Forensics, Computer Crime Investigation by John R. Vacca, Firewall Media, New Delhi.
3. Computer Forensics and Investigations by Nelson, Phillips Enfinger, Steuart, CENGAGE Learning

REFERENCE BOOKS:

1. Real Digital Forensics by Keith J. Jones, Richard Bejtich, Curtis W. Rose, Addison- Wesley Pearson Education
2. Forensic Compiling, A Tractitioneris Guide by Tony Sammes and Brian Jenkinson, Springer International edition.

22IT831OE: BIG DATA TECHNOLOGIES (Open Elective –III)**B.Tech. IV Year II Sem.**

L	T	P	C
3	0	0	3

Course Objectives

- The purpose of this course is to provide the students with knowledge of Big data Analytics principles and techniques.
- This course is also designed to give an exposure of the frontiers of Big data Analytics

Courses Outcomes

- Ability to explain the foundations, definitions, and challenges of Big Data and various Analytical tools.
- Ability to program using HADOOP and Map reduce, NOSQL
- Ability to understand the importance of Big Data in Social Media and Mining.

UNIT - I**Getting an Overview of Big Data**

Big Data, History of Data Management - Evolution of Big Data, Structuring Big Data, Elements of Big Data, Big Data Analytics, Careers in Big Data, Future of Big Data

Technologies for Handling Big Data

Distributed and Parallel Computing for Big Data, Introducing Hadoop, Cloud Computing and Big Data, In-Memory Computing Technology for Big Data.

UNIT - II**Understanding Hadoop Ecosystem**

Hadoop Ecosystem, Hadoop Distributed File System, MapReduce, Hadoop YARN, Hbase, Hive, Pig and Pig Latin, Sqoop, ZooKeeper, Flume, Oozie

Understanding MapReduce Fundamentals and HBase

The MapReduce Framework, Techniques to Optimize MapReduce Jobs, Uses of MapReduce, Role of HBase in Big Data Processing

UNIT - III**Exploring Hive**

Introducing Hive, Getting Started with Hive, Data Types in Hive, Built-In Functions in Hive, Hive DDL, Data Manipulation in Hive, Data Retrieval Queries, Using JOINS in Hive

Analyzing Data with Pig

Introducing Pig, Running Pig, Getting Started with Pig Latin, Working with Operators in Pig, Working with Functions in Pig

UNIT - IV**Using Oozie**

Introducing Oozie, Installing and Configuring Oozie, Understanding the Oozie Workflow, Oozie Coordinator, Oozie Bundle, Oozie Parameterization with EL, Oozie Job Execution Model, Accessing Oozie, Oozie SLA

NoSQL Data Management

Introduction to NoSQL, Aggregate Data Models, Key Value Data Model, Document Databases, Relationships, Graph Databases, Schema-Less Databases, Materialized Views, Distribution Models, Sharding, MapReduce Partitioning and Combining, Composing MapReduce Calculations

UNIT - V

ZooKeeper: Installing and Running ZooKeeper, An Example, Group Membership in ZooKeeper, Creating the Group, Joining a Group, Listing Members in a Group, The ZooKeeper Service, Data

Model, Operations, Implementation, Consistency, Sessions, Building Applications with ZooKeeper, A Configuration, Service, The Resilient ZooKeeper Application, A Lock Service, More Distributed Data Structures and Protocols, ZooKeeper in Production

Sqoop: Getting Sqoop, Sqoop Connectors, A Sample Import, Generated Code, Imports: A Deeper Look, Working with Imported Data, Importing Large Objects, Performing an Export, Exports: A Deeper Look.

TEXT BOOKS:

1. Big data, blackbook, DreamTech Press, 2015
2. Hadoop: The Definitive Guide, Tom White, 3rd Edition, O'Reilly Media, 2012.

REFERENCE BOOKS:

1. Big Data Analytics, Seema Acharya, Subhashini Chellappan, Wiley 2015.
2. Simon Walkowiak, Big Data Analytics with R, Packt Publishing, ISBN: 9781786466457
3. Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today's Business, Michael Minelli, Michehe Chambers, 1st Edition, Ambiga Dhiraj, Wiley CIO Series, 2013.
4. Big Data Analytics: Disruptive Technologies for Changing the Game, Arvind Sathi, 1st Edition, IBM Corporation, 2012.

22IT832OE: DEVOPS (Open Elective – III)**B.Tech. IV Year II Sem.**

L	T	P	C
3	0	0	3

Pre-Requisites:

1. Software Engineering
2. Software Project Management

Course Objectives:

- Understand the skill sets and high-functioning teams involved in Agile, DevOps and related methods to reach a continuous delivery capability.
- Implement automated system update and DevOps lifecycle.

Course Outcomes:

- Understand the various components of DevOps environment.
- Identify Software development models and architectures of DevOps
- Use different project management and integration tools.
- Select an appropriate testing tool and deployment model for project.

UNIT- I**Introduction to DevOps:**

Introduction, Agile development model, DevOps and ITIL. DevOps process and Continuous Delivery, Release management, Scrum, Kanban, delivery pipeline, identifying bottlenecks.

UNIT- II:**Software development models and DevOps:**

DevOps Lifecycle for Business Agility, DevOps, and Continuous Testing. DevOps influence on Architecture: Introducing software architecture, The monolithic scenario, Architecture rules of thumb, The separation of concerns, Handling database migrations, Micro services and the data tier, DevOps, architecture, and resilience.

UNIT- III**Introduction to project management:**

The need for source code control, the history of source code management, Roles and code, source code management system and migrations, shared authentication, Hosted Git servers, Different Git server implementations, Docker intermission, Gerrit, The pull request model, GitLab.

UNIT- IV**Integrating the system:**

Build systems, Jenkins build server, Managing build dependencies, Jenkins plugins, and file system layout, The host server, Build slaves, Software on the host, Triggers, Job chaining and build pipelines, Build servers and infrastructure as code, Building by dependency order, Build phases, Alternative build servers, Collating quality measures.

UNIT- V**Testing Tools and Deployment:**

Various types of testing, Automation of testing Pros and cons, Selenium - Introduction, Selenium features, JavaScript testing, Testing backend integration points, Test-driven development, REPL-driven development. Deployment of the system: Deployment systems, Virtualization stacks, code execution at the client, Puppet master and agents, Ansible, Deployment tools: Chef, SaltStack and Docker.

TEXT BOOK:

1. Joakim Verona., Practical DevOps, Packt Publishing, 2016.

REFERENCE BOOKS:

1. Deepak Gaikwad, Viral Thakkar. DevOps Tools from Practitioner's Viewpoint. Wiley publications.
2. Len Bass, Ingo Weber, Liming Zhu. DevOps: A Software Architect's Perspective. Addison Wesley.