



## SHADAN WOMEN'S COLLEGE OF ENGINEERING & TECHNOLOGY

*A UGC Autonomous Institution, Affiliated to JNTUH*

Khairatabad, Hyderabad, Telangana (India).

**B.Tech.in INFORMATION TECHNOLOGY**

**II YEAR COURSE STRUCTURE & SYLLABUS (R23 Regulations)**

**Applicable from AY 2024-25 Batch**

### II Year I Semester

S. No.	Course Code	Course	L	T	P	Credits
1	IT301PC	Digital Electronics	3	0	0	3
2	IT302PC	Data Structures	3	0	0	3
3	IT303PC	Computer Oriented Statistical Methods	3	1	0	4
4	IT304PC	Computer Organization and Microprocessor	3	0	0	3
5	IT305PC	Introduction to IoT	2	0	0	2
6	IT306PC	Digital Electronics Lab	0	0	2	1
7	IT307PC	Data Structures Lab	0	0	3	1.5
8	IT308PC	Internet of Things Lab	0	0	3	1.5
9	IT309PC	Data visualization- R Programming/ Power BI	0	0	2	1
		<b>Total</b>	<b>14</b>	<b>1</b>	<b>12</b>	<b>20</b>

### II Year II Semester

S. No.	Course Code	Course	L	T	P	Credits
1	IT401PC	Discrete Mathematics	3	0	0	3
2	SM402MS	Business Economics & Financial Analysis	3	0	0	3
3	IT403PC	Operating Systems	3	0	0	3
4	IT404PC	Database Management Systems	3	0	0	3
5	IT405PC	Java Programming	2	0	0	2
6	IT406PC	Operating Systems Lab	0	0	2	1
7	IT407PC	Database Management Systems Lab	0	0	2	1
8	IT408PC	Java Programming Lab	0	0	2	1
9	IT409PC	Real-time Research Project/ Societal Related Project	0	0	4	2
10	IT410PC	Node JS/ React JS/ Django	0	0	2	1
	MC410	Constitution Of India				
		<b>Total</b>	<b>17</b>	<b>0</b>	<b>12</b>	<b>17</b>

**CS301PC : DIGITAL ELECTRONICS****B.Tech. II Year I Sem.**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>3</b>	<b>0</b>	<b>0</b>	<b>3</b>

**Course Objectives:** This course aims at through understanding of binary number system, logic gates, combination logic and synchronous and asynchronous logic.

**UNIT - I:**

**BOOLEAN ALGEBRA AND LOGIC GATES:** Digital Systems, Binary Numbers, Number base conversions, Octal and Hexadecimal Numbers, complements, Signed binary numbers, Binary codes, Binary Storage and Registers, Binary logic.

Basic Definitions, Axiomatic definition of Boolean Algebra, Basic theorems and properties of Boolean algebra, Boolean functions, canonical and standard forms, other logic operations, Digital logic gates.

**UNIT - II:**

**GATE – LEVEL MINIMIZATION:** The map method, Four-variable map, Five-Variable map, product of sums simplification Don't-care conditions, NAND and NOR implementation other Two-level implementations, Exclusive – Or function.

**UNIT - III:**

**COMBINATIONAL LOGIC:** Combinational Circuits, Analysis procedure Design procedure, Binary Adder-Subtractor Decimal Adder, Binary multiplier, magnitude comparator, Decoders, Encoders, Multiplexers, HDL for combinational circuits.

**UNIT - IV:**

**SEQUENTIAL LOGIC:** Sequential circuits, latches, Flip-Flops Analysis of clocked sequential circuits, state Reduction and Assignment, Design Procedure. Registers, shift Registers, Ripple counters, synchronous counters, other counters.

**UNIT - V**

**MEMORIES AND ASYNCHRONOUS SEQUENTIAL LOGIC:** Introduction, Random-Access Memory, Memory Decoding, Error Detection and correction Read-only memory, Programmable logic Array programmable Array logic, Sequential Programmable Devices.

Introduction, Analysis Procedure, Circuits with Latches, Design Procedure, Reduction of state and FlowTables, Race-Free state Assignment Hazards, Design Example.

**TEXT BOOKS:**

1. Digital Design – Third Edition, M. Morris Mano, Pearson Education/PHI.
2. Digital Principles and Applications Albert Paul Malvino Donald P. Leach TATA McGraw Hill Edition.
3. Fundamentals of Logic Design, Roth, 5<sup>th</sup> Edition, Thomson.

**REFERENCE BOOKS:**

1. Switching and Finite Automata Theory by Zvi. Kohavi, Tata McGraw Hill.
2. Switching and Logic Design, C.V.S. Rao, Pearson Education
3. Digital Principles and Design – Donald D.Givone, Tata McGraw Hill, Edition.
4. Fundamentals of Digital Logic and Microcomputer Design, 5<sup>TH</sup> Edition, M. Rafiquzzaman JohnWiley.

**CS302PC: DATA STRUCTURES****B.Tech. II Year I Sem.****L T P C**  
**3 0 0 3****Prerequisites:** Programming for Problem Solving**Course Objectives**

- Exploring basic data structures such as stacks and queues.
- Introduces a variety of data structures such as hash tables, search trees, tries, heaps, graphs.
- Introduces sorting and pattern matching algorithms

**Course Outcomes**

- Ability to select the data structures that efficiently model the information in a problem.
- Ability to assess efficiency trade-offs among different data structure implementations or combinations.
- Implement and know the application of algorithms for sorting and pattern matching.
- Design programs using a variety of data structures, including hash tables, binary and general tree structures, search trees, tries, heaps, graphs, and AVL-trees.

**UNIT - I**

Introduction to Data Structures, abstract data types, Linear list – singly linked list implementation, insertion, deletion and searching operations on linear list, Stacks- Operations, array and linked representations of stacks, stack applications, Queues- operations, array and linked representations.

**UNIT - II**

Dictionaries: linear list representation, skip list representation, operations - insertion, deletion and searching.

Hash Table Representation: hash functions, collision resolution-separate chaining, open addressing- linear probing, quadratic probing, double hashing, rehashing, extendible hashing.

**UNIT - III**

Search Trees: Binary Search Trees, Definition, Implementation, Operations- Searching, Insertion and Deletion, B- Trees, B+ Trees, AVL Trees, Definition, Height of an AVL Tree, Operations – Insertion, Deletion and Searching, Red –Black, Splay Trees.

**UNIT - IV**

Graphs: Graph Implementation Methods. Graph Traversal Methods.

Sorting: Quick Sort, Heap Sort, External Sorting- Model for external sorting, Merge Sort.

**UNIT - V**

Pattern Matching and Tries: Pattern matching algorithms-Brute force, the Boyer –Moore algorithm, the Knuth-Morris-Pratt algorithm, Standard Tries, Compressed Tries, Suffix tries.

**TEXT BOOKS:**

1. Fundamentals of Data Structures in C, 2 nd Edition, E. Horowitz, S. Sahni and Susan Anderson-Freed, Universities Press.
2. Data Structures using C – A. S.Tanenbaum, Y. Langsam, and M.J. Augenstein, PHI/Pearson Education.

**REFERENCE BOOK:**

1. Data Structures: A Pseudocode Approach with C, 2 nd Edition, R. F. Gilberg and B.A.Forouzan, Cengage Learning.

**CS303PC: COMPUTER ORIENTED STATISTICAL METHODS****B.Tech. II Year I Sem.**

L	T	P	C
3	1	0	4

**Pre-requisites:** Mathematics courses of first year of study.**Course Objectives:** To learn

- The theory of Probability, Probability distributions of single and multiple random variables
- The sampling theory, testing of hypothesis and making statistical inferences
- Stochastic process and Markov chains.

**Course outcomes:** After learning the contents of this paper the student must be able to

- Apply the concepts of probability and distributions to case studies.
- Formulate and solve problems involving random variables and apply statistical methods for analyzing experimental data.
- Apply concept of estimation and testing of hypothesis to case studies.
- Correlate the concepts of one unit to the concepts in other units.

**UNIT - I: Probability****10 L**

Sample Space, Events, Counting Sample Points, Probability of an Event, Additive Rules, Conditional Probability, Independence, and the Product Rule, Baye's Rule,

**Random Variables and Probability Distributions:** Concept of a Random Variable, Discrete Probability Distributions, Continuous Probability Distributions.

**UNIT - II: Expectation and discrete distributions****10 L**

Mean of a Random Variable, Variance and Covariance of Random Variables, Means and Variances of Linear Combinations of Random Variables, Chebyshev's Theorem.

**Discrete Probability Distributions:** Binomial Distribution, Poisson distribution.

**UNIT - III: Continuous and Sampling Distributions****10 L**

Uniform Distribution, Normal Distribution, Areas under the Normal Curve, Applications of the Normal Distribution, Normal Approximation to the Binomial Distributions.

**Fundamental Sampling Distributions:** Random Sampling, Some Important Statistics, Sampling Distributions, Sampling Distribution of Means and the Central Limit Theorem, t - Distribution, F-Distribution.

**UNIT - IV: Sample Estimation & Tests of Hypotheses****10 L**

Introduction, Statistical Inference, Classical Methods of Estimation, Single Sample: Estimating the mean, standard error of a point estimate, prediction interval. Two sample: Estimating the difference between two means, Single sample: Estimating a proportion, Two samples: Estimating the difference between two proportions, Two samples: Estimating the ratio of two variances.

Statistical Hypotheses: General Concepts, Testing a Statistical Hypothesis, Single sample: Tests concerning a single mean, Two samples: tests on two means, One sample: test on a single proportion. Two samples: tests on two proportions, Two- sample tests concerning variances.

**UNIT-V: Stochastic Processes and Markov Chains****8L**

Introduction to Stochastic processes- Markov process. Transition Probability, Transition Probability Matrix, First order and Higher order Markov process, n-step transition probabilities, Markov chain, Steady state condition, Markov analysis.

**TEXT BOOKS:**

1. Ronald E. Walpole, Raymond H. Myers, Sharon L. Myers, Keying Ye, Probability & Statistics For Engineers & Scientists, 9<sup>th</sup> Ed. Pearson Publishers.

2. S C Gupta and V K Kapoor, Fundamentals of Mathematical statistics, Khanna publications.
3. S.D.Sharma, Operations Research, Kedarnath and Ramnath Publishers, Meerut, Delhi.

**REFERENCE BOOKS:**

1. T.T. Soong, Fundamentals of Probability and Statistics For Engineers, John Wiley & Sons, Ltd,2004.
2. Sheldon M Ross, Probability and statistics for Engineers and scientists, academic press.
3. Miller and Freund's, Probability and Statistics for Engineers, 8<sup>th</sup> Edition, Pearson Educations.

**CS304PC: COMPUTER ORGANIZATION AND ARCHITECTURE****B.Tech. II Year I Sem.**

L	T	P	C
3	0	0	3

**Co-requisite:** A Course on “Digital Electronics”.**Course Objectives**

- The purpose of the course is to introduce principles of computer organization and the basic architectural concepts.
- It begins with basic organization, design, and programming of a simple digital computer and introduces simple register transfer language to specify various computer operations.
- Topics include computer arithmetic, instruction set design, microprogrammed control unit, pipelining and vector processing, memory organization and I/O systems, and multiprocessors

**Course Outcomes**

- Understand the basics of instruction sets and their impact on processor design.
- Demonstrate an understanding of the design of the functional units of a digital computer system.
- Evaluate cost performance and design trade-offs in designing and constructing a computer processor including memory.
- Design a pipeline for consistent execution of instructions with minimum hazards.
- Recognize and manipulate representations of numbers stored in digital computers

**UNIT - I**

Digital Computers: Introduction, Block diagram of Digital Computer, Definition of Computer Organization, Computer Design and Computer Architecture.

Register Transfer Language and Micro operations: Register Transfer language, Register Transfer, Bus and memory transfers, Arithmetic Micro operations, logic micro operations, shift micro operations, Arithmetic logic shift unit.

Basic Computer Organization and Design: Instruction codes, Computer Registers Computer instructions, Timing and Control, Instruction cycle, Memory Reference Instructions, Input – Output and Interrupt.

**UNIT - II**

Microprogrammed Control: Control memory, Address sequencing, micro program example, design of control unit.

Central Processing Unit: General Register Organization, Instruction Formats, Addressing modes, Data Transfer and Manipulation, Program Control.

**UNIT - III**

Data Representation: Data types, Complements, Fixed Point Representation, Floating Point Representation.

Computer Arithmetic: Addition and subtraction, multiplication Algorithms, Division Algorithms, Floating – point Arithmetic operations. Decimal Arithmetic unit, Decimal Arithmetic operations.

**UNIT - IV**

Input-Output Organization: Input-Output Interface, Asynchronous data transfer, Modes of Transfer, Priority Interrupt Direct memory Access.

Memory Organization: Memory Hierarchy, Main Memory, Auxiliary memory, Associate Memory, Cache Memory.

**UNIT - V**

Reduced Instruction Set Computer: CISC Characteristics, RISC Characteristics.

Pipeline and Vector Processing: Parallel Processing, Pipelining, Arithmetic Pipeline, Instruction Pipeline, RISC Pipeline, Vector Processing, Array Processor.  
Multi Processors: Characteristics of Multiprocessors, Interconnection Structures, Interprocessor arbitration, Interprocessor communication and synchronization, Cache Coherence.

**TEXT BOOK:**

1. Computer System Architecture – M. Morris Mano, Third Edition, Pearson/PHI.

**REFERENCE BOOKS:**

1. Computer Organization – Carl Hamacher, Zvonks Vranesic, SafeaZaky, V th Edition, McGrawHill.
2. Computer Organization and Architecture – William Stallings Sixth Edition, Pearson/PHI.
3. Structured Computer Organization – Andrew S. Tanenbaum, 4 th Edition, PHI/Pearson.

**CS305PC: OBJECT ORIENTED PROGRAMMING THROUGH JAVA****B.Tech. II Year I Sem.**

L	T	P	C
3	0	0	3

**Course Objectives**

- To Understand the basic object-oriented programming concepts and apply them in problem solving.
- To Illustrate inheritance concepts for reusing the program.
- To Demonstrate multitasking by using multiple threads and event handling
- To Develop data-centric applications using JDBC.
- To Understand the basics of java console and GUI based programming

**Course Outcomes**

- Demonstrate the behavior of programs involving the basic programming constructs like control structures, constructors, string handling and garbage collection.
- Demonstrate the implementation of inheritance (multilevel, hierarchical and multiple) by using extend and implement keywords
- Use multithreading concepts to develop inter process communication.
- Understand the process of graphical user interface design and implementation using AWT or swings.
- Develop applets that interact abundantly with the client environment and deploy on the server.

**UNIT - I**

Object oriented thinking and Java Basics- Need for oop paradigm, summary of oop concepts, coping with complexity, abstraction mechanisms. A way of viewing world – Agents, responsibility, messages, methods, History of Java, Java buzzwords, data types, variables, scope and lifetime of variables, arrays, operators, expressions, control statements, type conversion and casting, simple java program, concepts of classes, objects, constructors, methods, access control, this keyword, garbage collection, overloading methods and constructors, method binding, inheritance, overriding and exceptions, parameter passing, recursion, nested and inner classes, exploring string class.

**UNIT - II**

Inheritance, Packages and Interfaces – Hierarchical abstractions, Base class object, subclass, subtype, substitutability, forms of inheritance specialization, specification, construction, extension, limitation, combination, benefits of inheritance, costs of inheritance. Member access rules, super uses, using final with inheritance, polymorphism- method overriding, abstract classes, the Object class. Defining, Creating and Accessing a Package, Understanding CLASSPATH, importing packages, differences between classes and interfaces, defining an interface, implementing interface, applying interfaces, variables in interface and extending interfaces. Exploring java.io.

**UNIT - III**

Exception handling and Multithreading-- Concepts of exception handling, benefits of exception handling, Termination or resumptive models, exception hierarchy, usage of try, catch, throw, throws and finally, built in exceptions, creating own exception subclasses. String handling, Exploring java.util. Differences between multithreading and multitasking, thread life cycle, creating threads, thread priorities, synchronizing threads, inter thread communication, thread groups, daemon threads. Enumerations, autoboxing, annotations, generics.

**UNIT - IV**

Event Handling: Events, Event sources, Event classes, Event Listeners, Delegation event model, handling mouse and keyboard events, Adapter classes. The AWT class hierarchy, user interface components- labels, button, canvas, scrollbars, text components, check box, checkbox groups, choices,



lists panels – scrollpane, dialogs, menubar, graphics, layout manager – layout manager types – border,grid, flow, card and grid bag.

#### **UNIT - V**

Applets – Concepts of Applets, differences between applets and applications, life cycle of an applet, types of applets, creating applets, passing parameters to applets. Swing – Introduction, limitations of AWT, MVC architecture, components, containers, exploring swing- JApplet, JFrame and JComponent, Icons and Labels, text fields, buttons – The JButton class, Check boxes, Radio buttons, Combo boxes, Tabbed Panes, Scroll Panes, Trees, and Tables.

#### **TEXT BOOKS:**

1. Java the complete reference, 7th edition, Herbert schildt, TMH.
2. Understanding OOP 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. An Introduction to OOP, third edition, T. Budd, Pearson education.
3. Introduction to Java programming, Y. Daniel Liang, Pearson education.
4. An introduction to Java programming and object-oriented application development, R.A. Johnson- Thomson.
5. Core Java 2, Vol 1, Fundamentals, Cay.S. Horstmann and Gary Cornell, eighth Edition, Pearson Education.
6. Core Java 2, Vol 2, Advanced Features, Cay.S. Horstmann and Gary Cornell, eighth Edition, Pearson Education
7. Object Oriented Programming with Java, R.Buyya, S.T.Selvi, X.Chu, TMH.
8. Java and Object Orientation, an introduction, John Hunt, second edition, Springer. 9. Maurach's Beginning Java2 JDK 5, SPD.

**CS306PC: DATA STRUCTURES LAB****B.Tech. II Year I Sem.**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>0</b>	<b>0</b>	<b>3</b>	<b>1.5</b>

**Prerequisites:** A Course on “Programming for problem solving”.

**Course Objectives:**

- It covers various concepts of C programming language
- It introduces searching and sorting algorithms
- It provides an understanding of data structures such as stacks and queues.

**Course Outcomes:**

- Ability to develop C programs for computing and real-life applications using basic elements like control statements, arrays, functions, pointers and strings, and data structures like stacks, queues and linked lists.
- Ability to Implement searching and sorting algorithms

**List of Experiments:**

1. Write a program that uses functions to perform the following operations on singly linkedlist.:
  - i) Creation
  - ii) Insertion
  - iii) Deletion
  - iv) Traversal
2. Write a program that uses functions to perform the following operations on doubly linkedlist.:
  - i) Creation
  - ii) Insertion
  - iii) Deletion
  - iv) Traversal
3. Write a program that uses functions to perform the following operations on circular linkedlist.:
  - i) Creation
  - ii) Insertion
  - iii) Deletion
  - iv) Traversal
4. Write a program that implement stack (its operations) using
  - i) Arrays
  - ii) Pointers
5. Write a program that implement Queue (its operations) using
  - i) Arrays
  - ii) Pointers
6. Write a program that implements the following sorting methods to sort a given list of integers in ascending order
  - i) Quick sort
  - ii) Heap sort
  - iii) Merge sort
7. Write a program to implement the tree traversal methods( Recursive and Non Recursive).
8. Write a program to implement
  - i) Binary Search tree
  - ii) B Trees
  - iii) B+ Trees
  - iv) AVLtrees
  - v) Red - Black trees
9. Write a program to implement the graph traversal methods.
10. Implement a Pattern matching algorithms using Boyer- Moore, Knuth-Morris-Pratt

**TEXT BOOKS:**

1. Fundamentals of Data Structures in C, 2<sup>nd</sup> Edition, E. Horowitz, S. Sahni and Susan AndersonFreed, Universities Press.
2. Data Structures using C – A. S. Tanenbaum, Y. Langsam, and M. J. Augenstein, PHI/PearsonEducation.

**REFERENCE BOOK:**

1. Data Structures: A Pseudocode Approach with C, 2<sup>nd</sup> Edition, R. F. Gilberg and B. A. Fourezan, Cengage Learning.

**CS307PC: OBJECT ORIENTED PROGRAMMING THROUGH JAVA LAB****B.Tech. II Year I Sem.**

L	T	P	C
0	0	3	1.5

**Course Objectives:**

- To write programs using abstract classes.
- To write programs for solving real world problems using the java collection framework.
- To write multithreaded programs.
- To write GUI programs using swing controls in Java.
- To introduce java compiler and eclipse platform.
- To impart hands-on experience with java programming.

**Course Outcomes:**

- Able to write programs for solving real world problems using the java collection framework.
- Able to write programs using abstract classes.
- Able to write multithreaded programs.
- Able to write GUI programs using swing controls in Java.

**Note:**

1. Use LINUX and MySQL for the Lab Experiments. Though not mandatory, encourage the use of the Eclipse platform.
2. The list suggests the minimum program set. Hence, the concerned staff is requested to add more problems to the list as needed.

**List of Experiments:**

1. Use Eclipse or Net bean platform and acquaint yourself with the various menus. Create a test project, add a test class, and run it. See how you can use auto suggestions, auto fill. Try code formatter and code refactoring like renaming variables, methods, and classes. Try debug step by step with a small program of about 10 to 15 lines which contains at least one if else condition and a for loop.
2. Write a Java program that works as a simple calculator. Use a grid layout to arrange buttons for the digits and for the +, -, \*, % operations. Add a text field to display the result. Handle any possible exceptions like divided by zero.
3. A) Develop an applet in Java that displays a simple message.  
B) Develop an applet in Java that receives an integer in one text field, and computes its factorial Value and returns it in another text field, when the button named "Compute" is clicked.
4. Write a Java program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields, Num1 and Num2. The division of Num1 and Num 2 is displayed in the Result field when the Divide button is clicked. If Num1 or Num2 were not an integer, the program would throw a Number Format Exception. If Num2 were Zero, the program would throw an Arithmetic Exception. Display the exception in a message dialog box.
5. Write a Java program that implements a multi-thread application that has three threads. First thread generates a random integer every 1 second and if the value is even, the second thread computes the square of the number and prints. If the value is odd, the third thread will print the value of the cube of the number.
6. Write a Java program for the following:  
Create a doubly linked list of elements.

Delete a given element from the above list. Display the contents of the list after deletion.

7. Write a Java program that simulates a traffic light. The program lets the user select one of three lights: red, yellow, or green with radio buttons. On selecting a button, an appropriate message with "Stop" or "Ready" or "Go" should appear above the buttons in the selected color. Initially, there is no message shown.

8. Write a Java program to create an abstract class named Shape that contains two integers and an empty method named print Area (). Provide three classes named Rectangle, Triangle, and Circle such that each one of the classes extends the class Shape. Each one of the classes contains only the method print Area () that prints the area of the given shape.

9. Suppose that a table named Table.txt is stored in a text file. The first line in the file is the header, and the remaining lines correspond to rows in the table. The elements are separated by commas. Write a java program to display the table using Labels in Grid Layout.

10. Write a Java program that handles all mouse events and shows the event name at the center of the window when a mouse event is fired (Use Adapter classes).

11. Write a Java program that loads names and phone numbers from a text file where the data is organized as one line per record and each field in a record are separated by a tab (\t). It takes a name or phone number as input and prints the corresponding other value from the hash table (hint: use hash tables).

12. Write a Java program that correctly implements the producer – consumer problem using the concept of inter thread communication.

13. Write a Java program to list all the files in a directory including the files present in all its subdirectories.

#### **REFERENCE BOOKS:**

1. Java for Programmers, P. J. Deitel and H. M. Deitel, 10th Edition Pearson education.
2. Thinking in Java, Bruce Eckel, Pearson Education.
3. Java Programming, D. S. Malik and P. S. Nair, Cengage Learning.
4. Core Java, Volume 1, 9th edition, Cay S. Horstmann and G Cornell, Pearson.

**CS308PC: SKILL DEVELOPMENT COURSE (DATA VISUALIZATION - R PROGRAMMING/ POWER BI)****B.Tech. II Year I Sem.**

L	T	P	C
0	0	2	1

**Course Objectives:**

- Effective use of Business Intelligence (BI) technology (Tableau) to apply data visualization
- To discern patterns and relationships in the data.
- To build Dashboard applications.
- To communicate the results clearly and concisely.
- To be able to work with different formats of data sets.

**Course Outcomes:** At the end of the course a student should be able to

- Understand How to import data into Tableau.
- Understand Tableau concepts of Dimensions and Measures.
- Develop Programs and understand how to map Visual Layouts and Graphical Properties.
- Create a Dashboard that links multiple visualizations.
- Use graphical user interfaces to create Frames for providing solutions to real world problems.

**Lab Problems:**

1. Understanding Data, What is data, where to find data, Foundations for building Data Visualizations, Creating Your First visualization?
2. Getting started with Tableau Software using Data file formats, connecting your Data to Tableau, creating basic charts(line, bar charts, Tree maps), Using the Show me panel.
3. Tableau Calculations, Overview of SUM, AVR, and Aggregate features, Creating custom calculations and fields.
4. Applying new data calculations to your visualizations, Formatting Visualizations, Formatting Tools and Menus, Formatting specific parts of the view.
5. Editing and Formatting Axes, Manipulating Data in Tableau data, Pivoting Tableau data.
6. Structuring your data, Sorting and filtering Tableau data, Pivoting Tableau data.
7. Advanced Visualization Tools: Using Filters, Using the Detail panel, using the Size panels, customizing filters, Using and Customizing tooltips, Formatting your data with colors.
8. Creating Dashboards & Storytelling, creating your first dashboard and Story, Design for different displays, adding interactivity to your Dashboard, Distributing & Publishing your Visualization.
9. Tableau file types, publishing to Tableau Online, Sharing your visualizations, printing, and Exporting.
10. Creating custom charts, cyclical data and circular area charts, Dual Axis charts.

**REFERENCE BOOKS:**

1. Microsoft Power BI cookbook, Brett Powell, 2nd edition.
2. R Programming for Data Science by Roger D. Peng (References)
3. The Art of R Programming by Norman Matloff Cengage Learning India.

**\*MC309: GENDER SENSITIZATION LAB****B.Tech. II Year I Sem.**

L	T	P	C
0	0	2	0

**COURSE DESCRIPTION**

This course offers an introduction to Gender Studies, an interdisciplinary field that asks critical questions about the meanings of sex and gender in society. The primary goal of this course is to familiarize students with key issues, questions and debates in Gender Studies, both historical and contemporary. It draws on multiple disciplines – such as literature, history, economics, psychology, sociology, philosophy, political science, anthropology and media studies – to examine cultural assumptions about sex, gender, and sexuality.

This course integrates analysis of current events through student presentations, aiming to increase awareness of contemporary and historical experiences of women, and of the multiple ways that sex and gender interact with race, class, caste, nationality and other social identities. This course also seeks to build an understanding and initiate and strengthen programmes combating gender-based violence and discrimination. The course also features several exercises and reflective activities designed to examine the concepts of gender, gender-based violence, sexuality, and rights. It will further explore the impact of gender-based violence on education, health and development.

**Objectives of the Course**

- To develop students' sensibility with regard to issues of gender in contemporary India.
- To provide a critical perspective on the socialization of men and women.
- To introduce students to information about some key biological aspects of genders.
- To expose the students to debates on the politics and economics of work.
- To help students reflect critically on gender violence.
- To expose students to more egalitarian interactions between men and women.

**Learning Outcomes**

- Students will have developed a better understanding of important issues related to gender in contemporary India.
- Students will be sensitized to basic dimensions of the biological, sociological, psychological and legal aspects of gender. This will be achieved through discussion of materials derived from research, facts, everyday life, literature and film.
- Students will attain a finer grasp of how gender discrimination works in our society and how to counter it.
- Students will acquire insight into the gendered division of labor and its relation to politics and economics.
- Men and women students and professionals will be better equipped to work and live together as equals.
- Students will develop a sense of appreciation of women in all walks of life.
- Through providing accounts of studies and movements as well as the new laws that provide protection and relief to women, the textbook will empower students to understand and respond to gender violence.

**Unit-I: UNDERSTANDING GENDER**

Introduction: Definition of Gender-Basic Gender Concepts and Terminology-Exploring Attitudes towards Gender-Construction of Gender-Socialization: Making Women, Making Men  
- Preparing for Womanhood. Growing up Male. First lessons in Caste.

**Unit – II: GENDER ROLES AND RELATIONS**

Two or Many? -Struggles with Discrimination-Gender Roles and Relations-Types of Gender Roles- Gender Roles and Relationships Matrix-Missing Women-Sex Selection and Its Consequences- Declining Sex Ratio. Demographic Consequences-Gender Spectrum: Beyond the Binary

### Unit – III: GENDER AND LABOUR

Division and Valuation of Labour-Housework: The Invisible Labor- “My Mother doesn’t Work.” “Share the Load.”-Work: Its Politics and Economics -Fact and Fiction. Unrecognized and Unaccounted work.

-Gender Development Issues-Gender, Governance and Sustainable Development-Gender and Human Rights-Gender and Mainstreaming

### Unit – IV: GENDER - BASED VIOLENCE

The Concept of Violence- Types of Gender-based Violence-Gender-based Violence from a Human Rights Perspective-Sexual Harassment: Say No!-Sexual Harassment, not Eve-teasing- Coping with Everyday Harassment- Further Reading: “*Chupulu*”.

Domestic Violence: Speaking Out Is Home a Safe Place? -When Women Unite [Film]. Rebuilding Lives. Thinking about Sexual Violence Blaming the Victim-“I Fought for my Life....”

### Unit – V: GENDER AND CULTURE

Gender and Film-Gender and Electronic Media-Gender and Advertisement-Gender and Popular Literature- Gender Development Issues-Gender Issues-Gender Sensitive Language- Gender and Popular Literature - Just Relationships: Being Together as Equals

Mary Kom and Onler. Love and Acid just do not Mix. Love Letters. Mothers and Fathers. Rosa Parks-The Brave Heart.

**Note:** Since it is Interdisciplinary Course, Resource Persons can be drawn from the fields of English Literature or Sociology or Political Science or any other qualified faculty who has expertise in this field from engineering departments.

- Classes will consist of a combination of activities: dialogue-based lectures, discussions, collaborative learning activities, group work and in-class assignments. Apart from the above prescribed book, Teachers can make use of any authentic materials related to the topics given in the syllabus on “Gender”.

👉 **ESSENTIAL READING:** The Textbook, “*Towards a World of Equals: A Bilingual Textbook on Gender*” written by A.Suneetha, Uma Bhugubanda, Duggirala Vasanta, Rama Melkote, Vasudha Nagaraj, Asma Rasheed, Gogu Shyamala, Deepa Sreenivas and Susie Tharu published by Telugu Akademi, Telangana Government in 2015.

#### ASSESSMENT AND GRADING:

- Discussion & Classroom Participation: 20%
- Project/Assignment: 30%
- End Term Exam: 50%

**CS401PC: DISCRETE MATHEMATICS****B.Tech. II Year II Sem.**

L	T	P	C
3	0	0	3

**Course Objectives:**

- Introduces elementary discrete mathematics for computer science and engineering.
- Topics include formal logic notation, methods of proof, induction, sets, relations, algebraic structures, elementary graph theory, permutations and combinations, counting principles; recurrence relations and generating functions.

**Course Outcomes:**

- Understand and construct precise mathematical proofs
- Apply logic and set theory to formulate precise statements
- Analyze and solve counting problems on finite and discrete structures
- Describe and manipulate sequences
- Apply graph theory in solving computing problems

**UNIT - I**

**Mathematical logic:** Introduction, Statements and Notation, Connectives, Normal Forms, Theory of Inference for the Statement Calculus, The Predicate Calculus, Inference Theory of the Predicate Calculus.

**UNIT - II**

**Set theory:** Introduction, Basic Concepts of Set Theory, Representation of Discrete Structures, Relations and Ordering, Functions.

**UNIT - III**

**Algebraic Structures:** Introduction, Algebraic Systems, Semi groups and Monoids, Lattices as Partially Ordered Sets, Boolean Algebra.

**UNIT - IV**

**Elementary Combinatorics:** Basics of Counting, Combinations and Permutations, Enumeration of Combinations and Permutations, Enumerating Combinations and Permutations with Repetitions, Enumerating Permutation with Constrained Repetitions, Binomial Coefficient, The Binomial and Multinomial Theorems, The Principle of Exclusion.

**UNIT - V**

**Graph Theory:** Basic Concepts, Isomorphism and Subgraphs, Trees and their Properties, Spanning Trees, Directed Trees, Binary Trees, Planar Graphs, Euler's Formula, Multi-graphs and Euler Circuits, Hamiltonian Graphs, Chromatic Numbers, The Four-Color Problem.

**TEXT BOOKS:**

1. Discrete Mathematical Structures with Applications to Computer Science: J.P. Tremblay, R. Manohar, McGraw-Hill, 1<sup>st</sup> ed.
2. Discrete Mathematics for Computer Scientists & Mathematicians: Joe I. Mott, Abraham Kandel, Theodore P. Baker, Prentis Hall of India, 2<sup>nd</sup> ed.

**REFERENCE BOOKS:**

1. Discrete and Combinatorial Mathematics - an applied introduction: Ralph.P. Grimald, Pearson education, 5<sup>th</sup> edition.
2. Discrete Mathematical Structures: Thomas Kosy, Tata McGraw Hill publishing co.



**SM402MS: BUSINESS ECONOMICS AND FINANCIAL ANALYSIS****B.Tech. II Year II Sem.**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>3</b>	<b>0</b>	<b>0</b>	<b>3</b>

**Course Objective:** To learn the basic business types, impact of the economy on Business and Firms specifically. To analyze the Business from the Financial Perspective.

**Course Outcome:** The students will understand the various Forms of Business and the impact of economic variables on the Business. The Demand, Supply, Production, Cost, Market Structure, Pricing aspects are learnt. The Students can study the firm's financial position by analysing the Financial Statements of a Company.

**Unit – I: Introduction to Business and Economics**

**Business:** Structure of Business Firm, Theory of Firm, Types of Business Entities, Limited Liability Companies, Sources of Capital for a Company, Non-Conventional Sources of Finance.

**Economics:** Significance of Economics, Micro and Macro Economic Concepts, Concepts and Importance of National Income, Inflation, Money Supply and Inflation, Business Cycle, Features and Phases of Business Cycle. Nature and Scope of Business Economics, Role of Business Economist, Multidisciplinary nature of Business Economics.

**UNIT - II: Demand and Supply Analysis**

**Elasticity of Demand:** Elasticity, Types of Elasticity, Law of Demand, Measurement and Significance of Elasticity of Demand, Factors affecting Elasticity of Demand, Elasticity of Demand in decision making, Demand Forecasting: Characteristics of Good Demand Forecasting, Steps in Demand Forecasting, Methods of Demand Forecasting.

**Supply Analysis:** Determinants of Supply, Supply Function and Law of Supply.

**UNIT - III: Production, Cost, Market Structures & Pricing**

**Production Analysis:** Factors of Production, Production Function, Production Function with one variable input, two variable inputs, Returns to Scale, Different Types of Production Functions.

**Cost analysis:** Types of Costs, Short run and Long run Cost Functions.

**Market Structures:** Nature of Competition, Features of Perfect competition, Monopoly, Oligopoly, Monopolistic Competition. **Pricing:** Types of Pricing, Product Life Cycle based Pricing, Break Even Analysis, Cost Volume Profit Analysis.

**UNIT - IV: Financial Accounting:** Accounting concepts and Conventions, Accounting Equation, Double-Entry system of Accounting, Rules for maintaining Books of Accounts, Journal, Posting to Ledger, Preparation of Trial Balance, Elements of Financial Statements, Preparation of Final Accounts(Simple Problems).

**UNIT - V: Financial Ratios Analysis:** Concept of Ratio Analysis, Importance and Types of Ratios, Liquidity Ratios, Turnover Ratios, Profitability Ratios, Proprietary Ratios, Solvency, Leverage Ratios – Analysis and Interpretation (simple problems).

**TEXT BOOKS:**

1. D. D. Chaturvedi, S. L. Gupta, Business Economics - Theory and Applications, InternationalBook House Pvt. Ltd. 2013.
2. Dhanesh K Khatri, Financial Accounting, Tata Mc –Graw Hill, 2011.
3. Geethika Ghosh, Piyali Gosh, Purba Roy Choudhury, Managerial Economics, 2e, Tata McGraw Hill Education Pvt. Ltd. 2012.

**REFERENCE BOOKS:**

1. Paresh Shah, Financial Accounting for Management 2e, Oxford Press, 2015.
2. S. N. Maheshwari, Sunil K Maheshwari, Sharad K Maheshwari, Financial Accounting, 5e, VikasPublications, 2013.

**CS403PC: OPERATING SYSTEMS****B.Tech. II 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 "Computer Organization and Architecture".

**Course Objectives:**

- Introduce operating system concepts (i.e., processes, threads, scheduling, synchronization, deadlocks, memory management, file and I/O subsystems and protection)
- Introduce the issues to be considered in the design and development of operating system
- Introduce basic Unix commands, system call interface for process management, interprocess communication and I/O in Unix

**Course Outcomes:**

- Will be able to control access to a computer and the files that may be shared
- Demonstrate the knowledge of the components of computers and their respective roles in computing.
- Ability to recognize and resolve user problems with standard operating environments.
- Gain practical knowledge of how programming languages, operating systems, and architectures interact and how to use each effectively.

**UNIT - I**

**Operating System - Introduction**, Structures - Simple Batch, Multiprogrammed, Time-shared, Personal Computer, Parallel, Distributed Systems, Real-Time Systems, System components, Operating System services, System Calls

**Process** - Process concepts and scheduling, Operations on processes, Cooperating Processes, Threads

**UNIT - II**

**CPU Scheduling** - Scheduling Criteria, Scheduling Algorithms, Multiple -Processor Scheduling. System call interface for process management-fork, exit, wait, waitpid, exec

**Deadlocks** - System Model, Deadlocks Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, and Recovery from Deadlock

**UNIT - III**

**Process Management and Synchronization** - The Critical Section Problem, Synchronization Hardware, Semaphores, and Classical Problems of Synchronization, Critical Regions, Monitors

**Interprocess Communication Mechanisms:** IPC between processes on a single computer system, IPC between processes on different systems, using pipes, FIFOs, message queues, shared memory.

**UNIT - IV**

**Memory Management and Virtual Memory** - Logical versus Physical Address Space, Swapping, Contiguous Allocation, Paging, Segmentation, Segmentation with Paging, Demand Paging, Page Replacement, Page Replacement Algorithms.

**UNIT - V**

**File System Interface and Operations** -Access methods, Directory Structure, Protection, File System Structure, Allocation methods, Free-space Management. Usage of open, create, read, write, close, lseek, stat, ioctl system calls.

**TEXT BOOKS:**

1. Operating System Principles- Abraham Silberchatz, Peter B. Galvin, Greg Gagne 7th Edition, John Wiley.
2. Advanced programming in the UNIX environment, W.R. Stevens, Pearson education.

**REFERENCE BOOKS:**

1. Operating Systems- Internals and Design Principles, William Stallings, Fifth Edition– 2005, Pearson Education/PHI
2. Operating System A Design Approach- Crowley, TMH.
3. Modern Operating Systems, Andrew S. Tanenbaum 2nd edition, Pearson/PHI
4. UNIX programming environment, Kernighan and Pike, PHI/ Pearson Education
5. UNIX Internals -The New Frontiers, U. Vahalia, Pearson Education.

**CS404PC: DATABASE MANAGEMENT SYSTEMS****B.Tech. II Year II Sem.**

L	T	P	C
3	0	0	3

**Prerequisites:** A course on “Data Structures”.**Course Objectives:**

- To understand the basic concepts and the applications of database systems.
- To master the basics of SQL and construct queries using SQL.
- Topics include data models, database design, relational model, relational algebra, transaction control, concurrency control, storage structures and access techniques.

**Course Outcomes:**

- Gain knowledge of fundamentals of DBMS, database design and normal forms
- Master the basics of SQL for retrieval and management of data.
- Be acquainted with the basics of transaction processing and concurrency control.
- Familiarity with database storage structures and access techniques

**UNIT - I****Database System Applications:** A Historical Perspective, File Systems versus a DBMS, the Data Model, Levels of Abstraction in a DBMS, Data Independence, Structure of a DBMS**Introduction to Database Design:** Database Design and ER Diagrams, Entities, Attributes, and Entity Sets, Relationships and Relationship Sets, Additional Features of the ER Model, Conceptual Design With the ER Model**UNIT - II****Introduction to the Relational Model:** Integrity constraint over relations, enforcing integrity constraints, querying relational data, logical database design, introduction to views, destroying/altering tables and views.

Relational Algebra, Tuple relational Calculus, Domain relational calculus.

**UNIT - III****SQL: QUERIES, CONSTRAINTS, TRIGGERS:** form of basic SQL query, UNION, INTERSECT, and EXCEPT, Nested Queries, aggregation operators, NULL values, complex integrity constraints in SQL, triggers and active databases.**Schema Refinement:** Problems caused by redundancy, decompositions, problems related to decomposition, reasoning about functional dependencies, First, Second, Third normal forms, BCNF, lossless join decomposition, multivalued dependencies, Fourth normal form, Fifth normal form.**UNIT - IV**

Transaction Concept, Transaction State, Implementation of Atomicity and Durability, Concurrent Executions, Serializability, Recoverability, Implementation of Isolation, Testing for serializability, Lock Based Protocols, Timestamp Based Protocols, Validation- Based Protocols, Multiple Granularity, Recovery and Atomicity, Log-Based Recovery, Recovery with Concurrent Transactions.

**UNIT - V**

Data on External Storage, File Organization and Indexing, Cluster Indexes, Primary and Secondary Indexes, Index data Structures, Hash Based Indexing, Tree based Indexing, Comparison of File Organizations, Indexes- Intuitions for tree Indexes, Indexed Sequential Access Methods (ISAM),

B+ Trees: A Dynamic Index Structure.

**TEXT BOOKS:**

1. Database System Concepts, Silberschatz, Korth, McGraw hill, V edition.3rd Edition
2. Database Management Systems, Raghurama Krishnan, Johannes Gehrke, Tata Mc Graw Hill

**REFERENCE BOOKS:**

1. Database Systems design, Implementation, and Management, Peter Rob & Carlos Coronel 7thEdition.
2. Fundamentals of Database Systems, Elmasri Navrate, Pearson Education
3. Introduction to Database Systems, C. J. Date, Pearson Education
4. Oracle for Professionals, The X Team, S.Shah and V. Shah, SPD.
5. Database Systems Using Oracle: A Simplified guide to SQL and PL/SQL, Shah, PHI.
6. Fundamentals of Database Management Systems, M. L. Gillenson, Wiley Student Edition.

**CS405PC: SOFTWARE ENGINEERING****B.Tech. II Year II 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.

**CS406PC: OPERATING SYSTEMS LAB****B.Tech. II Year II Sem.**

L	T	P	C
0	0	2	1

**Prerequisites:** A course on “Programming for Problem Solving”, A course on “Computer Organization and Architecture”.

**Co-requisite:** A course on “Operating Systems”.

**Course Objectives:**

- To provide an understanding of the design aspects of operating system concepts through simulation
- Introduce basic Unix commands, system call interface for process management, interprocess communication and I/O in Unix

**Course Outcomes:**

- Simulate and implement operating system concepts such as scheduling, deadlock management, file management and memory management.
- Able to implement C programs using Unix system calls

**List of Experiments:**

1. Write C programs to simulate the following CPU Scheduling algorithms a) FCFS b) SJF c) RoundRobin d) priority
2. Write programs using the I/O system calls of UNIX/LINUX operating system (open, read, write, close, fcntl, seek, stat, opendir, readdir)
3. Write a C program to simulate Bankers Algorithm for Deadlock Avoidance and Prevention.
4. Write a C program to implement the Producer – Consumer problem using semaphores using UNIX/LINUX system calls.
5. Write C programs to illustrate the following IPC mechanisms a) Pipes b) FIFOs c) Message Queues d) Shared Memory
6. Write C programs to simulate the following memory management techniques a) Paging b) Segmentation
7. Write C programs to simulate Page replacement policies a) FCFS b) LRU c) Optimal

**TEXT BOOKS:**

1. Operating System Principles- Abraham Silberchatz, Peter B. Galvin, Greg Gagne 7<sup>th</sup> Edition, John Wiley
2. Advanced programming in the Unix environment, W.R.Stevens, Pearson education.

**REFERENCE BOOKS:**

1. Operating Systems – Internals and Design Principles, William Stallings, Fifth Edition– 2005, Pearson Education/PHI
2. Operating System - A Design Approach-Crowley, TMH.
3. Modern Operating Systems, Andrew S Tanenbaum, 2nd edition, Pearson/PHI
4. UNIX Programming Environment, Kernighan and Pike, PHI/Pearson Education
5. UNIX Internals: The New Frontiers, U. Vahalia, Pearson Education



**CS407PC: DATABASE MANAGEMENT SYSTEMS LAB****B.Tech. II Year II Sem.**

L	T	P	C
0	0	2	1

**Co-requisites:** "Database Management Systems"**Course Objectives:**

- Introduce ER data model, database design and normalization
- Learn SQL basics for data definition and data manipulation

**Course Outcomes:**

- Design database schema for a given application and apply normalization
- Acquire skills in using SQL commands for data definition and data manipulation.
- Develop solutions for database applications using procedures, cursors and triggers

**List of Experiments:**

1. Concept design with E-R Model
2. Relational Model
3. Normalization
4. Practicing DDL commands
5. Practicing DML commands
6. A. Querying (using ANY, ALL, UNION, INTERSECT, JOIN, Constraints etc.)  
B. Nested, Correlated subqueries
7. Queries using Aggregate functions, GROUP BY, HAVING and Creation and dropping of Views.
8. Triggers (Creation of insert trigger, delete trigger, update trigger)
9. Procedures
10. Usage of Cursors

**TEXT BOOKS:**

1. Database Management Systems, Raghurama Krishnan, Johannes Gehrke, Tata Mc Graw Hill, 3<sup>rd</sup> Edition
2. Database System Concepts, Silberschatz, Korth, McGraw Hill, V edition.

**REFERENCE BOOKS:**

1. Database Systems design, Implementation, and Management, Peter Rob & Carlos Coronel 7<sup>th</sup> Edition.
2. Fundamentals of Database Systems, Elmasri Navrate, Pearson Education
3. Introduction to Database Systems, C.J. Date, Pearson Education
4. Oracle for Professionals, The X Team, S. Shah and V. Shah, SPD.
5. Database Systems Using Oracle: A Simplified guide to SQL and PL/SQL, Shah, PHI.
6. Fundamentals of Database Management Systems, M. L. Gillenson, Wiley Student Edition.

**CS409PC: SKILL DEVELOPMENT COURSE (NODE JS/ REACT JS/  
DJANGO)**

**B.Tech. II Year II Sem.**

**L T P C**  
**0 0 2 1**

**Prerequisites:** Object Oriented Programming through Java, HTML Basics

**Course Objectives:**

- To implement the static web pages using HTML and do client side validation using JavaScript.
- To design and work with databases using Java
- To develop an end to end application using java full stack.
- To introduce Node JS implementation for server side programming.
- To experiment with single page application development using React.

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

- Build a custom website with HTML, CSS, and Bootstrap and little JavaScript.
- Demonstrate Advanced features of JavaScript and learn about JDBC
- Develop Server – side implementation using Java technologies like
- Develop the server – side implementation using Node JS.
- Design a Single Page Application using React.

**Exercises:**

1. Build a responsive web application for shopping cart with registration, login, catalog and cart pages using CSS3 features, flex and grid.
2. Make the above web application responsive web application using Bootstrap framework.
3. Use JavaScript for doing client – side validation of the pages implemented in experiment 1 and experiment 2.
4. Explore the features of ES6 like arrow functions, callbacks, promises, async/await. Implement an application for reading the weather information from openweathermap.org and display the information in the form of a graph on the web page.
5. Develop a java stand alone application that connects with the database (Oracle / mySql) and perform the CRUD operation on the database tables.
6. Create an xml for the bookstore. Validate the same using both DTD and XSD.
7. Design a controller with servlet that provides the interaction with application developed in experiment 1 and the database created in experiment 5.
8. Maintaining the transactional history of any user is very important. Explore the various session tracking mechanism (Cookies, HTTP Session)
9. Create a custom server using http module and explore the other modules of Node JS like OS, path, event.
10. Develop an express web application that can interact with REST API to perform CRUD operations on student data. (Use Postman)
11. For the above application create authorized end points using JWT (JSON Web Token).
12. Create a react application for the student management system having registration, login, contact, about pages and implement routing to navigate through these pages.
13. Create a service in react that fetches the weather information from openweathermap.org and display the current and historical weather information using graphical representation using chart.js
14. Create a TODO application in react with necessary components and deploy it into github.

**REFERENCE BOOKS:**

1. Jon Duckett, Beginning HTML, XHTML, CSS, and JavaScript, Wrox Publications, 2010
2. Bryan Basham, Kathy Sierra and Bert Bates, Head First Servlets and JSP, O'Reilly Media, 2nd Edition, 2008.

3. Vasan Subramanian, Pro MERN Stack, Full Stack Web App Development with Mongo, Express, React, and Node, 2<sup>nd</sup> Edition, A Press.

**\*MC410: CONSTITUTION OF INDIA****B.Tech. II Year II Sem.**

L	T	P	C
3	0	0	0

**Course Objectives:** Students will be able to:

- Understand the premises informing the twin themes of liberty and freedom from a civil rights perspective.
- To address the growth of Indian opinion regarding modern Indian intellectuals' constitutional role and entitlement to civil and economic rights as well as the emergence of nationhood in the early years of Indian nationalism.
- To address the role of socialism in India after the commencement of the Bolshevik Revolution in 1917 and its impact on the initial drafting of the Indian Constitution.

**Course Outcomes:** Students will be able to:

- Discuss the growth of the demand for civil rights in India for the bulk of Indians before the arrival of Gandhi in Indian politics.
- Discuss the intellectual origins of the framework of argument that informed the conceptualization of social reforms leading to revolution in India.
- Discuss the circumstances surrounding the foundation of the Congress Socialist Party [CSP] under the leadership of Jawaharlal Nehru and the eventual failure of the proposal of direct elections through adult suffrage in the Indian Constitution
- Discuss the passage of the Hindu Code Bill of 1956.

**Unit - 1** History of Making of the Indian Constitution- History of Drafting Committee.**Unit - 2** Philosophy of the Indian Constitution- Preamble Salient Features**Unit - 3** Contours of Constitutional Rights & Duties - Fundamental Rights

- Right to Equality
- Right to Freedom
- Right against Exploitation
- Right to Freedom of Religion
- Cultural and Educational Rights
- Right to Constitutional Remedies
- Directive Principles of State Policy
- Fundamental Duties.

**Unit - 4** Organs of Governance: Parliament, Composition, Qualifications and Disqualifications, Powers and Functions, Executive, President, Governor, Council of Ministers, Judiciary, Appointment and Transfer of Judges, Qualifications, Powers and Functions**Unit - 5** Local Administration: District's Administration head: Role and Importance, Municipalities: Introduction, Mayor and role of Elected Representative, CEO of Municipal Corporation. Panchayat raj: Introduction, PRI: Zila Panchayat. Elected officials and their roles, CEO Zila Panchayat: Position and role. Block level: Organizational Hierarchy (Different departments), Village level: Role of Elected and Appointed officials, Importance of grass root democracy**Unit - 6** Election Commission: Election Commission: Role and Functioning. Chief Election Commissioner and Election Commissioners. State Election Commission: Role and Functioning. Institute and Bodies for the welfare of SC/ST/OBC and women.**Suggested Reading:**

1. The Constitution of India, 1950 (Bare Act), Government Publication.
2. Dr. S. N. Busi, Dr. B. R. Ambedkar framing of Indian Constitution, 1st Edition, 2015.
3. M. P. Jain, Indian Constitution Law, 7th Edn., Lexis Nexis, 2014.
4. D.D. Basu, Introduction to the Constitution of India, Lexis Nexis, 2015.



## SHADAN WOMEN'S COLLEGE OF ENGINEERING & TECHNOLOGY

*A UGC Autonomous Institution, Affiliated to JNTUH*

Khairatabad, Hyderabad, Telangana (India).

**B.Tech. in INFORMATION TECHNOLOGY**

**COURSE STRUCTURE & SYLLABUS (R23 Regulations)**

**Applicable from AY 2025 - 26 Batch**

### III YEAR I SEMESTER

S. No.	Course Code	Course Title	L	T	P	Credits
1	IT501PC	Software Engineering	3	0	0	3
2	IT502PC	Data Communications and Computer Networks	3	1	0	4
3	IT503PC	Machine Learning	3	0	0	3
4		Professional Elective - I	3	0	0	3
5		Professional Elective - II	3	0	0	3
6	IT504PC	Software Engineering & Computer Networks Lab	0	0	2	1
7	IT505PC	Machine Learning Lab	0	0	2	1
8	EN508HS	Advanced English Communication Skills Lab	0	0	2	1
9	IT506PC	UI design- Flutter	0	0	2	1
10	*MC510	Intellectual Property Rights	3	0	0	0
<b>Total</b>			<b>18</b>	<b>1</b>	<b>8</b>	<b>20</b>

### III YEAR II SEMESTER

S. No.	Course Code	Course Title	L	T	P	Credits
1	IT601PC	Automata Theory and Compiler Design	3	0	0	3
2	IT602PC	Algorithms Design and Analysis	3	0	0	3
3	IT603PC	Embedded Systems	3	0	0	3
4		Professional Elective -III	3	0	0	3
5		Open Elective-I	3	0	0	3
6	IT604PC	Compiler Design Lab	0	0	2	1
7	IT605PC	Embedded Systems Lab	0	0	2	1
8		Professional Elective-III Lab	0	0	2	1
9	IT606PC	Industrial Oriented Mini Project/ Internship/ Skill Development Course (Big data-Spark)	0	0	4	2
10	*MC609	Environmental Science	3	0	0	0
<b>Total</b>			<b>18</b>	<b>0</b>	<b>10</b>	<b>20</b>

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

**\*MC – Satisfactory/Unsatisfactory**

**#Skill Course - 1 credit with 2 Practical Hours**

**Professional Elective - I**

IT511PE	Biometrics
IT512PE	Advanced Computer Architecture
IT513PE	Data Analytics
IT514PE	Image Processing
IT515PE	Principles of Programming Languages

**Professional Elective - II**

IT521PE	Computer Graphics
IT522PE	Quantum Computing
IT523PE	Advanced Operating Systems
IT524PE	Distributed Databases
IT525PE	Pattern Recognition

**Professional Elective - III**

IT631PE	Full Stack Development
IT632PE	Data Mining
IT633PE	Scripting Languages
IT634PE	Mobile Application Development
IT635PE	Software Testing Methodologies

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

**Open Elective -1:**

1. IT611OE: Java Programming
2. IT612OE: Object Oriented Programming using C++

**Open Elective -2:**

1. IT721OE: Full Stack development
2. IT722OE: Scripting Languages

**Open Elective -3:**

1. IT831OE: Big Data Technologies
2. IT832OE: DevOp

**IT501PC: 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.



**IT502PC: 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.

**IT503PC: MACHINE LEARNING****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

**IT511PE: BIOMETRICS (Professional Elective – I)****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](https://www.tutorialspoint.com/biometrics/biometrics_quick_guide.htm)

**IT512PE: ADVANCED COMPUTER ARCHITECTURE (Professional Elective – I)****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, 2<sup>nd</sup> Edition, Tata McGraw Hill Publishers.

**REFERENCE BOOKS:**

1. Computer Architecture, J.L. Hennessy and D.A. Patterson, 4<sup>th</sup> 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.

**IT513PE: DATA ANALYTICS (Professional Elective – I)****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.

**IT514PE: IMAGE PROCESSING (Professional Elective – I)****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, 2<sup>nd</sup> 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, 2004.

**IT515PE: PRINCIPLES OF PROGRAMMING LANGUAGES (Professional Elective – I)****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.

**IT521PE: COMPUTER GRAPHICS (Professional Elective – II)****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.

**3-D 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.

**IT522PE: QUANTUM COMPUTING (Professional Elective – II)****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

**IT523PE: 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, Niranjana G. Shivaratri, Tata McGraw-Hill Edition 2001

**REFERENCE BOOK**

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

**IT524PE: DISTRIBUTED DATABASES (Professional Elective – II)****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.

**IT525PE: PATTERN RECOGNITION (Professional Elective – II)****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
- Analyza 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 Morkov 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, 1<sup>st</sup> 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.



**IT504PC: 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

**IT505PC: MACHINE LEARNING LAB****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.

**EN508HS: ADVANCED ENGLISH COMMUNICATION SKILLS LAB****III Year B.Tech. I Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>0</b>	<b>0</b>	<b>2</b>	<b>1</b>

**1. Introduction**

The introduction of the Advanced English Communication Skills Lab is considered essential at the B.Tech 3<sup>rd</sup> 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**, 10<sup>th</sup> 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*. (2<sup>nd</sup> 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*. (5<sup>th</sup> 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*. (4<sup>TH</sup> 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.

**IT506PC: UI DESIGN-FLUTTER****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.  
a) 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.

**\*MC510: INTELLECTUAL PROPERTY RIGHTS****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.

**IT601PC: AUTOMATA THEORY AND COMPILER DESIGN****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  $\epsilon$ -transitions to NFA without  $\epsilon$ -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, 3<sup>rd</sup> 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, 2<sup>nd</sup> 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.



**IT602PC: ALGORITHMS DESIGN AND ANALYSIS****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.

**IT603PC: 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.

**IT631PE: FULL STACK DEVELOPMENT (Professional Elective – III)****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, 2<sup>nd</sup> Edition, Addison-Wesley, 2019.
2. Mark Tielens Thomas, React in Action, 1<sup>st</sup> Edition, Manning Publications.

**REFERENCE BOOKS:**

1. Vasan Subramanian, Pro MERN Stack, Full Stack Web App Development with Mongo, Express, React, and Node, 2<sup>nd</sup> 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', 1<sup>st</sup> edition, Apress, 2018.
3. Kirupa Chinnathambi, Learning React: A Hands-On Guide to Building Web Applications Using React and Redux, 2<sup>nd</sup> edition, Addison-Wesley Professional, 2018.

**IT632PE: 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, 3<sup>rd</sup> Edition, Morgan Kaufmann/Elsevier, 2012.
2. Margaret H Dunham, Data Mining Introductory and Advanced Topics, 2<sup>nd</sup> Edition, Pearson Education, India, 2006.

**REFERENCE BOOKS:**

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

**IT633PE: SCRIPTING LANGUAGES (Professional Elective – III)****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.

**IT634PE: MOBILE APPLICATION DEVELOPMENT (Professional Elective – III)****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

**IT635PE: SOFTWARE TESTING METHODOLOGIES (Professional Elective – III)****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.

**IT6110E: 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.

**IT6120E: 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.

**IT604PC: 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.

**IT605PC: 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.



**IT611PE: FULL STACK DEVELOPMENT LAB (Professional Elective – III)****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, 2<sup>nd</sup> Edition, Addison-Wesley, 2019.
2. Mark Tielens Thomas., React in Action, 1<sup>st</sup> Edition, Manning Publications.

**REFERENCE BOOKS:**

1. Vasan Subramanian, Pro MERN Stack, Full Stack Web App Development with Mongo, Express, React, and Node, 2<sup>nd</sup> 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’, 1<sup>st</sup> 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, 2<sup>nd</sup> edition, Addison-Wesley Professional, 2018.

**IT612PE: 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

**IT613PE: SCRIPTING LANGUAGES LAB (Professional Elective – III)****B.Tech. III Year II Sem.**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>0</b>	<b>0</b>	<b>2</b>	<b>1</b>

**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 Pramatic 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.

**IT614PE: MOBILE APPLICATION DEVELOPMENT LAB (Professional Elective – III)****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.

**IT615PE: SOFTWARE TESTING METHODOLOGIES LAB (Professional Elective – III)****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, 2<sup>nd</sup> 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.

**IT606PC: BIG DATA-SPARK****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](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\\_s\\_hared/overview](https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01258388119638835242_s_hared/overview)
3. [https://infyspringboard.onwingspan.com/web/en/app/toc/lex\\_auth\\_0126052684230082561692\\_shared/overview](https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_0126052684230082561692_shared/overview)



**\*MC609: 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, 4<sup>th</sup> Edition, New age international publishers.
5. Text book of Environmental Science and Technology - Dr. M. Anji Reddy 2007, BS Publications.