PALLAVI ENGINEERING COLLEGE HYDERABAD
B.Tech. in COMPUTER SCIENCE AND ENGINEERING COURSE
STRUCTURE & SYLLABUS (PR24 Regulations)

Applicable from AY 2025-26 Batch

III YEAR I SEMESTER

S. No.	Course Code	Course Title	L	T	P	Credits
1	PCS501PC	Design and Analysis of Algorithms	3	1	0	4
2	PCS502PC	Computer Networks	3	0	0	3
3	PCS503PC	DevOps	3	0	0	3
4		Professional Elective-I	3	0	0	3
5		Professional Elective -II	3	0	0	3
6	PCS504PC	Computer Networks Lab	0	0	2	1
7	PCS505PC	DevOps Lab	0	0	2	1
8	PEN508HS	Advanced English Communication Skills Lab	0	0	2	1
9	PCS506PC	UI design- Flutter	0	0	2	1
10	*PMC510	Intellectual Property Rights	3	0	0	0
		Total	18	1	8	20

III YEAR II SEMESTER

S. No.	Course Code	Course Title	L	T	P	Credits
1	PCS601PC	Machine Learning	3	0	0	3
2	PCS602PC	Formal Languages and Automata Theory	3	0	0	3
3	PCS603PC	Artificial Intelligence	3	0	0	3
4		Professional Elective - III	3	0	0	3
5		Open Elective-I	3	0	0	3
6	PCS604PC	Machine Learning Lab	0	0	2	1
7	PCS605PC	Artificial Intelligence Laboratory	0	0	2	1
8		Professional Elective-III Lab	0	0	2	1
9	PCS606PC	Industrial Oriented Mini Project/ Internship/ Skill Development Course (Big data-Spark)	0	0	4	2
10	*PMC609	Environmental Science	3	0	0	0
		Total	18	0	8	20

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

(B. Yuvasai)

Dr L K Suresh Kumar

C. Raghavendra
(C. R. Rao)

(Dr. Sai Hareesh)

Dr. S. Suresh kumar

(Dhananjaya Rao Kodali)

(Dr. C.R. Rao)

IV YEAR I SEMESTER

S. No.	Course Code	Course Title	L.	T	P	Credits
1	PCS701PC	Cryptography and Network Security	3	0	0	3
2	PCS702PC	Compiler Design	3	0	0	3
3		Professional Elective -IV	3	0	0	3
4		Professional Elective -V	3	0	0	3
5		Open Elective - II	3	0	0	3
6	PCS703PC	Cryptography and Network Security Lab	0	0	2	1
7	PCS704PC	Compiler Design Lab	0	0	2	1
8	PCS705PC	Project Stage - I	0	0	6	3
		Total Credits	15	0	10	20

IV YEAR II SEMESTER

S. No.	Course Code	Course Title	L.	T	P	Credits
1	PCS801PC	Organizational Behavior	3	0	0	3
2		Professional Elective - VI	3	0	0	3
3		Open Elective - III	3	0	0	3
4		Project Stage - II including Seminar	0	0	22	11
		Total Credits	9	0	22	20

*MC – Satisfactory/Unsatisfactory

#Skill Course - 1 credit with 2 Practical Hours

Professional Elective - I

PCS511PE	Quantum Computing
PCS512PE	Advanced Computer Architecture
PCS513PE	Data Analytics
PCS514PE	Image Processing
PCS515PE	Principles of Programming Languages

Professional Elective - II

PCS521PE	Computer Graphics
PCS522PE	Embedded Systems
PCS523PE	Information Retrieval Systems
PCS524PE	Distributed Databases
PCS525PE	Natural Language Processing

Professional Elective - III

PCS631PE	Full Stack Development
PCS632PE	Internet of Things
PCS633PE	Scripting Languages
PCS634PE	Mobile Application Development
PCS635PE	Software Testing Methodologies

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

Professional Elective -IV

PCS741PE	Graph Theory
PCS742PE	Cyber Security
PCS743PE	Soft Computing
PCS744PE	Cloud Computing
PCS745PE	Ad hoc & Sensor Networks

Professional Elective -V

PCS751PE	Advanced Algorithms
PCS752PE	Agile Methodology
PCS753PE	Robotic Process Automation
PCS754PE	Blockchain Technology
PCS755PE	Software Process & Project Management

Professional Elective – VI

PCS861PE	Computational Complexity
PCS862PE	Distributed Systems
PCS863PE	Deep Learning
PCS864PE	Human Computer Interaction
PCS865PE	Cyber Forensics

Open Elective -1:
1. PCS611OE: Data Structures
2. PCS612OE: Database Management Systems

Open Elective -2:
1. PCS721OE: Operating Systems
2. PCS722OE: Software Engineering

Open Elective -3:
1. PCS831OE: Algorithms Design and Analysis
2. PCS832OE: Introduction to Computer Networks

PCS301PC: DIGITAL ELECTRONICS

B.Tech. II Year I Sem.

L T P C
3 0 0 8

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 Flow Tables, 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, 5th 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, 5TH Edition, M. Rafiquzzaman John Wiley.

PCS302PC: 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.

PCS303PC: 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, 9th 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, 8th Edition, Pearson Educations.

PCS304PC: 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, McGraw Hill.
2. Computer Organization and Architecture - William Stallings Sixth Edition, Pearson/PHI.
3. Structured Computer Organization - Andrew S. Tanenbaum, 4 th Edition, PHI/Pearson.

A.Sai Hareesh

PCS305PC: 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.

PCS306PC: DATA STRUCTURES LAB

B.Tech. II Year I Sem.

L	T	P	C
0	0	3	1.5

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 linked list.:
   i) Creation   ii) Insertion   iii) Deletion   iv) Traversal
2. Write a program that uses functions to perform the following operations on doubly linked list.:
   i) Creation   ii) Insertion   iii) Deletion   iv) Traversal
3. Write a program that uses functions to perform the following operations on circular linked list.:
   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)   AVL trees   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, 2nd 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, 2nd Edition, R. F. Gilberg and B. A. Forouzan, Cengage Learning.

PCS307PC: 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.

PCS308PC: 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.

*PMC309: 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 OutIs 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 Bhrugubanda, DuggiralaVasanta, 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%

PCS401PC: 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, 1st ed.
2. Discrete Mathematics for Computer Scientists & Mathematicians: Joe I. Mott, Abraham Kandel, Teodore P. Baker, Prentis Hall of India, 2nd ed.

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

PSM402MS: BUSINESS ECONOMICS AND FINANCIAL ANALYSIS

B.Tech. II Year II Sem.

L T P C
3 0 0 3

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, International Book 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 Mc Graw 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, Vikas Publications, 2013.

PCS403PC: 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.

PCS404PC: 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 7th 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.

PCS405PC: 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 Rambaugh, 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.

PCS406PC: 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) Round Robin 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 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

PCS407PC: 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 Cursorst.

TEXT BOOKS:

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

REFERENCE BOOKS:

1. Database Systems design, Implementation, and Management, Peter Rob & Carlos Coronel 7th 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.

PCS409PC: 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 the 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, 2nd Edition, A Press.

*PMC410: 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 ZilaPanchayat: 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.

PCS501PC: DESIGN AND ANALYSIS OF ALGORITHMS

B.Tech. III Year I Sem.

L T P C
3 1 0 4

Prerequisites:
1. A course on "Computer Programming and Data Structures".
2. A course on "Advanced Data Structures".

Course Objectives:
- Introduces the notations for analysis of the performance of algorithms and the data structure of disjoint sets.
- 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, hamitonian cycles.

UNIT - III
Dynamic Programming: General method, applications- Optimal binary search tree, 0/1 knapsack problem, All pairs shortest path problem, Traveling salesperson 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 - Traveling salesperson 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, 1998.

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.

PCS502PC: COMPUTER NETWORKS

B.Tech. III Year I Sem.   L T P C
                                                  3 0 0 3

Prerequisites
1. A course on "Programming for problem solving"
2. A course on "Data Structures"

Course Objectives
- The objective of the course is to equip the students with a general overview of the concepts and fundamentals of computer networks.
- Familiarize the students with the standard models for the layered approach to communication between machines in a network and the protocols of the various layers.

Course Outcomes
- Gain the knowledge of the basic computer network technology.
- Gain the knowledge of the functions of each layer in the OSI and TCP/IP reference model.
- Obtain the skills of subnetting and routing mechanisms.
- Familiarity with the essential protocols of computer networks, and how they can be applied in network design and implementation.

UNIT - I
Network hardware, Network software, OSI, TCP/IP Reference models, Example Networks: ARPANET, Internet.
Physical Layer: Guided Transmission media: twisted pairs, coaxial cable, fiber optics, Wireless Transmission.
Data link layer: Design issues, framing, Error detection and correction.

UNIT - II
Elementary data link protocols: simplex protocol, A simplex stop and wait protocol for an error-free channel, A simplex stop and wait protocol for noisy channel.
Sliding Window protocols: A one-bit sliding window protocol, A protocol using Go-Back-N, A protocol using Selective Repeat, Example data link protocols.
Medium Access sublayer: The channel allocation problem, Multiple access protocols: ALOHA, Carrier sense multiple access protocols, collision free protocols. Wireless LANs, Data link layer switching.

UNIT - III
Network Layer: Design issues, Routing algorithms: shortest path routing, Flooding, Hierarchical routing, Broadcast, Multicast, distance vector routing, Congestion Control Algorithms, Quality of Service, Internetworking, The Network layer in the internet.

UNIT - IV
Transport Layer: Transport Services, Elements of Transport protocols, Connection management, TCP and UDP protocols.

UNIT - V
Application Layer -Domain name system, SNMP, Electronic Mail; the World WEB, HTTP, Streaming audio and video.

TEXT BOOK:
1. Computer Networks -- Andrew S Tanenbaum, David. j. Wetherall, 5th Edition. Pearson Education/PHI

REFERENCE BOOKS:
1. An Engineering Approach to Computer Networks-S. Keshav, 2nd Edition, Pearson Education
2. Data Communications and Networking - Behrouz A. Forouzan. Third Edition TMH.

PCS503PC: DEVOPS

B.Tech. III Year I Sem.   L T P C
                                                                          3   0   0   3

Pre-Requisites:
1. Software Engineering
2. Software Project Management

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

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

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

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

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

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

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

TEXT BOOKS:
1. Joakim Verona., Practical DevOps, Packt Publishing, 2016.

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

PCS511PE: QUANTUM COMPUTING (Professional Elective – I)

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

PCS512PE: 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, Multivetor and SIMD computers.

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

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

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

PCS513PE: 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, Addision Wisley, 2006.
2. Data Mining Analysis and Concepts, M. Zaki and W. Meira
3. Mining of Massive Datasets, Jure Leskovec Stanford Univ. Anand Rajaraman Milliway Labs Jeffrey D Ullman Stanford Univ.

PCS514PE: 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, 2nd Ed, 2004.

REFERENCE BOOKS:
1. Fundamentals of Digital Image Processing: A. K. Jain, PHI.
2. Digital Image Processing using MAT LAB: 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.

PCS515PE: 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 Circui t 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. Louden, 2nd Edition, Thomson, 2003.

PCS521PE: 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.

PCS522PE: EMBEDDED SYSTEMS (Professional Elective – Ii)

B.Tech. III Year I 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.

PCS523PE: INFORMATION RETRIEVAL SYSTEMS (Professional Elective – II)

B.Tech. III Year I Sem.

L T P C
3 0 0 3

Prerequisites:
1. Data Structures

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

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

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

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

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

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

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

TEXT BOOK:
1. Information Storage and Retrieval Systems - Theory and Implementation, Second Edition, Gerald J. Kowalski, Mark T. Maybury, Springer

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

PCS524PE: 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.

PCS525PE: NATURAL LANGUAGE PROCESSING (Professional Elective – II)

B.Tech. III Year I Sem.

L T P C
3 0 0 3

Prerequisites:
1. Data structures and compiler design

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

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

UNIT - I
Finding the Structure of Words: Words and Their Components, Issues and Challenges, Morphological Models
Finding the Structure of Documents: Introduction, Methods, Complexity of the Approaches, Performances of the Approaches, Features

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

UNIT – III
Syntax II: Models for Ambiguity Resolution in Parsing, Multilingual Issues
Semantic Parsing I: Introduction, Semantic Interpretation, System Paradigms, Word Sense

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

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

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

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

PCS504PC: COMPUTER NETWORKS LAB

B.Tech. III Year I Sem.   L T P C
                                                                        0 0 2 1

Course Objectives

- To understand the working principle of various communication protocols.
- To understand the network simulator environment and visualize a network topology and observe its performance
- To analyze the traffic flow and the contents of protocol frames

Course Outcomes

- Implement data link layer farming methods
- Analyze error detection and error correction codes.
- Implement and analyze routing and congestion issues in network design.
- Implement Encoding and Decoding techniques used in presentation layer
- To be able to work with different network tools

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 Dijsktra'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
8. Write a program for congestion control using Leaky bucket algorithm.
9. Write a program for frame sorting techniques used in buffers.
10. **Wireshark**
    i.   Packet Capture Using Wire shark
    ii. Starting Wire shark
    iii. Viewing Captured Traffic
    iv. Analysis and Statistics & Filters.
    How to run Nmap scan
    Operating System Detection using Nmap
    Do the following using NS2 Simulator
    i.   NS2 Simulator-Introduction
    ii. Simulate to Find the Number of Packets Dropped
    iii. Simulate to Find the Number of Packets Dropped by TCP/UDP
    iv. Simulate to Find the Number of Packets Dropped due to Congestion
    v.   Simulate to Compare Data Rate & Throughput.
    vi. Simulate to Plot Congestion for Different Source/Destination
    vii. Simulate to Determine the Performance with respect to Transmission of Packets

TEXT BOOK:

1. Computer Networks, Andrew S Tanenbaum, David. j. Wetherall, 5th Edition. Pearson Education/PHI.

REFERENCE BOOKS:

1. An Engineering Approach to Computer Networks, S. Keshav, 2nd Edition, Pearson Education.
2. Data Communications and Networking - Behrouz A. Forouzan. 3rd Edition, TMH.

PCS505PC: DEVOPS LAB

B.Tech. III Year I Sem.   L T P C
                                                                                  0 0 2 1

Course Objectives:

- Develop a sustainable infrastructure for applications and ensure high scalability. DevOps aims to shorten the software development lifecycle to provide continuous delivery with high-quality.

Course Outcomes:

1. Understand the need of DevOps tools
2. Understand the environment for a software application development
3. Apply different project management, integration and development tools
4. Use Selenium tool for automated testing of application

List of Experiments:

1. Write code for a simple user registration form for an event.
2. Explore Git and GitHub commands.
3. Practice Source code management on GitHub. Experiment with the source code in exercise 1.
4. Jenkins installation and setup, explore the environment.
5. Demonstrate continuous integration and development using Jenkins.
6. Explore Docker commands for content management.
7. Develop a simple containerized application using Docker.
8. Integrate Kubernetes and Docker
9. Automate the process of running containerized application for exercise 7 using Kubernetes.
10. Install and Explore Selenium for automated testing.
11. Write a simple program in JavaScript and perform testing using Selenium.
12. Develop test cases for the above containerized application using selenium.

TEXT BOOKS:

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

REFERENCE BOOKS:

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

PEN508HS: ADVANCED ENGLISH COMMUNICATION SKILLS LAB

B.Tech. III Year I Sem.

```
L  T  P  C
0  0  2  1
```

1. Introduction

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

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

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

2. Objectives:

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

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

3. Syllabus:

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

1. **Activities on Listening and Reading Comprehension:** Active Listening - Development of Listening Skills Through Audio clips - Benefits of Reading - Methods and Techniques of Reading - Basic Steps to Effective Reading - Common Obstacles - Discourse Markers or Linkers - Sub- skills of reading - Reading for facts, negative facts and Specific Details- Guessing Meanings from Context, Inferring Meaning - Critical Reading – Reading Comprehension - Exercises for Practice.

2. **Activities on Writing Skills**: Vocabulary for Competitive Examinations - Planning for Writing – Improving Writing Skills - Structure and presentation of different types of writing - Free Writing and Structured Writing - Letter Writing -Writing a Letter of Application -Resume vs. Curriculum Vitae - Writing a Résumé – Styles of Résumé - e-Correspondence - Emails - Blog Writing - (N)etiquette - Report Writing - Importance of Reports - Types and Formats of Reports- Technical Report Writing- Exercises for Practice.

3. **Activities on Presentation Skills -** Starting a conversation - responding appropriately and relevantly - using the right language and body language - Role Play in different situations including Seeking Clarification, Making a Request, Asking for and Refusing Permission, Participating in a Small Talk - Oral presentations (individual and group) through JAM sessions- PPTs - Importance of Presentation Skills - Planning, Preparing, Rehearsing and Making a Presentation - Dealing with Glossophobia or Stage Fear – Understanding Nuances of Delivery - Presentations through Posters/Projects/Reports - Checklist for Making a Presentation and Rubrics of Evaluation

4. **Activities on Group Discussion (GD):** Types of GD and GD as a part of a Selection Procedure - Dynamics of Group Discussion- Myths of GD - Intervention, Summarizing - Modulation of Voice, Body Language, Relevance, Fluency and Organization of Ideas — Do's and Don'ts - GD Strategies – Exercises for Practice.

5. **Interview Skills**: Concept and Process - Interview Preparation Techniques - Types of Interview Questions - Pre-interview Planning, Opening Strategies, Answering Strategies - Interview Through Tele-conference & Video-conference - Mock Interviews.

4. **Minimum Requirement:**

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

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

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

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

6. Books Recommended:

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

PCS506PC: 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 Widges and customize widgets for specific UI elements, Themes
- Understand to include animation apart from fetching data

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

List of Experiments: Students need to implement the following experiments
1. a) Install Flutter and Dart SDK.
   b) Write a simple Dart program to understand the language basics.

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

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

4. a) Set up navigation between different screens using Navigator.
   b) Implement navigation with named routes.

5. a) Learn about stateful and stateless widgets.
   b) Implement state management using set State and Provider.

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

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

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

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

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

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

*PMC510: 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.

PCS601PC: MACHINE LEARNING

B.Tech. III Year II 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

PCS602PC: FORMAL LANGUAGES AND AUTOMATA THEORY

B.Tech. III Year II Sem.

L T P C
3 0 0 3

Course Objectives

- To provide introduction to some of the central ideas of theoretical computer science from the perspective of formal languages.
- To introduce the fundamental concepts of formal languages, grammars and automata theory.
- Classify machines by their power to recognize languages.
- Employ finite state machines to solve problems in computing.
- To understand deterministic and non-deterministic machines.
- To understand the differences between decidability and undecidability.

Course Outcomes

- Understand the concept of abstract machines and their power to recognize the languages.
- Employ finite state machines for modeling and solving computing problems.
- Design context free grammars for formal languages.
- Distinguish between decidability and undecidability.

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 €-transitions to NFA without €-transitions. Conversion of NFA to DFA, Moore and Melay machines

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.

Closure Properties of Regular Languages: Closure properties of Regular languages, Decision Properties of Regular Languages, Equivalence and Minimization of Automata.

UNIT - III

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

Push Down Automata: Definition of the Pushdown Automaton, the Languages of a PDA, Equivalence of PDA's and CFG's, Acceptance by final state, Acceptance by empty stack, Deterministic Pushdown Automata. From CFG to PDA, From PDA to CFG.

UNIT - IV

Normal Forms for Context- Free Grammars: Eliminating useless symbols, Eliminating €-Productions. Chomsky Normal form Greibach Normal form.

Pumping Lemma for Context-Free Languages: Statement of pumping lemma, Applications

Closure Properties of Context-Free Languages: Closure properties of CFL's, Decision Properties of CFL's Turing Machines: Introduction to Turing Machine, Formal Description, Instantaneous description, The language of a Turing machine

UNIT - V

Types of Turing machine: Turing machines and halting

Undecidability: Undecidability, A Language that is Not Recursively Enumerable, An Undecidable Problem That is RE, Undecidable Problems about Turing Machines, Recursive languages, Properties of recursive languages, Post's Correspondence Problem, Modified Post Correspondence problem, Other Undecidable Problems, Counter machines.

TEXT BOOKS:

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

REFERENCE BOOKS:

1. Introduction to Languages and The Theory of Computation, John C Martin, TMH.
2. Introduction to Computer Theory, Daniel I.A. Cohen, John Wiley.
3. A Textbook on Automata Theory, P. K. Srimani, Nasir S. F. B, Cambridge University Press.
4. Introduction to the Theory of Computation, Michael Sipser, 3rd edition, Cengage Learning.
5. Introduction to Formal languages Automata Theory and Computation Kamala Krithivasan, Rama R, Pearson.

PCS603PC: ARTIFICIAL INTELLIGENCE

B.Tech. III Year II Sem.
```
L  T  P  C
3  0  0  3
```

Prerequisites:
1. Programming for problem solving, Data Structures.

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

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

UNIT - I

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

UNIT - II

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

UNIT - III

Logic and Knowledge Representation
First-Order Logic: Representation, Syntax and Semantics of First-Order Logic, Using First-Order Logic, Knowledge Engineering in First-Order Logic.
Inference in First-Order Logic: Propositional vs. First-Order Inference, Unification and Lifting, Forward Chaining, Backward Chaining, Resolution.

UNIT - IV

Knowledge Representation: Ontological Engineering, Categories and Objects, Events. Mental Events and Mental Objects, Reasoning Systems for Categories, Reasoning with Default Information.
Classical Planning: Definition of Classical Planning, Algorithms for Planning with State-Space Search, Planning Graphs, other Classical Planning Approaches, Analysis of Planning approaches.

UNIT - V

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

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

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

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

PCS631PE: 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:
- Understand Full stack components for developing web application.
- Apply packages of NodeJS to work with Data, Files, Http Requests and Responses.
- Use MongoDB data base for storing and processing huge data and connects with NodeJS application.
- Design faster and effective single page applications using Express and Angular.
- Create interactive user interfaces with react components.

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

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

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

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

UNIT-V

React:

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

TEXT BOOKS:

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

REFERENCE BOOKS:

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

PCS632PE: INTERNET OF THINGS (Professional Elective – III)

B.Tech. III Year II Sem.   **L T P C**
                                                                  3 0 0 3

Pre-Requisites: Computer organization, Computer Networks
Course Objectives:

- To introduce the terminology, technology and its applications
- To introduce the concept of M2M (machine to machine) with necessary protocols
- To introduce the Python Scripting Language which is used in many IoT devices
- To introduce the Raspberry PI platform, that is widely used in IoT applications
- To introduce the implementation of web-based services on IoT devices

Course Outcomes:

- Interpret the impact and challenges posed by IoT networks leading to new architectural models.
- Compare and contrast the deployment of smart objects and the technologies to connect them to network.
- Appraise the role of IoT protocols for efficient network communication.
- Identify the applications of IoT in Industry.

UNIT - I

Introduction to Internet of Things -Definition and Characteristics of IoT, Physical Design of IoT, Logical Design of IoT, IoT Enabling Technologies, IoT Levels and Deployment Templates
Domain Specific IoTs - Home automation, Environment, Agriculture, Health and Lifestyle

UNIT - II

IoT and M2M - M2M, Difference between IoT and M2M, SDN and NFV for IoT,
IoT System Management with NETCOZF, YANG- Need for IoT system Management, Simple Network management protocol Network operator requirements, NETCONF, YANG, IoT Systems Management with NETCONF-YANG

UNIT - III

IoT Systems – Logical design using Python-Introduction to Python – Python Data types & Data structures, Control flow, Functions, Modules, Packaging, File handling, Data/Time operations, Classes, Exception, Python packages of Interest for IoT

UNIT - IV

IoT Physical Devices and Endpoints - Raspberry Pi, Linux on Raspberry Pi, Raspberry Pi Interfaces, Programming Raspberry PI with Python, Other IoT devices.

IoT Physical Servers and Cloud Offerings – Introduction to Cloud Storage models and communication APIs, WAMP-AutoBahn for IoT, Xively Cloud for IoT, Python web application framework – Django, Designing a RESTful web API

UNIT V

Case studies- Home Automation, Environment-weather monitoring-weather reporting- air pollution monitoring, Agriculture.

TEXT BOOK:

1. Internet of Things - A Hands-on Approach, Arshdeep Bahga and Vijay Madisetti, Universities Press, 2015, ISBN: 9788173719547.

REFERENCE BOOK:

1. Getting Started with Raspberry Pi, Matt Richardson & Shawn Wallace, O'Reilly (SPD), 2014, ISBN: 9789350239759.

PCS633PE: 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 Pramatic Progammers 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.

PCS634PE: 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 improves 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

PCS635PE: 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.

PCS611OE: DATA STRUCTURES (Open Elective – I)

B.Tech. III Year II Sem.

L	T	P	C
3	0	0	3

Prerequisites
1. A course on "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
1. Ability to select the data structures that efficiently model the information in a problem.
2. Ability to assess efficiency trade-offs among different data structure implementations or combinations.
3. Implement and know the application of algorithms for sorting and pattern matching.
4. 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.

PCS612OE: DATABASE MANAGEMENT SYSTEMS (Open Elective – I)

B.Tech. III 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 7th 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.

PCS604PC: MACHINE LEARNING LAB

B.Tech. III Year II 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.

PCS605PC: ARTIFICIAL INTELLIGENCE LAB

B.Tech. III Year II Sem.

L T P C
0 0 2 1

Course Objectives:

- Become familiar with basic principles of AI toward problem solving, knowledge representation, and learning.

Course Outcomes:

- Apply basic principles of AI in solutions that require problem solving, knowledge representation, and learning.

LIST OF EXPERIMENTS

Write a Program to Implement the following using Python.

1. Breadth First Search
2. Depth First Search
3. Tic-Tac-Toe game
4. 8-Puzzle problem
5. Water-Jug problem
6. Travelling Salesman Problem
7. Tower of Hanoi
8. Monkey Banana Problem
9. Alpha-Beta Pruning
10. 8-Queens Problem

TEXT BOOK:

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

REFERENCE BOOKS:

1. Artificial Intelligence, 3rd Edn, E. Rich and K. Knight (TMH)
2. Artificial Intelligence, 3rd Edn., Patrick Henny Winston, Pearson Education.
3. Artificial Intelligence, Shivani Goel, Pearson Education.

PCS611PE: 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.

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

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

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

PCS612PE: INTERNET OF THINGS LAB (PROFESSIONAL ELECTIVE – III)

B.Tech. III Year II Sem.   L T P C
                                                                     0 0 2 1

Course Objectives
- To introduce the raspberry PI platform, that is widely used in IoT applications
- To introduce the implementation of distance sensor on IoT devices

Course Outcomes
- Ability to introduce the concept of M2M (machine to machine) with necessary protocols and get awareness in implementation of distance sensor
- Get the skill to program using python scripting language which is used in many IoT devices

List of Experiments

1. Using Raspberry pi
    - a. Calculate the distance using a distance sensor.
    - b. Interface an LED and switch with Raspberry pi.
    - c. Interface an LDR with Raspberrry Pi.

2. Using Arduino
    - a. Calculate the distance using a distance sensor.
    - b. Interface an LED and switch with Aurdino.
    - c. Interface an LDR with Aurdino
    - d. Calculate temperature using a temperature sensor.

3. Using Node MCU
    - a. Calculate the distance using a distance sensor.
    - b. Interface an LED and switch with Raspberry pi.
    - c. Interface an LDR with Node MCU
    - d. Calculate temperature using a temperature sensor.

4. Installing OS on Raspberry Pi
    - a) Installation using PiImager
    - b) Installation using image file
        - Downloading an Image
        - Writing the image to an SD card
            - using Linux
            - using Windows
        - Booting up Follow the instructions given in the URL
          https://www.raspberrypi.com/documentation/computers/getting-started.html

5. Accessing GPIO pins using Python
    - a) Installing GPIO Zero library.
         update your repositories list:
          install the package for Python 3:
    - b) Blinking an LED connected to one of the GPIO pin
    - c) Adjusting the brightness of an LED Adjust the brightness of an LED (0 to 100, where 100 means maximum brightness) using the in-built PWM wavelength.

6. Create a DJANGO project and an app.
7. Create a DJANGO view for weather station REST API
8. Create DJANGO template

9. Configure MYSQL with DJANGO framework

TEXT BOOKS:
1. Internet of Things - A Hands-on Approach, Arshdeep Bahga and Vijay Madisetti, Universities Press, 2015, ISBN: 9788173719547.
2. Getting Started with Raspberry Pi, Matt Richardson & Shawn Wallace, O'Reilly (SPD), 2014, ISBN: 9789350239759.

REFERENCE BOOKS:
1. Bernd Scholz-Reiter, Florian Michahelles, "Architecting the Internet of Things", ISBN 978-3-642-19156-5 e-ISBN 978-3-642-19157-2, Springer, 2016
2. N. Ida, Sensors, Actuators and Their Interfaces, Scitech Publishers, 2014.

PCS613PE: SCRIPTING LANGUAGES LAB (Professional Elective – III)

B.Tech. III Year II Sem.   **L T P C**
                                                                      0 0 2 1

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

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

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

LIST OF EXPERIMENTS

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

TEXT BOOKS:
1. The World of Scripting Languages, David Barron,Wiley Publications.
2. Ruby Programming language by David Flanagan and Yukihiro Matsumoto O'Reilly
3. "Programming Ruby" The Pramatic Progammers 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.

PCS614PE: 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.

PCS615PE: 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
- Design and develop the best test strategies in accordance with the development model.
- Design and develop GUI, Bitmap and database checkpoints
- Develop database checkpoints for different checks
- 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
6. Database checkpoint for custom check
6. Database checkpoint for runtime record check
6.
    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

    a. Batch testing without parameter passing
    b. Batch testing with parameter passing
11. Data driven batch
11. Silent mode test execution without any interruption
12. Test case for calculator in windows application

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

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

PCS606PC: 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:

- Develop MapReduce Programs to analyze large dataset Using Hadoop and Spark
- Write Hive queries to analyze large dataset Outline the Spark Ecosystem and its components
- Perform the filter, count, distinct, map, flatMap RDD Operations in Spark.
- Build Queries using Spark SQL
- Apply Spark joins on Sample Data Sets
- 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_013301505844518912251 8 2_shared/overview

2. 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_012605268423008256169 2 _shared/overview

*PMC609: 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 Issues 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. NAPCC-GoI Initiatives.

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

TEXT BOOKS:

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

REFERENCE BOOKS:

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

PCS701PC: CRYPTOGRAPHY AND NETWORK SECURITY

B.Tech. IV Year I Sem.   L T P C
                                                                                        3 0 0 3

Course Objectives:
- Explain the importance and application of each of confidentiality, integrity, authentication and availability
- Understand various cryptographic algorithms.
- Understand the basic categories of threats to computers and networks
- Describe public-key cryptosystem.
- Describe the enhancements made to IPv4 by IPSec
- Understand Intrusions and intrusion detection

Course Outcomes:
- Student will be able to understand basic cryptographic algorithms, message and web authentication and security issues.
- Ability to identify information system requirements for both of them such as client and server.
- Ability to understand the current legal issues towards information security.

UNIT - I
Security Concepts: Introduction, The need for security, Security approaches, Principles of security, Types of Security attacks, Security services, Security Mechanisms, A model for Network Security
Cryptography Concepts and Techniques: Introduction, plain text and cipher text, substitution techniques, transposition techniques, encryption and decryption, symmetric and asymmetric key cryptography, steganography, key range and key size, possible types of attacks.

UNIT - II
Symmetric key Ciphers: Block Cipher principles, DES, AES, Blowfish, RC5, IDEA, Block cipher operation, Stream ciphers, RC4.
Asymmetric key Ciphers: Principles of public key cryptosystems, RSA algorithm, Elgamal Cryptography, Diffie-Hellman Key Exchange, Knapsack Algorithm.

UNIT - III
Cryptographic Hash Functions: Message Authentication, Secure Hash Algorithm (SHA-512),
Message authentication codes: Authentication requirements, HMAC, CMAC, Digital signatures, Elgamal Digital Signature Scheme.
Key Management and Distribution: Symmetric Key Distribution Using Symmetric & Asymmetric Encryption, Distribution of Public Keys, Kerberos, X.509 Authentication Service, Public – Key Infrastructure

UNIT - IV
Transport-level Security: Web security considerations, Secure Socket Layer and Transport Layer Security, HTTPS, Secure Shell (SSH)
Wireless Network Security: Wireless Security, Mobile Device Security, IEEE 802.11 Wireless LAN, IEEE 802.11i Wireless LAN Security

UNIT - V
E-Mail Security: Pretty Good Privacy, S/MIME IP Security: IP Security overview, IP Security architecture, Authentication Header, Encapsulating security payload, Combining security associations, Internet Key Exchange
Case Studies on Cryptography and security: Secure Multiparty Calculation, Virtual Elections, Single sign On, Secure Inter-branch Payment Transactions, Cross site Scripting Vulnerability.

TEXT BOOKS:

1. Cryptography and Network Security - Principles and Practice: William Stallings, Pearson Education, 6th Edition
2. Cryptography and Network Security: Atul Kahate, Mc Graw Hill, 3rd Edition

REFERENCE BOOKS:

1. Cryptography and Network Security: C K Shyamala, N Harini, Dr T R Padmanabhan, Wiley India, 1st Edition.
2. Cryptography and Network Security: Forouzan Mukhopadhyay, Mc Graw Hill, 3rd Edition
3. Information Security, Principles, and Practice: Mark Stamp, Wiley India.
4. Principles of Computer Security: WM. Arthur Conklin, Greg White, TMH
5. Introduction to Network Security: Neal Krawetz, CENGAGE Learning
6. Network Security and Cryptography: Bernard Menezes, CENGAGE Learning

PCS702PC: COMPILER DESIGN

B.Tech. IV Year I Sem.   L T P C
                                                                                                                         3 0 0 3

Prerequisites
1. A course on "Formal Languages and Automata Theory".
2. A course on "Computer Organization and architecture".
3. A course on "Data Structures".

Course Objectives:
- 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, code optimization techniques, intermediate code generation, code generation and data flow analysis.

Course Outcomes:
- Demonstrate the ability to design a compiler given a set of language features.
- Demonstrate the knowledge of patterns, tokens & regular expressions for lexical analysis.
- Acquire skills In using lex tool & yacc tool for developing a scanner and parser.
- Design and implement LL and LR parsers
- Design algorithms to do code optimization in order to improve the performance of a program in terms of space and time complexity.
- Design algorithms to generate machine code.

UNIT - I
Introduction: The structure of a compiler, the science of building a compiler, programming language basics
Lexical Analysis: The Role of the Lexical Analyzer, Input Buffering, Recognition of Tokens, The Lexical-Analyzer Generator Lex, Finite Automata, From Regular Expressions to Automata, Design of a Lexical-Analyzer Generator, Optimization of DFA-Based Pattern Matchers.

UNIT - II
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, Using Ambiguous Grammars and Parser Generators.

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

UNIT - IV
Run-Time Environments: Stack Allocation of Space, Access to Nonlocal Data on the Stack, Heap Management, Introduction to Garbage Collection, Introduction to Trace-Based Collection.
Code Generation: Issues in the Design of a Code Generator, The Target Language, Addresses in the Target Code, Basic Blocks and Flow Graphs, Optimization of Basic Blocks, A Simple Code Generator, Peephole Optimization, Register Allocation and Assignment, Dynamic Programming Code-Generation

UNIT - V

Machine-Independent Optimization: The Principal Sources of Optimization, Introduction to Data-Flow Analysis, Foundations of Data-Flow Analysis, Constant Propagation, Partial-Redundancy Elimination, Loops in Flow Graphs.

TEXT BOOK:

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

REFERENCE BOOKS:

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

PCS741PE: GRAPH THEORY (Professional Elective – IV)

B.Tech. IV Year I Sem.   L T P C
                                                                                3 0 0 3

Course Objectives:
- Understanding graphs, trees, connected paths, applications of trees and graphs.

Course Outcomes:
- Know some important classes of graph theoretic problems;
- Prove central theorems about trees, matching, connectivity, coloring and planar graphs;
- Describe and apply some basic algorithms for graphs;
- Use graph theory as a modeling tool.

UNIT - I
Introduction-Discovery of graphs, Definitions, Subgraphs, Isomorphic graphs, Matrix representations of graphs, Degree of a vertex, Directed walks, paths and cycles, Connectivity in digraphs, Eulerian and Hamilton digraphs, Eulerian digraphs, Hamilton digraphs, Special graphs, Complements, Larger graphs from smaller graphs, Union, Sum, Cartesian Product, Composition, Graphic sequences, Graph theoretic model of the LAN problem, Havel-Hakimi criterion, Realization of a graphic sequence.

UNIT - II
Connected graphs and shortest paths - Walks, trails, paths, cycles, Connected graphs, Distance, Cut-vertices and cut-edges, Blocks, Connectivity, Weighted graphs and shortest paths, Weighted graphs, Dijkstra"s shortest path algorithm, Floyd-Warshall shortest path algorithm.

UNIT - III
Trees- Definitions and characterizations, Number of trees, Cayley"s formula, Kircho-matrix-tree theorem, Minimum spanning trees, Kruskal"s algorithm, Prim"s algorithm, Special classes of graphs, Bipartite Graphs, Line Graphs, Chordal Graphs, Eulerian Graphs, Fleury"s algorithm, Chinese Postman problem, Hamilton Graphs, Introduction, Necessary conditions and sufficient conditions.

UNIT - IV
Independent sets coverings and matchings- Introduction, Independent sets and coverings: basic equations, Matchings in bipartite graphs, Hall"s Theorem, K¨onig"s Theorem, Perfect matchings in graphs, Greedy and approximation algorithms.

UNIT - V
Vertex Colorings- Basic definitions, Cliques and chromatic number, Mycielski"s theorem, Greedy coloring algorithm, Coloring of chordal graphs, Brooks theorem, Edge Colorings, Introduction and Basics, Gupta-Vizing theorem, Class-1 and Class-2 graphs, Edge-coloring of bipartite graphs, Class-2 graphs, Hajos union and Class-2 graphs, A scheduling problem and equitable edge-coloring.

TEXT BOOKS:
1. J. A. Bondy and U. S. R. Murty. Graph Theory, volume 244 of Graduate Texts in Mathematics. Springer, 1st edition, 2008.
2. J. A. Bondy and U. S. R. Murty. Graph Theory with Applications.

REFERENCE BOOKS:
1. Lecture Videos: http://nptel.ac.in/courses/111106050/13
2. Introduction to Graph Theory, Douglas B. West, Pearson.

PCS742PE: CYBER SECURITY (Professional Elective – IV)

B.Tech. IV Year I Sem.   **L T P C**
                                                              3 0 0 3

Course objectives:
- To understand various types of cyber-attacks and cyber-crimes.
- To learn threats and risks within the context of cyber security.
- To have an overview of the cyber laws & concepts of cyber forensics.
- To study the defensive techniques against these attacks.

Course Outcomes:
1. Analyze and evaluate the cyber security needs of an organization.
2. Understand Cyber Security Regulations and Roles of International Law.
3. Design and develop security architecture for an organization.
4. Understand fundamental concepts of data privacy attacks.

UNIT- I

Introduction to Cyber Security: Basic Cyber Security Concepts, layers of security, Vulnerability, threat, Harmful acts, Internet Governance – Challenges and Constraints, Computer Criminals, CIA Triad, Assets and Threat, motive of attackers, active attacks, passive attacks, Software attacks, hardware attacks, Cyber Threats-Cyber Warfare, Cyber Crime, Cyber terrorism, Cyber Espionage, etc., Comprehensive Cyber Security Policy.

UNIT - II

Cyberspace and the Law & Cyber Forensics: Introduction, Cyber Security Regulations, Roles of International Law. The INDIAN Cyberspace, National Cyber Security Policy.
Introduction, Historical background of Cyber forensics, Digital Forensics Science, The Need for Computer Forensics, Cyber Forensics and Digital evidence, Forensics Analysis of Email, Digital Forensics Lifecycle, Forensics Investigation, Challenges in Computer Forensics.

UNIT - III

Cybercrime: Mobile and Wireless Devices: Introduction, Proliferation of Mobile and Wireless Devices, Trends in Mobility, Credit card Frauds in Mobile and Wireless Computing Era, Security Challenges Posed by Mobile Devices, Registry Settings for Mobile Devices, Authentication service Security, Attacks on Mobile/Cell Phones, Organizational security Policies and Measures in Mobile Computing Era, Laptops.

UNIT- IV

Cyber Security: Organizational Implications: Introduction, cost of cybercrimes and IPR issues, web threats for organizations, security and privacy implications, social media marketing: security risks and perils for organizations, social computing and the associated challenges for organizations.

UNIT - V

Privacy Issues: Basic Data Privacy Concepts: Fundamental Concepts, Data Privacy Attacks, Data linking and profiling, privacy policies and their specifications, privacy policy languages, privacy in different domains- medical, financial, etc.
Cybercrime: Examples and Mini-Cases
Examples: Official Website of Maharashtra Government Hacked, Indian Banks Lose Millions of Rupees, Parliament Attack, Pune City Police Bust Nigerian Racket, e-mail spoofing instances.
Mini-Cases: The Indian Case of online Gambling, An Indian Case of Intellectual Property Crime, Financial Frauds in Cyber Domain.

TEXT BOOKS:
1. Nina Godbole and Sunit Belpure, Cyber Security Understanding Cyber Crimes, Computer Forensics and Legal Perspectives, Wiley

REFERENCE BOOKS:
1. B. B. Gupta, D.P. Agrawal, Haoxiang Wang, Computer and Cyber Security: Principles, Algorithm, Applications, and Perspectives, CRC Press
2. Cyber Security Essentials, James Graham, Richard Howard and Ryan Otson, CRC Press.
3. Introduction to Cyber Security, Chwan-Hwa(john) Wu, J.David Irwin, CRC Press T&F Group.

PCS743PE: SOFT COMPUTING (Professional Elective – IV)

B.Tech. IV Year I Sem.

L T P C
3 0 0 3

Course Objectives:
- Familiarize with soft computing concepts
- Introduce and use the idea of fuzzy logic and use of heuristics based on human experience
- Familiarize the Neuro-Fuzzy modeling using Classification and Clustering techniques
- Learn the concepts of Genetic algorithm and its applications
- Acquire the knowledge of Rough Sets.

Course Outcomes:
- Identify the difference between Conventional Artificial Intelligence to Computational Intelligence.
- Understand fuzzy logic and reasoning to handle and solve engineering problems
- Apply the Classification techniques on various applications.
- Perform various operations of genetic algorithms and Rough Sets.

UNIT - I
Introduction to Soft Computing: Evolutionary Computing, "Soft" computing versus "Hard" computing, Soft Computing Methods, Recent Trends in Soft Computing, Characteristics of Soft computing, Applications of Soft Computing Techniques.

UNIT- II
Fuzzy Systems: Fuzzy Sets, Fuzzy Relations, Fuzzy Logic, Fuzzy Rule-Based Systems

UNIT- III
Fuzzy Decision Making, Particle Swarm Optimization

UNIT- IV
Genetic Algorithms: Basic Concepts, Basic Operators for Genetic Algorithms, Crossover and Mutation Properties, Genetic Algorithm Cycle, Fitness Function, Applications of Genetic Algorithm.

UNIT- V
Rough Sets, Rough Sets, Rule Induction, and Discernibility Matrix, Integration of Soft Computing Techniques.

TEXT BOOK:
1. Soft Computing - Advances and Applications - Jan 2015 by B.K. Tripathy and J. Anuradha - Cengage Learning

REFERENCE BOOKS:
1. S. N. Sivanandam & S. N. Deepa, "Principles of Soft Computing", 2nd edition, Wiley India, 2008.
2. David E. Goldberg, "Genetic Algorithms-In Search, optimization and Machine learning", Pearson Education.
3. J. S. R. Jang, C.T. Sun and E.Mizutani, "Neuro-Fuzzy and Soft Computing", Pearson Education, 2004.
4. G.J. Klir & B. Yuan, "Fuzzy Sets & Fuzzy Logic", PHI, 1995.
5. Melanie Mitchell, "An Introduction to Genetic Algorithm", PHI, 1998.
6. Timothy J. Ross, "Fuzzy Logic with Engineering Applications", McGraw- Hill International editions, 1995

PCS744PE: CLOUD COMPUTING (Professional Elective – IV)

B.Tech. IV Year I Sem.   L T P C
                                                                     3 0 0 3

Pre-requisites:
   1. A course on "Computer Networks".
   2. A course on "Operating System".

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

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

UNIT - I
Computing Paradigms, Cloud Computing Fundamentals, Cloud Computing Architecture and Management

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

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

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

UNIT - V
Security in Cloud Computing, and Advanced Concepts in Cloud Computing

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

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

PCS745PE: AD-HOC & SENSOR NETWORKS (Professional Elective – IV)

B.Tech. IV Year I Sem.

L T P C
3 0 0 3

Prerequisites
1. Computer Networks
2. Distributed Systems
3. Mobile Computing

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

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

UNIT - I
Introduction to Ad Hoc Networks
Characteristics of MANETs, Applications of MANETs and Challenges of MANETs.
Routing in MANETs
Criteria for classification, Taxonomy of MANET routing algorithms, *Topology-based* routing algorithms-Proactive: DSDV, WRP; Reactive: DSR, AODV, TORA; Hybrid: ZRP; *Position- based* routing algorithms-Location Services-DREAM, Quorum-based, GLS; Forwarding Strategies, Greedy Packet, Restricted Directional Flooding-DREAM, LAR; Other routing algorithms-QoS Routing, CEDAR.

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

UNIT - III
Geocasting
Data-transmission Oriented-LBM; Route Creation Oriented-GeoTORA, MGR.
TCP over Ad Hoc TCP protocol overview, TCP and MANETs, Solutions for TCP over Ad hoc

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

UNIT - V
Upper Layer Issues of WSN
Transport layer, High-level application layer support, Adapting to the inherent dynamic nature of WSNs.

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

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

PCS751PE: ADVANCED ALGORITHMS (Professional Elective – V)

B.Tech. IV Year I Sem.   L T P C
                                                        3 0 0 3

Pre-Requisites: Algorithm Design and Analysis

Course Objectives:
- To familiarize advanced methods on analysis of algorithms.
- To familiarize with graphs and algorithms related shortest path
- To understand matrix computations and modulo representations
- To introduce randomized, approximation algorithms and computational complexity topics

Course Outcomes:
- Familiarize with advanced methods on analysis of algorithms
- Familiarize with the graphs, graph matching and shortest path algorithms
- Understand matrix computations and modulo representations
- Understand randomized, approximation algorithms and computational complexity topics

UNIT – I

Introduction to Algorithms, Classification of Algorithms, Asymptotic Analysis, Introduction to Recurrence equations - Linear recurrences, Non-linear recurrences, Formulation of recurrence equations, techniques for solving recurrence equations, Solving recurrence equations using polynomial reduction, Master's theorem

Graph: Definitions and Elementary Algorithms: Shortest path by BFS, shortest path in edge-weighted case (Dijkstra's), depth-first search and computation of strongly connected components, Multistage Graph, topological sorting

UNIT – II

Graph Matching: Algorithm to compute maximum matching. Characterization of maximum matching by augmenting paths, Edmond's Blossom algorithm to compute augmenting path, Bipartite matching problem

Matroids: Introduction to greedy paradigm, algorithm to compute a maximum weight maximal independent set, Optimal tree problems- optimal merge, huffman coding, tree vertex splitting problem.

Shortest Path in Graphs: Floyd-Warshall algorithm, Travelling Sales Person Problem and introduction to dynamic programming paradigm. Optimal Graph Problems - Minimum Spanning Tree, Single source shortest path.

UNIT - III

Flow-Networks: Maxflow - mincut theorem, Ford-Fulkerson Method to compute maximum flow, Edmond-Karp maximum-flow algorithm.

Matrix Computations: Strassen's algorithm and introduction to divide and conquer paradigm, Chain Matrix Multiplication, Matrix operations - Gaussian Elimination method, LUP-decomposition, Crout's method of decomposition, inverse of a triangular matrix,

UNIT - IV

Modulo Representation of integers/polynomials: Chinese Remainder Theorem, Conversion between base-representation and modulo-representation, interpolation problem. Multiplication of long integers by using Divide and Conquer paradigm, Schonhage-Strassen's Integer Multiplication algorithm.

String Algorithms: Naïve String, Rabin Karp, KMP, Boyer Moore, Harspool algorithms

UNIT - V
Basics of Computational Complexity: Introduction to computational complexity, complexity classes, Satisfiability problem and Cook's theorem, Examples of NP- Complete problems
Randomized algorithms: Introduction, Types of Randomized algorithms, Example of Randomized algorithms.
Approximation algorithms: Introduction, Types of Approximation algorithms, Examples of Approximation algorithms

TEXT BOOK:
1. Design and Analysis of Algorithms, S. Sridhar, Oxford University Press.

REFERENCE BOOKS:
1. Introduction to Algorithms, Cormen, Leiserson, Rivest, Stein.
2. The Design and Analysis of Computer Algorithms, Aho, Hopcroft, Ullman.
3. Algorithm Design, Kleinberg and Tardos.

PCS752PE: AGILE METHODOLOGY (Professional Elective – V)

B.Tech. IV Year I Sem.   **L T P C**
                                                                       3 0 0 3

Course Objectives:
- Knowledge on concepts of agile development, releasing, planning and developing

Course Outcomes:
- Identify basic concepts of agile methodology and Extreme programming
- Analyze real customer involvement in collaboration
- Discuss risk management and iteration planning
- Understanding incremental requirements, refactoring, incremental design and architecture

UNIT - I
Introduction Extreme Programming (XP) - Agile Development
Why Agile?, Understanding Success, Beyond Deadlines, Importance of Organizational Success, Introduction to Agility, Agile methods-Scrum and XP, Manifesto for Agile Software Development, Principles of Agile Process. Understanding XP (Extreme Programming) - XP life cycle, XP team, XP Concepts, Adopting XP - Knowing whether XP is suitable, Implementing XP, assessing Agility, Practicing XP - Thinking, Pair Programming, Energized work, Informative Workspace, Root cause Analysis, Retrospectives.

UNIT - II
Collaborating
Trust, Sit together, Real customer involvement, Ubiquitous language, Stand-Up meetings, coding standards, Iteration demo, Reporting.

UNIT - III
Releasing
Bugfree Release, Version Control, Ten-Minute Build, continuous integration, Collective ownership and Documentation.

UNIT – IV
Planning
Version, Release Planning, The Planning Game, Risk Management, Iteration Planning, Slack, Stories, and Estimating

UNIT - V
Developing
Incremental requirements, Customer tests, Test driven development, Refactoring, Incremental design and architecture, spike solutions, Performance optimization, Exploratory testing.

TEXT BOOK:
1. The art of Agile Development, James Shore and Shane Warden, 11th Indian Reprint, O'Reilly, 2018.

REFERENCE BOOKS:
1. Learning Agile, Andrew Stellman and Jennifer Greene, O'Reilly, 4th Indian Reprint, 2018
2. Practices of an Agile Developer, Venkat Subramaniam and Andy Hunt, SPD, 5th Indian Reprint, 2015
3. Agile Project Management - Jim Highsmith, Pearson Low price Edition 2004

PCS753PE: ROBOTIC PROCESS AUTOMATION (Professional Elective – V)

B.Tech. IV Year I Sem.   L T P C
                                                                               3 0 0 3

Course Objectives:
- Introduce robotic process automation, techniques of automation using UIPath RPA tool.

Course Outcomes:
- Understand the concepts of Robotic Process Automation.
- Apply the flow chart mechanism in various calculations.
- Applying UIPath tool for debugging process
- Design system managing techniques.
- Create application for process automation using UIPath tool.

UNIT - I

Robotic Process Automation: Introduction, Scope and techniques of automation, Robotic process automation, Components of RPA, RPA platforms, About UiPath
UIPath Stack Uipath Studio, Uipath Robot, Types of Robots, UiPath Orchestrator
UIPath Studio Projects, User interface
The User Interface: Task recorder, Advanced UI interactions: Input methods, Output methods

UNIT - II

Sequence, Flowchart, and Control Flow: Sequencing the workflow, Activities, Control Flow, various types of loops and decision making
Data Manipulation: Variables and scope, Collections, Arguments - Purpose and use, Data table usage with examples, File operation with step-by-step example, CSV/Excel to data table and vice versa

UNIT - III

Taking Control of the Controls: Finding and attaching windows, Finding the control, Techniques for waiting for a control, Act on controls – mouse and keyboard activities, Handling events, revisit recorder, When to use OCR, Types of OCR available, How to use OCR
Plugins and Extensions: Terminal Plugin, SAP Automation, Citrix automation and Credential management

UNIT - IV

Handling User Events and Assistant Bots: Assistant bots, Monitoring system event triggers, Monitoring image and element triggers, Launching an assistant bot on a keyboard event
Exception Handling, Debugging, and Logging: Exception handling, Common exceptions and ways to handle them, Logging and taking screenshots, Debugging techniques, Collecting crash dumps, Error reporting

UNIT - V

Managing and Maintaining the Code: Project organization, nesting workflows, Reusability of workflows, Commenting techniques, State Machine, When to use Flowcharts, State Machines, or Sequences, Using config files
Deploying and Maintaining the Bot: Publishing using publish utility, using Orchestration Server to control bots, deploy bots, License Management, Publishing and Managing updates

TEXT BOOK:
1. Learning Robotic Process Automation: Create Software robots and automate business processes with the leading RPA tool - UiPath: Create Software robots. with the leading RPA tool – UiPath Kindle Edition

REFERENCE BOOK:
1. Robotic Process Automation A Complete Guide - 2020 Edition Kindle Edition.

PCS754PE: BLOCKCHAIN TECHNOLOGY (Professional Elective – V)

B.Tech. IV Year I Sem.

L T P C
3 0 0 3

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

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

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

UNIT-I

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

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

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

UNIT-II

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

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

UNIT-III

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

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

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

UNIT-IV

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

Page **108** of 118

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

UNIT-V
Blockchain Case Studies: Case Study 1 - Retail, Case Study 2 - Banking and Financial Services, Case Study 3 - Healthcare, Case Study 4 - Energy and Utilities.
Blockchain Platform using Python: Introduction, Learn How to Use Python Online Editor, Basic Programming Using Python, Python Packages for Blockchain.
Blockchain platform using Hyperledger Fabric: Introduction, Components of Hyper ledger Fabric Network, Chain codes from Developer.ibm.com, Blockchain Application Using Fabric Java SDK.

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

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

PCS755PE: SOFTWARE PROCESS & PROJECT MANAGEMENT (Professional Elective – V)

B.Tech. IV Year I Sem.   L T P C
                                                                             3 0 0 3

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

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

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

UNIT - II
Software Project Management Renaissance
Conventional Software Management, Evolution of Software Economics, Improving Software Economics, Life-Cycle Phases and Process artifacts
Engineering and Production stages, inception phase, elaboration phase, construction phase, transition phase, artifact sets, management artifacts, engineering artifacts and pragmatic artifacts, model-based software architectures.

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

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

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

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

REFERENCE BOOKS:
1. An Introduction to the Team Software Process, Watts S. Humphrey, Pearson Education, 2000

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

PCS721OE: OPERATING SYSTEMS (Open Elective –II)

B.Tech. IV Year I Sem.   L T P C
                                                                                        3 0 0 3

Prerequisites:
1. A course on "Computer Programming and Data Structures".
2. A course on "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
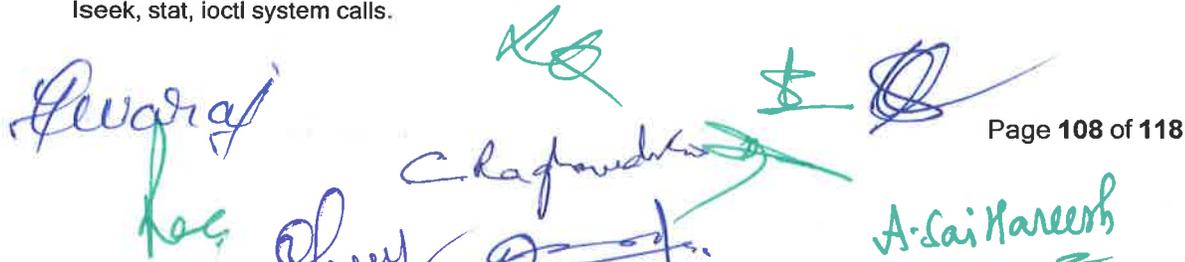**Interprocess Communication Mechanisms:** IPC between processes on a single computer system, IPC between processes on different systems, using pipes, FIFOs, message queues, shared memory.

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.

PCS722OE: SOFTWARE ENGINEERING (Open Elective –II)

B.Tech. IV 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 Rambaugh, 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.

PCS703PC: CRYPTOGRAPHY AND NETWORK SECURITY LAB

B.Tech. IV Year I Sem.   **L T P C**
                                              0 0 2 1

Course Objectives:
- Explain the objectives of information security
- Explain the importance and application of each of confidentiality, integrity, authentication and availability
- Understand various cryptographic algorithms.

Course Outcomes:
- Understand basic cryptographic algorithms, message and web authentication and security issues.
- Identify information system requirements for both of them such as client and server.
- Understand the current legal issues towards information security.

List of Experiments:
1. Write a C program that contains a string (char pointer) with a value 'Hello world'. The program should XOR each character in this string with 0 and display the result.
2. Write a C program that contains a string (char pointer) with a value 'Hello world'. The program should AND or and XOR each character in this string with 127 and display the result.
3. Write a Java program to perform encryption and decryption using the following algorithms
       a. Ceaser cipher b. Substitution cipher c. Hill Cipher
4. Write a C/JAVA program to implement the DES algorithm logic.
5. Write a C/JAVA program to implement the Blowfish algorithm logic.
6. Write a C/JAVA program to implement the Rijndael algorithm logic.
7. Write the RC4 logic in Java Using Java cryptography; encrypt the text "Hello world" using Blowfish. Create your own key using Java key tool.
8. Write a Java program to implement the RSA algorithm.
9. Implement the Diffie-Hellman Key Exchange mechanism using HTML and JavaScript.
10. Calculate the message digest of a text using the SHA-1 algorithm in JAVA.
11. Calculate the message digest of a text using the MD5 algorithm in JAVA

TEXT BOOKS:
1. Cryptography and Network Security - Principles and Practice: William Stallings, Pearson Education, 6th Edition
2. Cryptography and Network Security: Atul Kahate, McGraw Hill, 3rd Edition

REFERENCE BOOKS:
1. Cryptography and Network Security: C K Shyamala, N Harini, Dr T R Padmanabhan, Wiley India, 1st Edition.
2. Cryptography and Network Security: Forouzan Mukhopadhyay, McGraw Hill, 3rd Edition
3. Information Security, Principles, and Practice: Mark Stamp, Wiley India.
4. Principles of Computer Security: WM. Arthur Conklin, Greg White, TMH
5. Introduction to Network Security: Neal Krawetz, CENGAGE Learning
6. Network Security and Cryptography: Bernard Menezes, CENGAGE Learning

PCS704PC: COMPILER DESIGN LAB

B.Tech. IV Year I 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->E+T E->T T->T*F T->F
    F->(E)/id.
11. write recursive descent parser for the grammar S->(L)   S->a
L->L,S L->S
12. Write a C program to calculate first function for the grammar
        E->E+T E->T T->T*F T->F   F->(E)/id
13. Write a YACC program to implement a top down parser for the given grammar.
13. 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, Jeffry D. Ullman.

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

PCS801PC: ORGANIZATIONAL BEHAVIOUR

B.Tech. IV Year II Sem.

L T P C
3 0 0 3

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

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

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

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

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

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

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

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

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

PCS861PE: COMPUTATIONAL COMPLEXITY (Professional Elective – VI)

B.Tech. IV Year II Sem.	L	T	P	C
	3	0	0	3

Prerequisite: Design and Analysis of Algorithms.

Course Objectives:
- Introducing computational complexity-based algorithms and their implementations

Course Outcomes:
- Understand the complexity of time and space for computational models
- Understand optimizational problems
- Understand NP completeness problems
- Understand hierarchical theorems

UNIT – I

Introduction: Algorithms and complexity, Basic Complexity Classes-Deterministic time and the class P. Computational Tasks and models: Computational tasks – Search problems, Decision problems, Uniform models- Overview, General Principles, Concrete Model, Halting problem, restricted models.

UNIT – II

P vs. NP: Efficient Computation, The Search Version (Finding vs. Checking), The Decision Version (Proving Vs Verifying), Equivalence of the two formulations, Optimal Search Algorithms for NP Polynomial time reduction: The general notation of a Reduction, Reducing Optimization Problems to search problems, Self-Reducibility of search problems

UNIT – III

NP - Completeness: Definition, Cook's theorem, Existence of NP Complete Problems bounded halting and non-halting, Natural NP Complete Problems – The NP completeness of CSAT, The NP Completeness of SAT, Combinatorics and Graph Theory, additional properties of the standard reductions, Negative applications of NP Completeness, Positive applications of NP Completeness, NP Sets, Reflections on Complete problems, NP -complete optimization problems.

UNIT –IV

Diagonalization: Time Hierarchy theorem, Space Hierarchy theorem, Non-deterministic Time Hierarchy theorem, Ladner's theorem.

Space Complexity: Definition of space bounded computation, PSPACE completeness, NL Completeness, some space complexity classes- Savitch's theorem, Savitch's theorem, The essence of PSPACE

The polynomial time hierarchy and alternations: polynomial hierarchy, time versus alternations, properties of polynomial hierarchy, Complete problems in PH.

UNIT – V

Randomized computation: Probabilistic Turing machine, one sided and zero-sided error, Randomized reduction, Randomized space bounded computation.

Decision trees: Graphs and Decision Trees, Monotonic Graph properties, Topological criterion, Randomized decision trees.

TEXT BOOKS:
1. The Basics of Computational Complexity, Oded Goldreich, Cambridge University Press
2. Computational Complexity: A Modern Approach, Sanjeev Arora and Boaz Barak, Princeton University

REFERENCE BOOKS:
1. Computational Complexity, by Christos Papadimitriou
2. Theory of Computational Complexity, Ding-Zhu Du, Ker-I Ko, WILEY

PCS802PE: DISTRIBUTED SYSTEMS (Professional Elective –VI)

B.Tech. IV Year II Sem.   L T P C
                                                                                 3 0 0 3

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

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

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

UNIT - I
Characterization of Distributed Systems: Examples of Distributed systems, Resource sharing and web, challenges
System models: Architectural and Fundamental models, Networking and Internetworking, Interprocess Communication
Distributed objects and Remote Invocation: Communication between distributed objects, RPC, Events and notifications, Case study-Java RMI.

UNIT - II
Operating System Support- OS layer, Protection, Processes and Threads, Communication and Invocation, Operating system architecture.
Distributed File Systems-Introduction, File Service architecture.

UNIT - III
Peer to Peer Systems- Napster and its legacy, Peer to Peer middleware
Time and Global States-Introduction, Clocks, events and Process states, Synchronizing physical clocks, logical time and logical clocks, global states, distributed debugging.
Coordination and Agreement- Distributed mutual exclusion, Elections, Multicast communication, consensus and related problems.

UNIT - IV
Transactions and Concurrency Control- Introduction, Transactions, Nested Transactions, Locks, Optimistic concurrency control, Timestamp ordering.
Distributed Transactions-Introduction, Flat and Nested Distributed Transactions, Atomic commit protocols, Concurrency control in distributed transactions
Distributed deadlocks: Transaction recovery.

UNIT - V
Replication: Introduction, System model and group communication, Fault tolerant services, Transactions with replicated data.
Distributed shared memory: Design and Implementation issues, Consistency models.

TEXT BOOKS:
1. Distributed Systems Concepts and Design, G Coulouris, J Dollimore and T Kindberg, Fourth Edition, Pearson Education.

2. Distributed Systems, S. Ghosh, Chapman & Hall/CRC, Taylor & Francis Group, 2010.

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

PCS863PE: DEEP LEARNING (Professional Elective –VI)

B.Tech. IV Year II Sem.

L	T	P	C
3	0	0	3

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

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

UNIT -I
Machine Learning Basics

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

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

UNIT -II
Regularization for Deep Learning

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

UNIT-III
Convolutional Networks

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

UNIT -IV
Recurrent and Recursive Nets

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

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

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

TEXT BOOK:

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

REFERENCE BOOKS:

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

PCS864PE: HUMAN COMPUTER INTERACTION (Professional Elective –VI)

B.Tech. IV Year II Sem.   L T P C
                                                                                              3 0 0 3

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

Course Outcomes:
- Apply HCI and principles to interaction design.
- Design certain tools for blind or PH people
- Understand the social implications of technology and ethical responsibilities as engineers.
- Understand the importance of a design and evaluation methodology

UNIT - I
Introduction: Importance of user Interface - definition, importance of good design. Benefits of good design, A brief history of Screen design.
The graphical user interface – popularity of graphics, the concept of direct manipulation, graphical system, Characteristics, Web user - Interface popularity, characteristics- Principles of user interface.

UNIT - II
Design process - Human interaction with computers, importance of human characteristics, human consideration, Human interaction speeds, understanding business junctions.
Screen Designing: Design goals - Screen planning and purpose, organizing screen elements, ordering of screen data and content - screen navigation and flow - Visually pleasing composition - amount of information - focus and emphasis - presentation information simply and meaningfully - information retrieval on web - statistical graphics - Technological consideration in interface design.

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

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

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

TEXT BOOKS:

1. The essential guide to user interface design, Wilbert O Galitz, Wiley Dream Tech.
2. Human - Computer Interaction. Alan Dix, Janet Fincay, Gregory's, Abowd, Russell Bealg, Pearson Education.

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

PCS865PE: CYBER FORENSICS (Professional Elective –VI)

B.Tech. IV Year II Sem.   L T P C
                                                                    3 0 0 3

Prerequisites: Network Security.
Course Objectives:

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

Course Outcomes:

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

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

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

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

UNIT- IV
Current Forensic tools: evaluating computer forensic tool needs, computer forensics software tools, computer forensics hardware tools, validating and testing forensics software E-Mail Investigations: Exploring the role of e-mail in investigation, exploring the roles of the client and server in e-mail, investigating e-mail crimes and violations, understanding e-mail servers, using specialized e-mail forensic tools.
Cell phone and mobile device forensics: Understanding mobile device forensics, understanding acquisition procedures for cell phones and mobile devices.

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

TEXT BOOKS:
1. Kevin Mandia, Chris Prosise, "Incident Response and computer forensics", Tata McGraw Hill, 2006.

2. Computer Forensics, Computer Crime Investigation by John R. Vacca, Firewall Media, New Delhi.
3. Computer Forensics and Investigations by Nelson, Phillips Enfinger, Steuart, CENGAGE Learning

REFERENCE BOOKS:

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

A·Sai Hareerh

PCS831OE: ALGORITHMS DESIGN AND ANALYSIS (Open Elective –III)

B.Tech. IV 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.

PCS832OE: INTRODUCTION TO COMPUTER NETWORKS (Open Elective – III)

B.Tech. IV Year II Sem.

L	T	P	C
3	0	0	3

Prerequisites
1. A course on "Programming for problem solving"
2. A course on "Data Structures"

Course Objectives
- Equip the students with the concepts and fundamentals of computer networks.
- Familiarize the students with the standard models for the layered approach to communication between machines in a network and the protocols of the various layers.

Course Outcomes
1. Gain the knowledge of the basic computer network technology.
2. Gain the knowledge of the functions of each layer in the OSI and TCP/IP reference model.
3. Understand subnetting and routing mechanisms.
4. Familiarity with the essential application protocols of computer networks

UNIT - I
Network hardware, Network software, OSI, TCP/iP Reference models, Example Networks: ARPANET, Internet.

Physical Layer: Guided Transmission media: twisted pairs, coaxial cable, fiber optics, Wireless Transmission.

Data link layer: Design issues, framing, Error detection and correction.

UNIT - II
Elementary data link protocols: simplex protocol, A simplex stop and wait protocol for an error-free channel, A simplex stop and wait protocol for noisy channel.

Sliding Window protocols: A one-bit sliding window protocol, A protocol using Go-Back-N, A protocol using Selective Repeat, Example data link protocols.

Medium Access sub layer: The channel allocation problem, Multiple access protocols: ALOHA, Carrier sense multiple access protocols, collision free protocols, Data link layer switching.

UNIT - III
Network Layer: Design issues, Routing algorithms: shortest path routing, Flooding, Hierarchical routing, Broadcast, Multicast, distance vector routing, Congestion Control Algorithms, Quality of Service, Internetworking

UNIT - IV
Transport Layer: Transport Services, Elements of Transport protocols, Connection management, TCP and UDP protocols.

UNIT - V
Application Layer: Domain name system, Electronic Mail; the World WEB, HTTP, Streaming audio and video.

TEXT BOOK:
1. Computer Networks -- Andrew S Tanenbaum, David. j. Wetherall, 6th Edition. Pearson Education

REFERENCE BOOKS:
1. An Engineering Approach to Computer Networks-S. Keshav, 2nd Edition, Pearson Education
2. Data Communications and Networking - Behrouz A. Forouzan. Third Edition TMH.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Minutes of the Board of Studies (BoS) Meeting

Department: Computer Science and Engineering (CSE)
Date: 25/08/2025,   **Time:** 11:00 AM
Venue: Board Room, Pallavi Engineering College, Kuntloor, Hyderabad - 501505

Dr. A. Sai Hareesh, HOD of the CSE Department and Chairperson of the Board of Studies, welcomed all the BoS members. He provided a brief overview of the agenda and initiated the meeting by presenting a comprehensive report on the autonomous syllabus for the year 2025-2026.

PR25 from Academic year 2025-26 Onwards

S. No.	Couse Code	Name of the Course	Year, Semester (Branch) Common Syllabus for
1	PCS105ES	Programming for Problem Solving	I B.Tech., I Semester Common Syllabus for CSE, CSE-CS, CSE-DS, CSE-AI&ML, ECE, EEE, Civil Engineering
2	PCS107ES	Programming for Problem Solving Lab	I B.Tech., I Semester Common Syllabus for CSE, CSE-CS, CSE-DS, CSE-AI&ML, ECE, EEE, Civil Engineering
3	PCS110ES	IT Workshop	I B.Tech., I Semester Common Syllabus for CSE, CSE-CS, CSE-DS, CSE-AI&ML.
4	PCS204ES	Python Programming	I B.Tech., II Semester Common Syllabus for ECE, EEE, Civil Engineering
5	PCS205ES	Data Structures	I B.Tech., II Semester Common Syllabus for CSE, CSE-CS, CSE-DS, CSE-AI&ML, ECE, EEE
6	PCS207ES	Data Structures Lab	I B.Tech., II Semester Common Syllabus for CSE, CSE-CS, CSE-DS, CSE-AI&ML, ECE, EEE
7	PCS208ES	Python Programming Lab	I B.Tech., II Semester Common Syllabus for CSE, CSE-CS, CSE-DS, CSE-AI&ML, ECE, EEE, Civil Engineering
8	PCS301PC	Discrete Mathematics	II B.Tech., I Semester Syllabus for CSE
9	PCS302PC	Computer Organization and Architecture	II B.Tech., I Semester Common Syllabus for CSE, CSE-DS, CSE-AI&ML
10	PCS303PC	Object Oriented Programming through java	II B.Tech., I Semester Common Syllabus for CSE, CSE-DS, CSE-AI&ML
11	PCS304PC	Software Engineering	II B.Tech., I Semester Common Syllabus for CSE, CSE-CS, CSE-DS, CSE-AI&ML
12	PCS305PC	Data Base Management Systems	II B.Tech., I Semester Common Syllabus for CSE, CSE-CS, CSE-DS, CSE-AI&ML
13	PCS307PC	Object Oriented Programming through java Lab	II B.Tech., I Semester Common Syllabus for CSE, CSE-DS, CSE-AI&ML
14	PCS308PC	Software Engineering Lab	II B.Tech., I Semester Common Syllabus for CSE, CSE-CS, CSE-DS, CSE-AI&ML
15	PCS309PC	Data Base Management Systems Lab	II B.Tech., I Semester Common Syllabus for CSE, CSE-CS, CSE-DS, CSE-AI&ML
16	PCS310PC	Skill Development Course–1(NodeJs /React	II B.Tech., I Semester Common Syllabus for CSE, CSE-DS, CSE-AI&ML

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Minutes of the Board of Studies (BoS) Meeting

Department: Computer Science and Engineering (CSE)

Date: 25/08/2025,   **Time:** 11:00 AM

Venue: Board Room, Pallavi Engineering College, Kuntloor, Hyderabad - 501505

Dr. A. Sai Hareesh, HOD of the CSE Department and Chairperson of the Board of Studies, welcomed all the BoS members. He provided a brief overview of the agenda and initiated the meeting by presenting a comprehensive report on the autonomous syllabus for the year 2025-2026.

PR25 from Academic year 2025-26 Onwards

S. No.	Couse Code	Name of the Course	Year, Semester (Branch) Common Syllabus for
1	PCS105ES	Programming for Problem Solving	I B.Tech., I Semester Common Syllabus for CSE, CSE-CS, CSE-DS, CSE-AI&ML, ECE, EEE, Civil Engineering
2	PCS107ES	Programming for Problem Solving Lab	I B.Tech., I Semester Common Syllabus for CSE, CSE-CS, CSE-DS, CSE-AI&ML, ECE, EEE, Civil Engineering
3	PCS110ES	IT Workshop	I B.Tech., I Semester Common Syllabus for CSE, CSE-CS, CSE-DS, CSE-AI&ML
4	PCS204ES	Python Programming	I B.Tech., II Semester Common Syllabus for ECE, EEE, Civil Engineering
5	PCS205ES	Data Structures	I B.Tech., II Semester Common Syllabus for CSE, CSE-CS, CSE-DS, CSE-AI&ML, ECE, EEE
6	PCS207ES	Data Structures Lab	I B.Tech., II Semester Common Syllabus for CSE, CSE-CS, CSE-DS, CSE-AI&ML, ECE, EEE
7	PCS208ES	Python Programming Lab	I B.Tech., II Semester Common Syllabus for CSE, CSE-CS, CSE-DS, CSE-AI&ML, ECE, EEE, Civil Engineering
8	PCS301PC	Discrete Mathematics	II B.Tech., I Semester Syllabus for CSE
9	PCS302PC	Computer Organization and Architecture	II B.Tech., I Semester Common Syllabus for CSE, CSE-DS, CSE-AI&ML
10	PCS303PC	Object Oriented Programming through java	II B.Tech., I Semester Common Syllabus for CSE, CSE-DS, CSE-AI&ML
11	PCS304PC	Software Engineering	II B.Tech., I Semester Common Syllabus for CSE, CSE-CS, CSE-DS, CSE-AI&ML
12	PCS305PC	Data Base Management Systems	II B.Tech., I Semester Common Syllabus for CSE, CSE-CS, CSE-DS, CSE-AI&ML
13	PCS307PC	Object Oriented Programming through java Lab	II B.Tech., I Semester Common Syllabus for CSE, CSE-DS, CSE-AI&ML
14	PCS308PC	Software Engineering Lab	II B.Tech., I Semester Common Syllabus for CSE, CSE-CS, CSE-DS, CSE-AI&ML
15	PCS309PC	Data Base Management Systems Lab	II B.Tech., I Semester Common Syllabus for CSE, CSE-CS, CSE-DS, CSE-AI&ML
16	PCS310PC	Skill Development Course-1(NodeJs / React	II B.Tech., I Semester Common Syllabus for CSE, CSE-DS, CSE-AI&ML