CMR TECHNICAL CAMPUS
B.Tech. in INFORMATION TECHNOLOGY
COURSE STRUCTURE
II YEAR SYLLABUS (R22 Regulations)
Applicable from AY 2023-24 Batch

II YEAR I SEMESTER

S. No.	Course Code	Course Title	L	T	P	Credits
1	22IT301PC	Computer Oriented Statistical Methods	3	1	0	4
2	22EC302PC	Digital Electronics	3	0	0	3
3	22IT303PC	Programming with Python	3	0	0	3
4	22IT304PC	Computer Organization and Architecture	3	0	0	3
5	**22IT305PC**	**Object Oriented Programming through Java**	2	0	0	**3**
6	22IT306PC	Python Lab	0	0	3	1.5
7	**22IT307PC**	**Object Oriented Programming through Java**	0	0	3	1.5
8	**22IT308PC**	Skill Development Course (**Node JS/ React JS/ Django**)	0	0	2	1
9	22IT309PC	Gender Sensitization Lab	0	0	2	0
		Total	**14**	**1**	**12**	**20**

II YEAR II SEMESTER

S. No.	Course Code	Course Title	L	T	P	Credits
1	22IT401PC	Discrete Mathematics	3	0	0	3
2	22IT402PC	Database Management Systems	3	0	0	3
3	22IT403PC	Operating Systems	3	0	0	3
4	**22IT404PC**	**Web Technology**	2	0	0	**3**
5	**22IT405PC**	**Software Engineering**	3	0	0	3
6	22IT406PC	Database Management Systems Lab	0	0	2	1
7	**22IT407PC**	**Web Technology Lab**	0	0	2	1
8	**22IT408PC**	Skill Development Course (**Data visualization- R Programming/ Power BI**)	0	0	2	1
9	22IT409PC	Real-time Research Project/ Field-Based Research Project	0	0	4	2
10	22MB410ES	Constitution of India	3	0	0	0
		Total	**17**	**0**	**12**	**20**

COMPUTER ORIENTED STATISTICAL METHODS

B. Tech. III Semester	**L T P C**
Subject Code: 22IT301PC	**3 1 0 4**

Pre-requisites: Mathematics courses of first year of study.

Course Objectives: To learn

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

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

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

UNIT - I: Probability [10 Lectures]

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 Lectures]

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 Lectures]

Uniform Distribution, Normal Distribution, Areas under the Normal Curve, Applications of the NormalDistribution, 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 Lectures]

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 [10 Lectures]

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.

DIGITAL ELECTRONICS

B.Tech. III Semester **L P T C**

Subject Code: 22EC302ES **3 0 0 3**

Pre-requisite: Basic Electrical and Electronics Engineering

Course Objectives:

1. To Compare common forms ofnumber representation in logic circuits.
2. To learn basic techniques for the design of digital circuits and fundamental concepts used in the design of digital systems.
3. To understand the concepts of combinational logic circuits.
4. To design sequential logiccircuits.
5. To know various memories and logic families.

Course Outcomes: Upon completing this course, the student will be able to

1. Compare the numerical information in different forms and Boolean Algebra theorems.
2. Apply the various simplification methods to simplify the given Boolean function.
3. Analyze and design various combinational logic circuits.
4. Learn the concepts of sequential circuits.
5. Illustrate various memories and logic families.

UNIT – I **[10 Lectures]**

NUMBER SYSTEMS: Number Systems, Number base conversions, Binary, Octal, Decimal, and Hexadecimal Numbers, Complements of Numbers, Binary codes: Weighted and Non weighted codes.

BOOLEAN ALGEBRA: Basic Theorems and Properties, Switching Functions- Canonical and Standard Form, Algebraic Simplification, Digital Logic Gates, Universal Gates.

UNIT – II **[08 Lectures]**

GATE – LEVEL MINIMIZATION: The map method, Three-variable map, Four- variable map, product of sums simplification and Sum of Products Simplification using K- map, Don't-care conditions, NAND and NOR implementation.

UNIT – III **[10 Lectures]**

COMBINATIONAL LOGIC: Combinational Circuits, Design procedure, Design of Half Adder, Full Adder, Half Subtractor, Full Subtractor, BCD Adder, Magnitude Comparator, Decoders, Encoders, Multiplexers, and Demultiplexers.

UNIT – IV **[08 Lectures]**

SEQUENTIAL LOGIC: Sequential circuits, latches, Flip-Flops, Design Procedure**,** state Reduction and Assignment, Registers, shift Registers, synchronous counters, Asynchronous Counters, Ring Counter, Johnson Counter.

UNIT – V **[08 Lectures]**

MEMORIES AND LOGIC FAMILIES

Introduction, Random-Access Memory, Memory Decoding, Error Detection and correction Read-only memory, Programmable logic Array, Programmable Array logic. AND, OR and NOT Gates using Diodes, Classification of Integrated circuits, comparison of various logic families.

TEXT BOOKS:

1. M. Morris Mano, Digital Design, 3rd Edition, Pearson Education/PHI, 2021.
2. Albert Paul Malvino Donald P. Leach, Digital Principles and Applications, 2nd Edition, TATA McGraw Hill, 1976.
3. A. Anand Kumar, Pulse and Digital Circuits, 2nd Edition, PHI Publications,2012.

REFERENCE BOOKS:

1. Zvi. Kohavi, Switching and Finite Automata Theory, 3rd Edition,Tata McGraw Hill, 2009.
2. C.V.S. Rao, Switching and Logic Design, 1st Edition, Pearson Education, 2005.
3. Donald D.Givone, Digital Principles and Design, Tata McGraw Hill, Edition, 2017.

WEBLINKS:

1. http://nptel.ac.in/courses/117/106/117106086/ (By Prof. Goutam Saha, Electronics & Electrical Communication Engineering Dept, IIT Kharagpur).
2. www.nptelvideos.in/2012/12/digital-circuits-and-systems.html(By Prof. Santanu
3. Chattopadhyay, Electronics & Electrical Communication Engineering Dept, IIT Kharagpur).
4. https://www.smartzworld.com/notes/switching-theory-and-logic-design-stld/.

PROGRAMMING WITH PYTHON

B. Tech. III Semester **L T P C**
3 0 0 3

Subject Code: 22IT303PC
Prerequisites: Students should install Python on the Linux platform.
Course Objectives:

1. Learn Syntax and Semantics in Python.
2. Explore the arrays and strings in Python.
3. Understand Lists, Dictionaries and Regular expressions in Python.
4. Handle Files, Modules and exceptions in Python.
5. Implement Object Oriented Programming concepts in Python.

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

1. Examine Python syntax and semantics, flow control.
2. Demonstrate proficiency in handling Strings and arrays.
3. Relate Python Programs using core data structures like Lists, Dictionaries anduse Regular Expressions.
4. Correlate experiments on file handling, exception handling, and modules.
5. Develop the concepts of Object-Oriented Programming as used in Python.

UNIT–I **[10 Lectures]**
Introduction to Python: History, Features, Applications, First Python Program, Variables, Data Types, Numbers, Operators, Input and Output statements.
Control Statements: Conditional Statements, A Word on Indentation, Looping Statements, the else Suite, break, continue, pass, assert, return.

UNIT–II **[08 Lectures]**
Arrays in Python: Arrays, Types of Arrays, Working with Arrays using numpy, Creating Arrays, Operations on Arrays, Attributes of an Array, The reshape() Method, The flatten() Method, Matrices in numpy, Matrix Addition and Multiplication.
Strings and Characters: Creating Strings, Operations on Strings, Working with Characters, Sorting Strings, Searching Strings.

UNIT–II **[10 Lectures]**
Functions in Python: Defining a Function, Calling a Function, Parameters, Recursive Functions.
List: Creating Lists using range() Function, Operations on Lists, Methods to Process List, Sorting the List Elements.
Tuple: Creating Tuples, Accessing the Tuple Elements, Operations on Tuple, Functions to Process Tuples.
Dictionaries: Operations on Dictionaries, Dictionary Methods, Sorting the Elements of a Dictionary using Lambdas, Converting Lists into Dictionary, Converting Strings into Dictionary, Passing Dictionaries to Functions.

UNIT–IV [10 Lectures]

Files in Python: File Objects, File Built-in Function [open ()], File Built-in Methods, File Built- in Attributes, Standard Files, Command-line Arguments, File System, File Execution, Persistent Storage Modules, Related Modules.

Exceptions: Exceptions in Python, Detecting and Handling Exceptions, Context Management, *Exceptions as Strings, Raising Exceptions, Assertions, Standard Exceptions, *Creating Exceptions, Why Exceptions (Now)?, Why Exceptions at All?, Exceptions and the sys Module, Related Modules.

Modules: Modules and Files, Namespaces, Importing Modules, Importing Module Attributes, Module Built-in Functions, Packages, Other Features of Modules.

UNIT–V [08 Lectures]

OOPS using Python: Classes and Objects, Inheritance and Polymorphism, Abstract Classes and Interfaces.

Regular Expressions: Introduction, Special Symbols and Characters, Res and Python Multithreaded Programming: Introduction, Threads and Processes, Python, Threads, and the Global Interpreter Lock, Thread Module, Threading Module, Related Modules.

TEXTBOOKS:

1. Wesley J. Chun, Core Python Programming, Second Edition, Pearson, 2006.
2. R. Nageswara Rao Core Python Programming, Dream Tech Press, 2018.

REFERENCE BOOKS:

1. Python Programming using problem solving approach, Reemathareja, Oxford University Press, 2019.
2. Dietel and Dietel, Python How to Program, 2002.

WEB LINKS:

1. https://nptel.ac.in/courses/106106145
2. https://onlinecourses.nptel.ac.in/noc20_cs83/preview
3. https://www.visualpathedu.com/home/course/core-python/1
4. https://www.visualpathedu.com/home/course/advanced-python/3
5. https://www.udemy.com/topic/python/free/
6. https://www.coursera.org/specializations/python

COMPUTER ORGANIZATION AND MICROPROCESSOR

B. Tech. III Semester	**L**	**T**	**P**	**C**
	3	**0**	**0**	**3**

Subject Code: 22IT304PC
Prerequisite:
1. A Course on “Digital Logic Design and Microprocessors”.

Course Objectives:
1. Introduce principles of computer organization and the basic architectural concepts.
2. Explore the basic organization, design, and programming of a simple digital computer.
3. Introduces simple register transfer language to specify various computer operations.
4. Describing memory organization and I/O systems.
5. Topics include pipelining and vector processing.

Course Outcomes: After completion of this course, the students will be able to:
1. Identity of computer organization architecture.
2. Interpret the basics of instruction sets and their functionality.
3. Relate arithmetical operations by using data.
4. Correlate the functional units of the computer.
5. Design a pipeline for consistent execution of instructions.

UNIT–I **[10 Lectures]**

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

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

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.

UNIT–II **[8 Lectures]**

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

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

UNIT–III **[8 Lectures]**

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

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

UNIT–IV **[10 Lectures]**

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

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

UNIT–V **[10 Lectures]**

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, Cache Coherence.

TEXTBOOKS:

1. M. Moris Mano, Computer System Architecture, Third Edition, Pearson/PHI, 2016.
2. William Stallings, Computer Organization and Architecture, 6th ed, Pearson/PHI, 2010.

REFERENCE BOOKS:

1. Car Hamacher, ZvonksVranesic, SafeaZaky, Computer Organization, Vth Edition, McGrawHill, 2002.
2. AndrewS.Tanenbaum, StructuredComputerOrganization,4thEdition,PHI/Pearson, 2003.
3. B. Ram, Computer Fundamentals Architecture and Organization, 5th ed., New Age International Publications, 2000.

WEB LINKS:

1. https://nptel.ac.in/courses/106105163
2. https://onlinecourses.nptel.ac.in/noc20_cs64/preview
3. https://www.udemy.com/topic/computer-architecture/

OBJECT ORIENTED PROGRAMMING THROUGH JAVA

B.Tech. III Semester **L T P C**
3 0 0 3

Subject Code: 22IT305PC
Prerequisites: Basic knowledge on C Programming.

Course Objectives:

1. To introduce the object-oriented programming concepts.
2. To introduce the implementation of packages and interfaces
3. To introduce the concepts of exception handling and multithreading.
4. To introduce the concepts of collection framework.
5. To introduce the design of Graphical User Interface using applets and swing controls.

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

1. Solve real world problems using OOP techniques.
2. Apply the packages and interfaces, streams in I/O.
3. Examine development of exceptions, multithreaded applications with synchronization.
4. Analyze the usage of collection framework.
5. Design GUI based applications using applets and swings.

UNIT–I **[10 Lectures]**

Object-Oriented Thinking-Away of viewing world–Agents and Communities, messages and methods, Responsibilities, Classes and Instances, Class Hierarchies- Inheritance, Method binding, Overriding and Exceptions, Summary of Object-Oriented concepts. Java buzzwords, An Overview of Java, Data types, Variables and Arrays, operators, expressions, control statements, Introducing classes, Methods and Classes, String handling.

Inheritance– Inheritance concept, Inheritance basics, Member access, Constructors, Creating Multilevel hierarchy, super uses, using final with inheritance, Polymorphism-ad hoc polymorphism, pure polymorphism, method overriding, abstract classes, Object class, forms of inheritance specialization, specification, construction, extension, limitation, combination, benefits of inheritance, costs of inheritance.

UNIT–II **[8 Lectures]**

Packages- Defining a Package, CLASSPATH, Access protection, importing packages. Interfaces- defining an interface, implementing interfaces, Nested interfaces, applying interfaces, variables in interfaces and extending interfaces.

Stream based I/O (java.io)–The Stream classes-Byte streams and Character streams, Reading console Input and Writing Console Output, File class, Reading and writing Files, Random access file operations, The Console class, Serialization, Enumerations, auto boxing,generics.

UNIT–III **[6 Lectures]**

Exception handling - Fundamentals of exception handling, Exception types, Termination or resumptive models, Uncaught exceptions, using try and catch, multiple catch clauses, nested try statements, throw, throws and finally, built-in exceptions, creating own exception sub classes.

Multi threading- Differences between thread-based multitasking and process-based multi tasking, Java thread model, creating threads, thread priorities, synchronizing threads,inter thread communication.

UNIT–IV **[6 Lectures]**

The Collections Framework (java.util)- Collections overview, Collection Interfaces, The Collection classes- Array List, Linked List, Hash Set, Tree Set, Priority Queue, Array Deque. Accessing a Collection via an Iterator, Using an Iterator, The For-Each alternative, Map Interfaces and Classes, Comparators, Collection algorithms, Arrays, The Legacy Classes and Interfaces- Dictionary, Hash table

,Properties, Stack, Vector More Utility classes, String Tokenizer, Bit Set, Date, Calendar, Random, Formatter, Scanner

UNIT–V **[10 Lectures]**

GUI Programming with Swing – Introduction, limitations of AWT, MVC architecture, components, containers. Understanding Layout Managers, Flow Layout, Border Layout, Grid Layout, Card Layout, Grid Bag Layout.

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

A Simple Swing Application, Applets – Applets and HTML, Security Issues, Applets and Applications, passing parameters to applets. Creating a Swing Applet, Painting in Swing, A Paint example, Exploring Swing Controls – Jlabel and Image Icon, JTextField.

The Swing Buttons- JButton, JToggle Button, JCheck Box, JRadio Button, JTabbed Pane, JScroll Pane, JList, JCombo Box, Swing Menus, Dialogs.

TEXT BOOKS:

1. Herbert Schildt, Java The complete reference, 9th edition, McGraw Hill Education (India) Pvt. Ltd, 2014.
2. T. Budd, Understanding Object-Oriented Programming with Java, updated edition, Pearson Education, 2000.

REFERENCE BOOKS:

1. J. Nino and F.A. Hosch, An Introduction to programming and OO design using Java, John Wiley & sons, 2007.
2. Y. Daniel Liang, Introduction to Java programming, Pearson Education, 1998.
3. P. Radha Krishna, Object Oriented Programming through Java, University Press, 2007.
4. S. Malhotra, S. Chudhary, Programming in Java, 2nd edition, Oxford University Press, 2018.
5. R. A. Johnson, An Introduction to Java Programming and Object-oriented Application Development, Cengage Learning, 2012.

WEB LINKS:

1. https://nptel.ac.in/courses/106105191
2. https://www.visualpathedu.com/home/course/core-java-training/175
3. https://www.visualpathedu.com/home/course/advanced-java-training/176
4. https://www.coursera.org/specializations/object-oriented-programming
5. https://www.upgrad.com/blog/java-free-online-course/
6. https://www.edx.org/course/java-programming-fundamentals
7. https://education.oracle.com/java/java/pFamily_48

PYTHON LAB

B. Tech III Semester

L	T	P	C
0	0	3	1.5

Subject Code: 22IT306PC

Prerequisites: Students should install any Python IDEs such as IDLE, Anaconda etc.

Co-requisite: A course on “Programming with Python”

Course Objectives:

1. To be able to introduce core programming basics and program design with functions using Python programming language.

2. To understand a range of Object-Oriented Programming, as well as in- depth data and information processing techniques.

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

1. Illustrate Python Programs using core data structures like Lists, Dictionaries and use Regular Expressions.

2. Compare to conduct experiments on file handling, exception handling, and modules.

3. Interpret the concepts of Object-Oriented Programming as used in Python.

4. Use to explore python especially the object-oriented concepts, and the built- in objects of Python.

5. Design to create practical and contemporary applications such as TCP/IP network programming, Web applications, discrete event simulations.

List of Programs:

1. Write a program to demonstrate different number data types in Python.

2. Write a program to perform different Arithmetic Operations on numbers in Python.

3. Write a program to create, concatenate and print a string and accessing sub- string from a given string.

4. Write a python script to print the current date in the following format “Sun May 29 02:26:23 IST2017”

5. Write a program to create, append, and remove lists in python.

6. Write a program to demonstrate working with tuples in python.

7. Write a program to demonstrate working with dictionaries in python.

8. Write a python program to find the largest of three numbers.

9. Write a Python program to convert temperatures to and from Celsius, Fahrenheit. [Formula: c/5 = f-32/9]

10. Write a Python program to construct the following pattern, using a

nested for loop

11. Write a Python script that prints prime numbers less than 20.

12. Write a python program to find factorial of a number using Recursion.

13. Write a program that accepts the lengths of three sides of a triangle as inputs. The program output should indicate whether or not the triangle is a right triangle (Recall from the Pythagorean Theorem that in a right triangle, the square of one side equals the sum of the squares of the other two sides).

14. Write a python program to define a module to find Fibonacci Numbers and import the module to another program.

15. Write a python program to define a module and import a specific function in that module to another program.

16. Write a script named copyfile.py. This script should prompt the user for the names of two text files. The contents of the first file should be input and written to the second file.

17. Write a program that inputs a text file. The program should print all of the unique words in the file in alphabetical order.

18. Write a Python class to convert an integer to a roman numeral.

19. Write a Python class to implement pow (x,n)

20. Write a Python class to reverse a string word by word.

TEXTBOOKS:

1. Wesley J. Chun, Core Python Programming, Second Edition, Pearson, 2006.

2. R. Nageswara Rao Core Python Programming, Dream Tech Press, 2018.

REFERENCE BOOKS:

1. Python Programming using problem solving approach, Reemathareja, Oxford University Press, 2019.

2. Dietel and Dietel, Python How to Program, 2002.

WEB LINKS:

1. https://nptel.ac.in/courses/106106145

2. https://onlinecourses.nptel.ac.in/noc20_cs83/preview

3. https://www.visualpathedu.com/home/course/core-python/1

4. https://www.visualpathedu.com/home/course/advanced-python/3

5. https://www.udemy.com/topic/python/free/

6. https://www.coursera.org/specializations/python

OBJECT ORIENTED PROGRAMMING THROUGH JAVA LAB

B. Tech. III Semester L T P C

0 0 3 1.5

Subject Code: 22IT307PC:

Prerequisites: A course on “Programming for problem solving”

Co-requisites: A course on “OOPS Through Java”.

Course Objectives:

1. To introduce java compiler and eclipse platform and Net Beans ID

2. To write programs using abstract classes and Exception Handling mechanism

3. To write programs for solving real world Problems using java collection frame work

4. To write multithreaded programming.

5. To write GUI programs using swing controls in java

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

1. Define and to develop application programs using oops concepts

2. Extract to write programs using Exceptional Handling approach.

3. Articulate to write programs for solving real world problems using java collection frame work.

4. Illustrate to develop java application to interact with database by using JDBC

5. Prioritize 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 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 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 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.

3. Write a Java program that implements Bubble sort algorithm for sorting in descending order and also shows the number of interchanges occurred for the given set of integers.

4. Write a Java program that implements Quick sort algorithm for sorting a list of names in ascending order

5. 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.

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

7. 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.

8. Write a Java program that implements a multi-thread application that has three threads. First thread generates random integer every 1 second and if the value is even, second thread computes the square of the number and prints. If the value is odd, the third thread will print the value of cube of the number.

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

10. 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

11. 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. 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.

12. Write a Java program that simulates a traffic light. The program lets the users elect 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 selected color. Initially, there is no message shown.

13. 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).

14. 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 are cord are separated by at ab(\t). It takes a name or phone number as input and prints the corresponding other value from the hash table (hint: use hash tables).

15. 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.

TEXT BOOKS:

1. Herbert Schildt, Java The complete reference, 9th edition, McGraw Hill Education (India) Pvt.Ltd, 2014.

2. T. Budd, Understanding Object-Oriented Programming with Java, updated edition, Pearson Education, 2000.

REFERENCE BOOKS:

1. J. Nino and F.A. Hosch, An Introduction to programming and OO design using Java, John Wiley &sons, 2007.

2. Y. DanielLiang, Introduction to Java programming, Pearson Education, 1998.

3. S. Malhotra, S. Chudhary, Programming in Java, 2nd edition, Oxford University Press, 2018.

4. R. A. Johnson, An Introduction to Java Programming and Object-oriented Application Development, Cengage Learning, 2012.

WEB LINKS:

1. https://nptel.ac.in/courses/106106145
2. https://onlinecourses.nptel.ac.in/noc20_cs83/preview
3. https://www.visualpathedu.com/home/course/core-python/1
4. https://www.visualpathedu.com/home/course/advanced-python/3
5. https://www.udemy.com/topic/python/free/
6. https://www.coursera.org/specializations/python

SKILL DEVELOPMENT COURSE (NODE JS/ REACT JS/DJANGO)

B. Tech. III Semester L T P C
0 0 2 1

Subject Code: 22IT308PC

Prerequisites: Object Oriented Programming through Java, HTML Basics

Course Objectives:

1. To implement the static web pages using HTML and do client-side validation using JavaScript.

2. To design and work with databases using Java

3. To develop an end-to-end application using java full stack.

4. To introduce Node JS implementation for server-side programming.

5. To experiment with single page application development using React.

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

1. Illustrate a custom website with HTML, CSS, and Bootstrap and little JavaScript.

2. Extract advanced features of JavaScript and learn about JDBC

3. Use Server – side implementation using Java technologies like

4. Correlate the server – side implementation using Node JS.

5. 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 usingchart.js

14. Create a TODO application in react with necessary components and deploy it into github.

GENDER SENSITIZATION LAB

B.Tech. III Semester **L T P C**
0 0 2 0

Subject Code: 22EN309MC

Course Objectives:

1. To develop students' sensibility with regard to issues of gender in contemporary India.
2. To introduce students to information about some key biological aspects of genders.
3. To expose the students to debates on the politics and economics of work.
4. To help students reflect critically on gender violence.
5. To expose students to more egalitarian interactions between men and women

Course outcomes: After completion of this course the student will be able to :

1. Students will have developed a better understanding of important issues related to gender in contemporary India.
2. Students will attain a finer grasp of how gender discrimination works in our society and how to counter it.
3. Students will acquire insight into the gendered division of labour and its relation to politics and economics.
4. Men and women students and professionals will be better equipped to work and live together as equals.
5. Students will develop a sense of appreciation of women in all walks of life.

UNIT-I: UNDERSTANDING GENDER **[4 Lectures]**

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 **[4 Lectures]**

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 **[4 Lectures]**

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 **[4 Lectures]**

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 **[4 Lectures]**

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

TEXT BOOKS:

Writers: A.Suneetha, Uma Bhrugubanda, Duggirala Vasanta, Rama Melkote, Vasudha Nagaraj, Asma Rasheed, Gogu Shyamala, Deepa Sreenivas and Susie Tharu
Published by: Telugu Academy, Telangana Government Year: 2015

REFERENCE BOOKS:

1. Dr Rajpal Singh, Dr Anupama Sihag, Gender Sensization: Issues and Challenges, Raj Publications 2019.

WEB LINKS:

1. http://ncw.nic.in/sites/default/files/Booklet-%20Gender%20Sensitization_0.pdf

2.http://gmmrcg.in/Content/284_464_7.1.1%20weblink%20annual%20gender%20sensitization%20action%20plan.pdf

DISCRETE MATHEMATICS

B.Tech. IV Semester **L T P C**
3 0 0 3

Subject Code: 22IT401PC

Course Objectives:

1. Introduces elementary discrete mathematics for computer science and engineering.
2. 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: After completion of this course, the students will be able to

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

UNIT - I **[10 Lectures]**

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 **[8 Lectures]**

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

UNIT – III **[8 Lectures]**

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

UNIT – IV **[10 Lectures]**

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 **[10 Lectures]**

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 l. 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, Pearsoneducation, 5th edition.
2. Discrete Mathematical Structures: Thomas Kosy, Tata McGraw Hill publishing co

DATABASE MANAGEMENT SYSTEMS

B.Tech. IV Semester **L T P C**
3 0 0 3

Subject Code: 22IT402PC
Prerequisites: A course on "Data Structures".

Course Objectives:

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

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

1. Define fundamentals of DBMS, database design and normal forms
2. Compare the basics of Relational Model and Relational Algebra
3. Use the basics of SQL for retrieval and management of data.
4. Illustrate the basics of transaction processing and concurrency control.
5. Collaborate familiarity with database storage structures and access techniques.

UNIT – I **[10 Lectures]**

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 **[8 Lectures]**

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 **[10 Lectures]**

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 **[8 Lectures]**

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 **[10 Lectures]**

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. Raghurama Krishnan, Johannes Gehrke, Database Management Systems, Tata Mc GrawHill3rd Edition, 2002.
2. Silberschatz,Korth,Database SystemConcepts,McGrawhill,Vedition, 1980.

REFERENCE BOOKS:

1. Peter Rob & Carlos Corone, Database Systems design, Implementation, and Management, l7th Edition, 1993.
2. ElmasriNavrate, Fundamentals of Database Systems, Pearson Education, 2011.
3. C.J. Date, An Introduction to Database Systems, Pearson Education, 2004.
4. S. Shah and V. Shah, Oracle for Professionals, The XTeam, SPD, 2008.
5. Shah, Database Systems Using Oracle: A Simplified guide to SQL and PL/SQL, PHI, 2004.
6. M. L. Gillenson, Fundamentals of Database Management Systems, Wiley Student Edition, 2011.

WEB LINKS:

1. https://nptel.ac.in/courses/106105175
2. https://onlinecourses.nptel.ac.in/noc21_cs04/preview
3. https://onlinecourses.nptel.ac.in/noc22_cs91/preview
4. https://www.visualpathedu.com/home/course/oracle-database-development/113
5. https://www.mygreatlearning.com/database-management-system/free-courses
6. https://www.coursera.org/learn/database-management
7. https://www.coursera.org/learn/sql-practical-introduction-for-querying-databases

OPERATING SYSTEMS

B.Tech. IV Semester **L T P C**
3 0 0 3

Subject Code: 22IT403PC Prerequisites:

1. A course on “Computer Programming and Data Structures”.
2. A course on “Computer Organization and Architecture”.

Course Objectives:

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

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

1. Illustrate the operating system concepts.
2. The role of computing in CPU scheduling and its management.
3. Use to resolve user problems in the standard environment.
4. Correlate the data storage and retrieval.
5. Design files system inter phase and operations.

UNIT – I **[10 Lectures]**

Operating System - Introduction, Structures - Simple Batch, Multi programmed, 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 **[10 Lectures]**

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 **[08 Lectures]**

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 **[08 Lectures]**

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 **[08 Lectures]**

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.

TEXTBOOKS:

1. Abraham Silberchatz, Peter B. Galvin, Greg Gagne, Operating System Principles, 7th Edition, JohnWiley, 2006.

2. W.R. Stevens, Advanced programming in the UNIX environment, Pearson education, 1992.

REFERENCE BOOKS:

1. Stallings, Operating Systems – Internals and Design Principles, Fifth Edition, Pearson Education/PHI, 2005.
2. Crowley, Operating System A Design-oriented Approach, TMH, 1996.
3. Andrews. Tanenbaum, ModernOperatingSystems,2ndedition, Pearson/PHI, 1992.
4. Kernighan and Pike, UNIX programming environment, PHI/Pearson Education, 1983.
5. U. Vahalia, UNIX Internals -The New Frontiers, Pearson Education, 1995.

WEB LINKS:

1. https://nptel.ac.in/courses/106105214
2. https://onlinecourses.nptel.ac.in/noc21_cs88/preview
3. https://www.coursera.org/specializations/codio-introduction-operating-systems
4. https://www.coursera.org/learn/os-power-user
5. https://www.codecademy.com/learn/fundamentals-of-operating-systems
6. https://www.nesoacademy.org/cs/03-operating-system

WEB TECHNOLOGIES

B. Tech IV Semester **L T P C**
3 0 0 3

Subject Code: 22IT404PC
Prerequisites
1. A course on "Java Programming"

Course Objectives:
1. To introduce PHP language for server-side scripting.
2. To introduce XML and processing of XML Data with Java.
3. To introduce Server-side programming with Java Servlets.
4. Understand the JSP.
5. To introduce Client-side scripting with JavaScript and AJAX.

Course Outcomes: After completion of this course, the students will be able to
1. Define server-side scripting with PHP language.
2. Compare XML and how to parse and use XML Data with Java.
3. Relate to introduce Server-side programming with Java Servlets.
4. Illustrate JSP pages using Cookies and Session tracking.
5. Design client-side scripting, validation of forms and AJAX programming.

UNIT- I **[10 Lectures]**
Introduction to PHP: Declaring variables, data types, arrays, strings, operators, expressions, control structures, functions, reading data from web form controls like text boxes, radio buttons, lists etc., Handling File Uploads. Connecting to database (MySQL as reference), executing simple queries, handling results, Handling sessions and cookies
File Handling in PHP: File operations like opening, closing, reading, writing, appending, deleting etc. on text and binary files, listing directories.
UNIT- II **[08 Lectures]**
HTML Common tags- List, Tables, images, forms, Frames; Cascading Style sheets;
XML: Introduction to XML, Defining XML tags, their attributes and values, Document Type Definition, XML Schemes, Document Object Model, XHTML Parsing XML Data – DOM and SAX Parsers in java.
UNIT – III **[10 Lectures]**
Introduction to Servlets: Common Gateway Interface (CGI), Life cycle of a Servlet, deploying a servlet, The Servlet API, Reading Servlet parameters, Reading Initialization parameters, Handling Http Request & Responses, Using Cookies and Sessions, connecting to a database using JDBC.

UNIT – IV **[08 Lectures]**
Introduction to JSP: The Anatomy of a JSP Page, JSP Processing, Declarations, Directives, Expressions, Code Snippets, implicit objects, Using Beans in JSP Pages, Using Cookies and session for session tracking, connecting to database in JSP.
UNIT – V **[08 Lectures]**
Client-side Scripting: Introduction to JavaScript, JavaScript language – declaring variables, scope of variables, functions. Event handlers (onclick, on submit etc.), Document Object Model, Form validation.
TEXTBOOKS:
1. Uttam K Roy, Web Technologies, Oxford University Press, 2010.
2. Steven Holzner, The Complete Reference PHP, Tata McGraw-Hill, 2007.

REFERENCEBOOKS:

1. Chris Bates, Web Programming, building internet applications, 2edition, Wiley Dream tech, 2002.
2. Hans Bergsten, Java Server Pages, SPDO'Reilly, 2003.
3. D. Flanagan, JavaScript, O'Reilly Media, Inc., 2011.
4. Jon Duckett WROX, Beginning Web Programming with HTML, XHTML, and CSS, 2004.

WEB LINKS:

1. https://www.visualpathedu.com/home/course/ui-development/9
2. https://www.visualpathedu.com/home/course/javascript-training/10
3. https://www.visualpathedu.com/home/course/ui-ux-design-training/199
4. https://www.coursera.org/learn/django-database-web-apps
5. https://extension.harvard.edu/academics/programs/web-technologies-certificate/#outcomes
6. https://www.coursera.org/learn/html-css-javascript-for-web-developers
7. https://www.coursera.org/specializations/web-design
8. https://www.coursera.org/learn/web-development
9. https://www.udemy.com/course/web-developer-course-on-creating-a-business-website

SOFTWARE ENGINEERING

B. Tech IV Semester **L T P C**
3 0 0 3

Subject Code: 22IT405PC
Prerequisites: A course on "Programming for Problem Solving"
Course Objectives

1. Understanding the end-user requirements.
2. Build system models based on the requirements.
3. Design architectural process.
4. Strategic approach for testing methodologies.
5. Understanding software quality management.

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

1. Describe how to translate end-user requirements into the system.
2. Identify and apply the process model based on software requirements.
3. Articulate to build the design of systematic models.
4. Contrast to test strategies and generate a report.
5. Develop quantify the metrics for process and products.

UNIT – I **[10 Lectures]**
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 patterns, process assessment, personal and team process models.
Process models: The waterfall model, incremental process models, evolutionary process models, the unified process.

UNIT – II **[08 Lectures]**
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.
System models: Context models, behavioral models, data models, object models, structured methods.

UNIT – III **[10 Lectures]**
Design Engineering: The 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 **[08 Lectures]**
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.
Product metrics: Software quality, metrics for analysis model, metrics for design model, metrics for source code, metrics for testing, metrics for maintenance.

UNIT – V **[06 Lectures]**
Metrics for Process and Products: Software measurement, metrics for software quality.

Risk management: Reactive Vs proactive risk strategies, software risks, risk identification, risk projection, risk refinement, RMMM, RMMM plan.
Quality Management: Quality concepts, software quality assurance, software reviews, formal technical reviews, statistical software quality assurance, software reliability, the ISO 9000 quality standards.

TEXTBOOKS:

1. Roger S. Pressman, Software Engineering, A practitioner's Approach, 6thedition, McGraw Hill International Edition, 2015.
2. Sommerville, Software Engineering, 7 edition, Pearson Education, 2004.
3. Grady Booch, James Rambaugh, Ivar Jacobson,The unified modeling language user guide, Pearson Education, 1999.

REFERENCE BOOKS:

1. James F. Peters, Witold Pedrycz, Software Engineering: an Engineering approach, John Wiley, 2000.
2. Waman S Jawadekar, Software Engineering principles and practice, The Mc Graw-Hill Companies, 2004.
3. Meilerpage-Jones, Fundamentals of object-oriented designusing UML, Pearson Education, 2000.

WEB LINKS:

1. https://nptel.ac.in/courses/106105182
2. https://www.coursera.org/learn/introduction-to-software-engineering
3. https://www.coursera.org/specializations/software-engineering
4. https://www.edx.org/course/software-engineering-introduction

DATABASE MANAGEMENT SYSTEMS LAB

B. Tech. IV Semester **L T P C**
0 0 2 1

Subject Code: 22IT406PC
Prerequisites: A course on "Data Structures".
Co-requisites: "Database Management Systems"
Course Objectives:
1. Introduce ER data model, database design and normalization
2. Learn SQL basics for data definition and data manipulation

Course Outcomes: After completion of this course, the students will be able to:
1. Define database schema for a given application and apply normalization
2. Compare the skills in using SQL commands for data definition and data manipulation.
3. Relate the solutions for database applications using procedures
4. Correlate solutions for database applications using cursors
5. Develop solutions for database applications using 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.)
7. B. Nested, Correlated subqueries
8. Queries using Aggregate functions, GROUP BY, HAVING and Creation and dropping of Views.
9. Triggers (Creation of insert trigger, delete trigger, update trigger)
10. Procedures
11. Usage of Cursors

TEXT BOOKS:
1. Raghurama Krishnan, Johannes Gehrke, Database Management Systems, Tata Mc GrawHill3rd Edition, 2002.
2. Silberschatz, Korth, Database System Concepts, McGrawhill, Vedition, 1980.

REFERENCE BOOKS:
1. Peter Rob & Carlos Corone, Database Systems design, Implementation, and Management, l7th Edition, 1993.
2. Elmasri Navrate, Fundamentals of Database Systems, Pearson Education, 2011.
3. C.J. Date, An Introduction to Database Systems, Pearson Education, 2004.
4. S. Shah and V. Shah, Oracle for Professionals, The X Team, SPD, 2008.
5. Shah, Database Systems Using Oracle: A Simplified guide to SQL and PL/SQL, PHI, 2004.
6. M. L. Gillenson, Fundamentals of Database Management Systems, Wiley Student Edition, 2011.

WEB LINKS:

1. https://nptel.ac.in/courses/106105175
2. https://onlinecourses.nptel.ac.in/noc21_cs04/preview
3. https://onlinecourses.nptel.ac.in/noc22_cs91/preview
4. https://www.visualpathedu.com/home/course/oracle-database-development/113
5. https://www.mygreatlearning.com/database-management-system/free-courses
6. https://www.coursera.org/learn/database-management
7. https://www.coursera.org/learn/sql-practical-introduction-for-querying-databases

WEB TECHNOLOGIES LAB

B. Tech IV Semester **L T P C**
0 0 2 1

Subject Code: 22IT407PC
Prerequisites: A course on "Java Programming"
Co-requisites: "WEB TECHNOLOGIES"
Course Objectives
1. Understand the working principle of various communication protocols.
2. Analyze the traffic flow and the contents of protocol frames.
3. Explore the concepts of PHP concepts using HTML.
4. Know the XML documents for server-side scripting.

Course Outcomes: After completion of this course, the students will be able to:
1. Define data link layer farming methods
2. Compare routing and congestion issues in network design.
3. Relate PHP concepts in HTML.
4. Correlate server-side scripting using XML.
5. Pivot the JSP and Servlet.

Web Technologies Experiments
1. Write a PHP script to print prime numbers between1-50.
2. PHP script to
    a) Find the length of a string.
    b) Count no of words in a string.
    c) Reverse a string.
    d) Search for a specific string.
3. Write a PHP script to merge two arrays and sort them as numbers, in descending order.
4. Write a PHP script that reads data from one file and write into another file.
5. Develop static pages (using Only HTML) of an online book store.

The pages should resemble:
www.amazon.com. The website should consist the following pages.
a) Home page

a) Registration and user Login

b) User Profile Page

c) Books catalog

d) Shopping Cart

e) Payment By credit card

f) Order Conformation

1. Validate the Registration, user login, user profile and payment by credit card pages using Java Script.

2. Create and save an XML document on the server, which contains 10 users information. Write a program, which takes User Id as an input and returns the user details by taking the user information from the XML document.

3. Install TOMCAT web server. Convert the static web pages of assignment 2 into dynamic web pages using servlets and cookies. Hint: Users information (user id, password, credit card number) would be stored in web.xml. Each user should have a separate Shopping Cart.

4. Redo the previous task using JSP by converting the static web pages of assignment 2 into dynamic web pages. Create a database with user information and books information. The books catalogue
Should be dynamically loaded from the database. Follow the MVC architecture while doing the website.

TEXTBOOKS:

1. Uttam K Roy, Web Technologies, Oxford University Press, 2010.
2. Steven Holzner, The Complete Reference PHP, Tata McGraw-Hill, 2007.

REFERENCEBOOKS:

1. Chris Bates, Web Programming, building internet applications, 2edition, Wiley Dream tech, 2002.
2. Hans Bergsten, Java Server Pages, SPDO' Reilly, 2003.
3. D. Flanagan, JavaScript, O'Reilly Media, Inc., 2011.
4. Jon Duckett WROX, Beginning Web Programming with HTML, XHTML, and CSS, 2004.

WEB LINKS:

1. https://www.visualpathedu.com/home/course/ui-development/9
2. https://www.visualpathedu.com/home/course/javascript-training/10
3. https://www.visualpathedu.com/home/course/ui-ux-design-training/199
4. https://www.coursera.org/learn/django-database-web-apps
5. https://extension.harvard.edu/academics/programs/web-technologies-certificate/#outcomes
6. https://www.coursera.org/learn/html-css-javascript-for-web-developers
7. https://www.coursera.org/specializations/web-design
8. https://www.coursera.org/learn/web-development
9. https://www.udemy.com/course/web-developer-course-on-creating-a-business-website

SKILL DEVELOPMENT COURSE (Data Visualization-R Programming / Power BI)

B. Tech. IV Semester **L T P C**
0 0 2 1

Subject Code: 22IT408PC

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

Course Outcomes: After completion of this course, the students will be able to:
1. Understand How to import data into Tableau.
2. Understand Tableau concepts of Dimensions and Measures.
3. Develop Programs and understand how to map Visual Layouts and Graphical Properties.
4. Create a Dashboard that links multiple visualizations.
5. 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 tool tips, Formatting your data with colors.
8. Creating Dashboards &amp; Storytelling, creating your first dashboard and Story, Design for different displays, adding interactivity to your Dashboard, Distributing &amp; 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. Brett Powell, Microsoft Power BI cookbook, 2nd edition.
2. Roger D. Peng, R Programming for Data Science.
3. The Art of R Programming by Norman Matloff Cengage Learning India.

CONSTITUTION OF INDIA

B.Tech. IV Semester **L T P C**
3 0 0 0

Subject Code:22EN410MC
Prerequisites: None

Course Objectives:
1. Introduction of constitution and it's evolution.

2. To know the fundamentals of Constitution

3. To analyse the fundamental rights and duties of Indian constitution

4. To know the governance of government and administration

5. To appraise the role organs of government.

Course Outcomes: After Completion of this course, the students will be able to:
1. Outline the evolution of Constitution.

2. Relate constitutional fundamentals with the present Era.

3. Analyse Liberalism Federalism and Socialism.

4. Infer the knowledge of Administration and Governance.

5. Appraise and address the role of governments.

UNIT-1 **[5 Lectures]**
History of Making of the Indian Constitution-History of Drafting Committee.

UNIT- 2 **[5 Lectures]**
Philosophy of the Indian Constitution-Preamble Salient Features

UNIT- 3 **[3 Lectures]**
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 **[6 Lectures]**
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 **[6 Lectures]**

Local Administration: District's Administration head: Role and Importance, Municipalities: Introduction, Mayor and role of Elected Representative, CEO of Municipal Corporation. Panchayat raj: Introduction, PRI: Zila Panchayat. Elected officials and their roles, CEO Zila Panchayat: Position and role. Block level: Organizational Hierarchy (Different departments), Village level: Role of Elected and Appointed officials, Importance of grassroot democracy, 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. Dr. M.V. Pylee India's Constitution, 16th Edition, S. Chand 2016.

2. Dr. S. N.Busi, Dr.B. R. Ambedkar framing of Indian Constitution, 1st Edition, 2015.

3. M. P.Jain, Indian Constitutional Law, 7thEdn., Lexis Nexis, 2018.

4. D.D. Basu, Introduction to the Constitution of India, 23rd Edn Lexis Nexis,2018.

Web Links: https://youtu.be/4tI4QXhzqNU https://youtu.be/TXhSrKJ1ahk https://www.youtube.com/live/0FAdfh6NVtA?feature=share https://youtu.be/vq2Q1_v6TNU

.

CMR TECHNICAL CAMPUS

B.Tech. in INFORMATION TECHNOLOGY
COURSE STRUCTURE & SYLLABUS (R22 Regulations)
Applicable from AY 2022-23 Batch

III YEAR I SEMESTER

S. No.	Course Code	Course Title	L	T	P	Credits
1	22IT501PC	Algorithms Design and Analysis	3	0	0	3
2	22IT502PC	Machine Learning	3	0	0	3
3	22IT503PC	Data Communications and Computer Networks	3	1	0	4
4		Professional Elective - I	3	0	0	3
5		Professional Elective - II	3	0	0	3
6	22IT504PC	Software Engineering & Computer Networks Lab	0	0	2	1
7	22IT505PC	Machine Learning Lab	0	0	2	1
8	22EN506HS	Advanced English Communication Skills Lab	0	0	2	1
9	22IT507PC	UI design- Flutter	0	0	2	1
10	22MC508IP	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	22IT601PC	Automata Theory and Compiler Design	3	0	0	3
2	22IT602PC	Introduction to Data Science	3	0	0	3
3	22MB603HS	Business Economics & Financial Analysis	3	0	0	3
4		Professional Elective –III	3	0	0	3
5		Open Elective-I	3	0	0	3
6	22IT604PC	Compiler Design Lab	0	0	2	1
7	22IT605PC	Data Science Lab	0	0	2	1
8		Professional Elective-III Lab	0	0	2	1
9	22IT606PC	Industrial Oriented Mini Project/ Internship/ Skill Development Course (Big data-Spark)	0	0	4	2
10	22MB607MC	Environmental Science	3	0	0	0
		Total	**18**	**0**	**10**	**20**

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

IVYEAR I SEMESTER

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

IV YEAR II SEMESTER

S. No.	Course Code	Course Title	L	T	P	Credits
1	22MB801HS	Organizational Behavior	3	0	0	3
2		Professional Elective – VI	3	0	0	3
3		Open Elective – III	3	0	0	3
4	22IT802PC	Project Stage – II including Seminar	0	0	22	11
		Total	**9**	**0**	**22**	**20**

***MC – Satisfactory/Unsatisfactory**

#Skill Course - 1 credit with 2 Practical Hours

Professional Elective – I

22IT511PE	Biometrics
22IT512PE	Advanced Computer Architecture
22IT513PE	Data Analytics
22IT514PE	Image Processing
22IT515PE	Principles of Programming Languages

Professional Elective - II

22IT521PE	Computer Graphics
22IT522PE	Quantum Computing
22IT523PE	Advanced Operating Systems
22IT524PE	Distributed Databases
22IT525PE	Pattern Recognition

Professional Elective - Ill

22IT631PE	Full Stack Development
22IT632PE	Data Mining
22IT633PE	Scripting Languages
22IT634PE	Mobile Application Development
22IT635PE	Software Testing Methodologies

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

Professional Elective – Ill Lab

22IT631PL	Full Stack Development Lab
22IT632PL	Data Mining Lab
22IT633PL	Scripting Languages Lab
22IT634PL	Mobile Application Development Lab
22IT635PL	Software Testing Methodologies Lab

Professional Elective - IV

22IT741PE	Human Computer Interaction
22IT742PE	Embedded Systems
22IT743PE	Artificial Intelligence
22IT744PE	Information Retrieval Systems
22IT745PE	Ad-hoc & Sensor Networks

Professional Elective - V

22IT751PE	Intrusion Detection Systems
22IT752PE	Real Time Systems
22IT753PE	Blockchain Technology
22IT754PE	Deep Learning
22IT755PE	Software Process & Project Management

Professional Elective - VI

22IT861PE	Natural Language Processing
22IT862PE	Distributed Systems
22IT863PE	Augmented Reality & Virtual Reality
22IT864PE	Web Security
22IT865PE	Cyber Forensics

Open Elective -1:

1. 22IT611OE: Fundamentals of IOT
2. 22IT612OE: Object Oriented Programming using C++

Open Elective -2:

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

Open Elective -3:

1. 22IT831OE: DevOps
2. 22IT832OE: Big Data Technologies

ALGORITHMS DESIGN AND ANALYSIS

B.Tech. III Year II Sem.

L	T	P	C
3	0	0	3

Course Code: 22IT501PC

Prerequisites: Programming for problem solving and Data Structures

Course Objectives:

1. Introduces the notations for analysis of the performance of algorithms.
2. Describes major algorithmic techniques (divide-and-conquer, backtracking, dynamic programming, greedy, branch and bound methods) and mentions problems for which eachtechnique is appropriate.
3. Describes how to evaluate and compare different algorithms using worst, average, and best- caseanalysis.
4. Explains the difference between tractable and intractable problems,
5. To introduce the problems that are P, NP and NP complete.

Course Outcomes:

1. Analyze the performance of algorithms
2. Choose appropriate data structures and algorithm design methods for a specified application
3. Understand the choice of data structures and the algorithm design methods
4. Apply the backtracking technique and branch and bound.
5. Construct the algorithm using the non-deterministic algorithm.

UNIT – I **[10 Lectures]**

Introduction: Algorithm, Performance Analysis-Space complexity, Time complexity, AsymptoticNotations- 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 **[9 Lectures]**

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 **[8 Lectures]**

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 **[10 Lectures]**

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 **[10 Lectures]**

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 BOOKS:

1. Ellis Horowitz, Sartaj Sahni and S. Raja sekharan, Fundamentals of Computer Algorithms, 2 nd Edition, Universities Press, 2012.
2. P. H. Dave, H. B. Dave, Design and Analysis of Algorithms, 2nd edition, Pearson Education, 2013.

REFERENCE BOOKS:

1. M.T. Goodrich and R. Tomassia, Algorithm Design: Foundations, Analysis and Internet examples, John Wiley and sons, 2001.
2. S. Sridhar, Design and Analysis of Algorithms, Oxford Univ. Press, 2014.
3. Aho, Ullman and Hopcroft, Design and Analysis of algorithms, Pearson Education, 1st Edition, 2002.
4. R. Neapolitan and K. Naimipour, Foundations of Algorithms, 4th edition, Jones And Bartlett India Private Limited, 2010.
5. T. H. Cormen, C. E.Leiserson, R. L. Rivest, and Stein, Introduction to Algorithms, 3rd Edition, PHI, 2010

WEB LINKS:

1. https://onlinecourses.nptel.ac.in/noc19_cs47/preview
2. https://www.coursera.org/learn/analysis-of-algorithms
3. https://www.udemy.com/course/design-and-analysis-of-algorithms/
4. https://www.classcentral.com/course/youtube-design-and-analysis-of-algorithms-daa-46806

MACHINE LEARNING

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

Course Code: 22IT502PC

Prerequisites

1. Knowledge of Data Mining, AI.
2. Knowledge of statistical methods.

Course Objectives

1. To introduce the basic concepts of Machine Learning techniques, hypothesis, and version spaces.
2. To learn machine learning problems for different applications.
3. To Understand the concept of Decision Tree and Ensemble Learning.
4. To gain knowledge on evolutionary computing algorithms for real-world problems.
5. To explore Reinforcement and Bayesian Learning techniques.

Course Outcomes:

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

1. Distinguish between supervised, unsupervised, and semi-supervised learning techniques.
2. Predict the accuracy using Machine Learning Algorithms.
3. Build classifiers and design ensemble methods to increase classification accuracy.
4. Implement evolutionary computing algorithms for real-world problems.
5. Analze Reinforcement Learning and Bayesian Networks.

UNIT – I **[10 Lectures]**

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 **[9 Lectures]**

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 **[10 Lectures]**

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 **[9 Lectures]**

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 **[10 Lectures]**

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.

2. Tom M. Mitchell, Machine Learning, McGraw Hill Education, First Edition, 2017.

REFERENCE BOOKS:

1. Peter Flach, ―Machine Learning: The Art and Science of Algorithms that Make Sense of Data‖, First Edition, Cambridge University Press, 2012.

2. Jason Bell, ―Machine learning – Hands on for Developers and Technical Professionals‖, First Edition, Wiley, 2014.

3. Ethem Alpaydin, Introduction to Machine Learning, MIT Press, Fourth Edition, 2020.

4. Mehryar Mohri, Afshin Rostamizadeh, Ameet Talwalkar, Foundations of Machine Learning, Second Edition, MIT Press,2018.

WEB LINKS:

1. https://www.udemy.com/course/machine-learning-foundation-with-practical-approaches/

2. https://mll.iiit.ac.in/

3. https://onlinecourses.nptel.ac.in/noc23_ee87/preview

4. https://www.javatpoint.com/machine-learning

DATA COMMUNICATIONS AND COMPUTER NETWORKS

B.Tech. III Year I Sem.	**L**	**T**	**P**	**C**
	3	**1**	**0**	**4**

Course Code: 22IT503PC

Course Objectives:

Course Objectives:

1. To introduce the fundamental various types of computer networks.
2. To demonstrate the TCP/IP and OSI models with merits and demerits.
3. To explore the various layers of OSI Model.
4. To introduce UDP and TCP Models.
5. Enumerate the layers of the OSI model and TCP/IP. Explain the function(s) of each layer.

Course Outcomes:

1. Student will be able to understand network communication using the layered concept, Open System Interconnect (OSI) and the Internet Model.
2. Student will be able to understand various types of transmission media, network devices; and parameters of evaluation of performance for each media and device.
3. Student will be able to understand the concept of flow control, error control and LAN protocols; to explain the design of, and algorithms used in, the physical, data link layers.
4. Student will understand the working principles of LAN and the concepts behind physical and logical addressing, subnetting and super netting.
5. Student shall understand the functions performed by a Network Management System and to analyze connection establishment and congestion control with respect to TCP Protocol. Student shall understand the principles and operations behind various application layer protocols like HTTP, SMTP, FTP.

UNIT – I **[10 Lectures]**

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

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

UNIT – II **[9 Lectures]**

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

UNIT – III **[9 Lectures]**

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

UNIT – IV **[9 Lectures]**

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

UNIT – V **[10 Lectures]**

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

TEXT BOOKS:

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

REFERENCE BOOKS:

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

WEB LINKS:

1. https://sist.sathyabama.ac.in/sist_coursematerial/uploads/SITA1401.pdf
2. https://peda.net/kenya/ass/subjects2/computer-studies/form-4/itcn
3. https://www.cet.edu.in/noticefiles/265_DCCN%20Lecture%20Notes.pdf
4. https://www.geeksforgeeks.org/difference-between-computer-network-and-data-communication/

BIOMETRICS (Professional Elective – I)

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

Course Code: 22IT511PE

Prerequisites:

1. Data structures

Course Objectives:

1. To understand the concepts of Biometrics, to enable design of biometric system
2. knowledge of biometric foundations
3. understanding of behavioral and physical biometric modalities
4. abilities to build and evaluate a biometric system.
5. awareness of challenges in commercial systems.

Course Outcomes: The student will be able to

1. Understand the technological uplifts with biometrics compared to traditional securing mechanisms.
2. Gain knowledge in building blocks of research fields like Pattern Recognition, Image Processing and Machine Learning etc.
3. Evaluate and Design security systems with biometrics.
4. Have mastered the fundamental concepts and terminology related to biometric recognition of identity,
5. Understand the flow of processing in three major biometrics modalities
6. Be familiar with a selection of current research issues in biometrics,
7. Be aware of the social impact of biometric technology, and
8. Understand the underlying technologies including computer vision, pattern recognition and 2D/3D image processing

UNIT – I **[10 Lectures]**

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

UNIT – II **[9 Lectures]**

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

UNIT -III **[8 Lectures]**

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

UNIT – IV **[11 Lectures]**

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

UNIT – V **[9 Lectures]**

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

TEXT BOOKS:

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

REFERENCE BOOKS:

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

WEB LINKS:

1. https://www.biometricsinstitute.org/what-is-biometrics/types-of-biometrics/
2. https://usa.kaspersky.com/resource-center/definitions/biometrics
3. https://www.miteksystems.com/blog/advantages-and-disadvantages-of-biometrics
4. https://www.tutorialspoint.com/biometrics/biometrics_quick_guide.htm

ADVANCED COMPUTER ARCHITECTURE (Professional Elective – I)

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

Course Code: 22IT512PE

Prerequisites: A Course on "Computer Organization".

Course Objectives:

1. To know the concepts and principles of parallel and advanced computer architectures.
2. To develop the design techniques of Scalable and multithreaded Architectures.
3. To understand the memory and pipeline design.
4. To Apply the concepts and techniques of parallel and advanced computer architectures
5. To design modern computer systems.

Course Outcomes:

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

1. Expertise in Computational models and Computer Architectures.
2. Discover parallel processing applications and Advanced processor technology.
3. Employe the memory organization, pipeline design techniques.
4. Explore the concepts of Scalable Architectures, Superscalar processors, multiprocessors.
5. Impart the vector, multivector and compound vector processing principles.

UNIT – I **[11 Lectures]**

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 **[10 Lectures]**

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 **[9 Lectures]**

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 **[8 Lectures]**

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 **[10 Lectures]**

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

TEXT BOOKS:

1. Kai Hwang, Advanced Computer Architecture, 2nd Edition, McGraw Hill Education, 2010.
2. D. Sima, T. Fountain, P. Kacsuk, Advanced Computer Architectures: A Design Space Approach, Addison-Wesley, 1st edition,1997.

REFERENCE BOOKS:

1. J.L. Hennessy and D.A. Patterson, Computer Architecture: A Quantitative Approach, Morgan Kaufmann, 5th edition, 2011.
2. S.G.Shiva, Advanced Computer Architectures, CRC Press, 1st edition, 2005.
3. G. Hager and G. Wellein, Introduction to High Performance Computing for Scientists and Engineers, CRC Press, 1st edition, 2010.
4. B. Parhami, Computer Architecture: From Microprocessors to Supercomputers, Oxford University Press, Illustrated edition, 2005.

WEB LINKS:

1. https://onlinecourses.nptel.ac.in/noc23_cs07/preview
2. https://cs.hac.ac.il/staff/martin/Adv_Architecture/coursera.html
3. https://www.udemy.com/course/advance-computer-architecture-and-organization/
4. https://www.classcentral.com/course/swayam-advanced-computer-architecture-13884
5. https://link.springer.com/book/10.1007/978-981-10-2209-8
6.https://www.scribd.com/document/138651502/9A05704-Advanced-Computer-Architecture

DATA ANALYTICS (Professional Elective – I)

B.Tech. III Year I Sem.

L	T	P	C
3	0	0	3

Course Code: 22IT513PE

Prerequisites

1. A course on "Database Management Systems".
2. Knowledge of Probability and Statistics.

Course Objectives:

1. To understand the Data pre-processing mechanisms.
2. To learn the fundamental concepts of data analytics.
3. To explore the basic building models for classification.
4. To analyze supervised and unsupervised models.
5. To discover visualization techniques.

Course Outcomes:

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

1. Explore various Data Sources and Pre-processing mechanisms.
2. Carry out data analysis/statistical analysis.
3. Design Data Architecture.
4. Analyze the impact of data analytics for business decisions and strategy.
5. Implement standard data visualization and formal inference procedures.

UNIT – I **[10 Lectures]**

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 **[11 Lectures]**

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 **[10 Lectures]**

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 **[12 Lectures]**

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 **[9 Lectures]**

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. Jiawei Han, Micheline Kamber, Jian Pei Professor, Data Mining: Concepts and Techniques, Morgan Kaufmann, 3rd edition, 2011.
2. Student's Handbook for Associate Analytics – II, III.

REFERENCE BOOKS:

1. Tan, Steinbach and Kumar, Introduction to Data Mining, Pearson, 1st edition, 2005.
2. M. Zaki and W.Meira, Data Mining and Analysis: Fundamental Concepts and Algorithms, Cambridge University Press, 1st edition, 2014.
3. Jure Leskovec, Anand Rajaraman, Jeffrey David Ullman, Mining of Massive Datasets, Cambridge University Press, 3rd edition, 2020.

WEB LINKS:

1. https://nptel.ac.in/courses/106107220
2. https://www.coursera.org/courses?query=data%20analytics
3. https://www.udemy.com/topic/data-analysis/
4. https://careerfoundry.com/en/blog/data-analytics/free-data-analytics-courses/
5. https://www.mygreatlearning.com/data-analytics/free-courses
6. https://www.edx.org/learn/data-analysis

IMAGE PROCESSING (Professional Elective – I)

B.Tech. III Year I Sem.

L	T	P	C
3	0	0	3

Course Code: 22IT514PE

Prerequisites

1. Knowledge of Digital Signal Processing.
2. A course on “Computational Mathematics”.
3. A course on “Computer Oriented Statistical Methods”.

Course Objectives

1. To understand the concepts of Digital Image Processing.
2. To explore image enhancement techniques.
3. To know the image restoration models.
4. To analyze image segmentation models.
5. To compare image techniques.

Course Outcomes:

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

1. Explore the basic concepts of Image processing and two-dimensional signal acquisition, sampling, and quantization.
2. Design image enhancement mechanisms.
3. Apply image restoration models.
4. Implement image segmentation methods.
5. Impart the image compression techniques

UNIT – I [10 Lectures]

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 [9 Lectures]

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

UNIT – III [8 Lectures]

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

UNIT – IV [9 Lectures]

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

UNIT – V [10 Lectures]

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

TEXT BOOKS:

1. R.C. Gonzalez & R. E. Woods, Digital Image Processing, Pearson Education, Fourth edition, 2018.
2. Jain, Fundamentals of Digital Image Processing, Pearson Education India, First Edition, 2015.

REFERENCE BOOKS:

1. Bernd Jahne, Digital Image Processing: Concepts, Algorithms, and Scientific Applications, Springer Verlag, 4th edition, 1997.
2. D. Sundararajan, Digital Image Processing: A Signal Processing and Algorithmic Approach, Springer; Softcover reprint of the original 1st ed. 2017.

WEB LINKS:

1. https://archive.nptel.ac.in/courses/117/105/117105135/
2. https://www.coursera.org/specializations/image-processing
3. https://www.coursera.org/courses?query=image%20processing
4. https://www.udemy.com/topic/image-processing/
5. https://www.classcentral.com/course/computervision-imageprocessing-13567
6. https://www.v7labs.com/blog/image-processing-guide

PRINCIPLES OF PROGRAMMING LANGUAGES (Professional Elective – I)

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

Course Code: 22IT515PE

Prerequisites

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

Course Objectives

1. To introduce important paradigms of programming languages.
2. To understand high-level language design.
3. To describe the programming paradigms using subprograms, blocks, and abstract data types.
4. To understand concurrency control in a programming language.
5. To design functional, logic programming and scripting languages.

Course Outcomes:

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

1. Discover the syntax and semantics of formal languages.
2. Apply a suitable programming paradigm for a given computing application.
3. Implement functional programming.
4. Explore the concepts of concurrency model.
5. Compare and contrast the features of programming languages.

UNIT – I **[12 Lectures]**

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 **[12 Lectures]**

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

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

UNIT – III **[12 Lectures]**

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 **[10 Lectures]**

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 **[12 Lectures]**

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. Robert. W. Sebesta, Concepts of Programming Languages, Pearson, 11th edition, 2015.
2. D. A. Watt, Programming Language Design Concepts, John Wiley & Sons Inc, 1st edition, 2004.

REFERENCE BOOKS:

1. A.B. Tucker, R. E. Noonan, Programming Languages, McGraw-Hill Education, 2nd edition, 2006.
2. K. C. Louden, Programming Languages: Principles and Practice, Course Technology, 2nd edition, 2002.

WEB LINKS:

1. https://nptel.ac.in/courses/106102067
2. https://www.udemy.com/course/fundamentals-of-programming-languages/
3. https://freecomputerbooks.com/Principles-of-Programming-Languages.html
4. https://www.geeksforgeeks.org/principles-of-programming-languages-gq/
5.https://link.springer.com/book/10.1007/978-1-84882-032-6

COMPUTER GRAPHICS (Professional Elective – II)

B.Tech. III Year I Sem. **L T P C**

3 0 0 3

Course Code: 22IT521PE

Prerequisites

1. Knowledge of Mathematical Computation.
2. A course on "Computer Programming and Data Structures"

Course Objectives

1. To know the fundamental concepts of computer graphics.
2. To explore the knowledge of geometrical transformations.
3. To understand object representation using surface.
4. To explore the concept of geometrical projections.
5. To gain knowledge of computer animation.

Course Outcomes:

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

1. Impart mathematics of computer graphics and applications.
2. Design 2D geometrical transformations and 2D viewing functions.
3. Construct 3D object representation using surfaces, polygon rendering and color models.
4. Apply geometric projections for 3D objects and 3D viewing functions.
5. Analyze animation sequence and visible surface detection methods.

UNIT – I **[10 Lectures]**

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 **[10 Lectures]**

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 **[8 Lectures]**

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 **[9 Lectures]**

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 **[9 Lectures]**

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. Donald Hearn and M. Pauline Baker, Computer Graphics C version, Pearson Education India, 2nd edition, 2002.
2. John Hughes, Andries van Dam, Morgan McGuire, David Sklar, James Foley, Steven Feiner, Kurt Akeley, Computer Graphics: Principles & practice, Addison-Wesley Professional, 3rd edition, 2013.
3. Steven Harrington, Computer Graphics: A Programming Approach, McGraw-Hill Inc.,US, 2nd edition, 1983.

REFERENCE BOOKS:

1. David F Rogers, Procedural elements for Computer Graphics, McGraw-Hill Education, 2nd edition, 1997.
2. Neuman and Sproul, Principles of Interactive Computer Graphics, McGraw Hill Education, 2001.
3. Shalini Govil-Pai, Principles of Computer Graphics: Theory and Practice Using OpenGL and Maya, Springer, 5th edition, 2005.

WEB LINKS:

1. https://onlinecourses.nptel.ac.in/noc20_cs90/preview
2. https://www.coursera.org/courses?query=computer%20graphics
3. https://www.udemy.com/course/computer_graphics_subject/
4. https://www.classcentral.com/course/edx-computer-graphics-548
5. https://www.edx.org/learn/computer-graphics
6. https://freecomputerbooks.com/Computer-Graphics.html

7.https://ocw.mit.edu/courses/6-837-computer-graphics-fall-2012/

QUANTUM COMPUTING (Professional Elective – II)

B.Tech. III Year I Sem.

L	T	P	C
3	0	0	3

Course Code: 22IT522PE

Course Objectives

1. Develop a familiarity with the ray optics and gaussian beam.
2. Cultivate the fundamental understanding of electromagnetism.
3. Deliver the knowledge about Fourier and nonlinear optics.
4. Develop elementary problem-solving capability of quantum optics
5. The course aims to serve as an introduction to the quantum computational model with the goal of understanding basic quantum algorithms and analyzing them.
6. The course also addresses limitations of quantum algorithms and introduces the necessary tools and techniques to prove the same.

Course Outcome

1. Have deep understanding of EM wave propagation in guided medium.
2. Demonstrate various fabrication and cabling techniques of optical fiber and its application.
3. Solve the problems attributed spatial frequency filtering and quantum optics.
4. Deliver the knowledge about the various optical nonlinear phenomena.
5. Prove basic facts about quantum information channels
6. Simulate a simple quantum error-correcting code

UNIT – I **[9 Lectures]**

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

UNIT – II **[10 Lectures]**

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 **[8 Lectures]**

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 **[8 Lectures]**

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 **[9 Lectures]**

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

WEB LINKS:

1. https://www.ibm.com/topics/quantum-computing
2. https://www.techtarget.com/whatis/definition/quantum-computing
3. https://www.iberdrola.com/innovation/what-is-quantum-computing

ADVANCED OPERATING SYSTEMS (Professional Elective – II)

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

Course Code: 22IT523PE

Course Objectives

1. Understand the concepts of distributed operating systems.
2. Design approaches of classification algorithms.
3. Identify the issues of dead lock detection.
4. Design of multi-processing operating systems.
5. The concepts of process scheduling and memory management.

Course Outcomes

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

UNIT – I **[9 Lectures]**

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

UNIT – II **[10 Lectures]**

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

UNIT – III **[11 Lectures]**

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

UNIT – IV **[12 Lectures]**

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

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

UNIT – V **[12 Lectures]**

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

TEXT BOOK

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

REFERENCE BOOK

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

WEB LINKS:

1. https://omscs.gatech.edu/cs-6210-advanced-operating-systems
2. https://techworldthink.github.io/MCA/Download/S2/El%20-%20OS/FULL/OS_MD_1.pdf
3. https://mu.ac.in/wp-content/uploads/2022/06/Paper-1-Advance-Operating-System.pdf

DISTRIBUTED DATABASES (Professional Elective – II)

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

Course Code: 22IT524PE

Prerequisites:

1. A course on "Database Management Systems"

Course Objectives:

1. To introduce basic principles of distributed database systems.
2. To Explain the concepts of query processing and optimization in distributed databases.
3. To understand the transaction management procedure in distributed databases.
4. To know the principles and knowledge of parallel databases.
5. To explore the concepts of object database management systems.

Course Outcomes:

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

1. Design the distributed database systems.
2. Interpret query processing and optimization in distributed databases.
3. Summarize the transaction management process.
4. Know about parallel databases and reliability.
5. Develop the distributed object-oriented database systems and related developments

UNIT – I **[10 Lectures]**

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 **[10 Lectures]**

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 **[10 Lectures]**

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 **[10 Lectures]**

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 **[10 Lectures]**

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, Springer Nature Switzerland AG, 4th edition, 2020.
2. Stefano Ceri and Giuseppe Pelagatti, Distributed Databases: Principles and Systems, McGraw Hill Education, 2017.

REFERENCE BOOKS:

1. Hector Garcia-Molina, Jeffrey D. Ullman, Jennifer Widom: "Database Systems: The Complete Book", Pearson, 2nd edition, 2008.
2. Saeed K. Rahimi, Frank S. Haug, Distributed Database Management Systems: A Practical Approach, Wiley-IEEE Computer Society Pr; 1st edition, 2010.

WEB LINKS:

1. https://nptel.ac.in/courses/106106093
2. https://www.coursera.org/lecture/introduction-to-nosql-databases/distributed-databases-Y5y2o
3. https://www.udemy.com/course/from-0-to-1-the-cassandra-distributed-database/
4. https://tejas-kachare18.medium.com/distributed-databases-3755618fd23e
5. https://www.scylladb.com/2022/05/12/distributed-databases-compared/
6. https://devops.com/no-nonsense-guide-to-distributed-databases/

7.https://www.youtube.com/playlist?list=PLK32cwTOY7L8D9GxdYR2JkEIAA1J1sKYO

PATTERN RECOGNITION (Professional Elective – II)

B.Tech. III Year I Sem.

L	T	P	C
3	0	0	3

Course Code: 22IT525PE

Prerequisites:

- Programming for problem solving
- Computer Oriented Statistical Methods

COURSE OBJECTIVES

1. To implement pattern recognition and machine learning theories
2. To develop the mathematical tools required for the pattern recognition.
3. To design and implement certain important pattern recognition techniques
4. To apply the pattern recognition theories to applications of interest
5. To implement the entropy minimization, clustering transformation and feature ordering

COURSE OUTCOMES

1. Design systems and algorithms for pattern recognition (signal classification), with focus on sequences of patterns that are analyzed using, e.g., hidden Markov models (HMM)
2. Analyze classification problems probabilistically and estimate classifier performance
3. Understand and analyze methods for automatic training of classification systems
4. Apply Maximum-likelihood parameter estimation in relatively complex probabilistic models, such as mixture density models and hidden Markov models
5. Understand the principles of Bayesian parameter estimation and apply them in relatively simple probabilistic models

UNIT – I **[10 Lectures]**

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

UNIT – II **[10 Lectures]**

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

UNIT – III **[9 Lectures]**

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

UNIT – IV **[9 Lectures]**

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

UNIT – V **[8 Lectures]**

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

TEXT BOOK:

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

REFERENCE BOOKS:

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

WEB LINKS:

1. https://www.geeksforgeeks.org/pattern-recognition-introduction/
2. https://www.spiceworks.com/tech/artificial-intelligence/articles/what-is-pattern-recognition/
3. https://www.v7labs.com/blog/pattern-recognition-guide
4. https://link.springer.com/journal/11493

SOFTWARE ENGINEERING & COMPUTER NETWORKS LAB

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

Course Code: 22IT504PC

COURSE OBJECTIVES:

1. To have hands on experience in developing a software project by using various software engineering principles and methods in each of the phases of software development.
2. To Understand the functionalities of various layers of OSI model
3. To understand the operating System functionalities
4. Develop methods in each of the phases of software development
5. To know the concept of data transfer between nodes

COURSE OUTCOMES:

1. Ability to translate end-user requirements into system and software requirements
2. Ability to generate a high-level design of the system from the software requirements
3. Will have experience and/or awareness of testing problems and will be able to develop a simple testing report
4. Ability to understand the encryption and decryption concepts in Linux environment
5. Ability to apply appropriate algorithm for the finding of shortest route.
6. Ability to configure the routing table

Software Engineering

List of Experiments:

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

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

Computer Networks

List of Experiments:

1. Implement the data link layer framing methods such as character, character-stuffing and bit stuffing.
2. Write a program to compute CRC code for the polynomials CRC-12, CRC-16 and CRC CCIP
3. Develop a simple data link layer that performs the flow control using the sliding window protocol, and loss recovery using the Go-Back-N mechanism.

4. Implement 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

WEB LINKS:

1. https://www.wgu.edu/blog/difference-between-network-engineering-software-engineering2302.html
2. https://www.techtarget.com/searchnetworking/feature/Differences-between-network-engineers-vs-software-engineers
3. https://www.softwareengineerinsider.com/careers/computer-network-engineer.html

MACHINE LEARNING LAB

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

Course Code: 22IT505PC

Corequisite: A Course on “Machine Learning”.

Course Objective:

1. To overview of the various machine learning techniques using python.
2. To understand the data analysis using ML techniques.
3. To understand the complexity of Machine Learning algorithms and their limitations.
4. To prepare students to become Familiarity with Python programming in AI environment.
5. To train Students with Python programming to comprehend, analyze, design, and create AI platforms and solutions for real-life problems.

Course Outcomes:

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

1. Perform experiments in Machine Learning using real-world data.
2. Apply the modern notions in data analysis-oriented computing.
3. Practice common Machine Learning algorithms and implement their own.
4. Implement python programming in AI environment.
5. 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 BOOKS:

1. Aurelien Geron, Hands-On Machine Learning with Scikit-Learn and TensorFlow, Oreilly, March 2017.
2. Dr. M Gopal, Applied Machine Learning, 1st Edition, McGraw-Hill, 2001.

REFERENCE BOOKS:

1. Stephen Marshland, Machine Learning: An Algorithmic Perspective, Chapman and Hall/CRC, 2nd edition, 2014.
2. Davy Cielen, Arno Meysman, Mohamed Ali, Introducing Data Science: Big Data, Machine Learning, and More, Using Python Tools, Manning Publications, First Edition, 2016.

WEB LINKS:

1. https://deepakdvallur.weebly.com/machine-learning-laboratory.html
2. https://ml.informatik.uni-freiburg.de/
3. https://www.udemy.com/course/machine-learning-foundation-with-practical-approaches/
4. https://mll.iiit.ac.in/

ADVANCED ENGLISH COMMUNICATION SKILLS LAB

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

Course Code: 22EN506HS

Course Objectives:
1. To improve the students' fluency in English, through a well-developed vocabulary.
2. To read and comprehend texts in different contexts.
3. To communicate their ideas relevantly and coherently in writing.
4. To make students' industry ready.
5. To acquire behavioral skills for their personal and professional life.

Course Outcomes:
After completion of this course, the students will be able to:
1. Interpret the vocabulary to improve the fluency in English.
2. Comprehend effectively in different contexts.
3. Develop proficiency in academic reading and writing.
4. Increase possibilities of job prospects.
5. Communicate effectively in formal and informal contexts.

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:

1. To improve the students' fluency in English, with a focus on vocabulary
2. To enable them to listen to English spoken at normal conversational speed by educated English speakers
3. To respond appropriately in different socio-cultural and professional contexts
4. To communicate their ideas relevantly and coherently in writing
5. 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.

• Oxford Advanced Learner’s Compass, 7th Edition

• DELTA’s key to the Next Generation TOEFL Test: Advanced Skill Practice.

• Lingua TOEFL CBT Insider, by Dream tech

• TOEFL & GRE (KAPLAN, AARCO & BARRONS, USA, Cracking GRE by CLIFFS)

TEXT BOOKS:

1. M Asharaf Rizvi, Effective Technical Communication, McGraw Hill Education, Second edition, 2017.
2. Stephen Bailey, Academic Writing: A Handbook for International Students, Routledge, 5th edition, 2017.

REFERENCE BOOKS:

1. V. Hemalatha Nagarajan, Shiv K.Kumar, Learn Correct English – Grammar, Composition and Usage, Pearson, 1st edition, 2005.
2. Aruna Koneru, Professional Communication, McGraw Hill Education (India) Pvt. Ltd, 2017.
3. Meenakshi Raman & Sangeeta Sharma, Technical Communication, 3rd edition, Oxford University Press, 2015.
4. Paul V. Anderson, Technical Communication, 9th edition, Cengage Learning pvt Ltd, New Delhi, 2017.
5. Michael McCarthy, Felicity O'Dell, English Vocabulary in Use Elementary Book with Answers and Enhanced eBook: Vocabulary Reference and Practice, Cambridge University Press, 3rd edition, 2017.
6. David A. McMurrey & Joanne Buckley, Handbook of Technical Communication, Cengage, 1st edition, 2012.

7. Leena Sen, Communication Skills, PHI, 2nd Revised edition, 2007.

8. Colm Downes, Cambridge English for Job-hunting, Cambridge University Press, 1st edition, 2008.
9. Aysha Vishwamohan, English for Technical Communication for Engineering Students, McGraw Hill Education, 1st edition, 2008.

WEB LINKS:

1. https://nptel.ac.in/courses/109104031
2. https://www.udemy.com/topic/english-conversation/?p=3
3. https://www.coursera.org/specializations/business-english
4. (99+) ADVANCED ENGLISH COMMUNICATION SKILLS LAB: Syllabus ADVANCED ENGLISH COMMUNICATION SKILLS LAB | Rock Rishi - Academia.edu

UI DESIGN-FLUTTER

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

Course Code: 22IT507PC

Course Objectives:

1. To learn installation of SDK of Flutter, Xcode and Android Emulator
2. To understanding Stateless and Stateful Widgets and Widget Tree
3. To know Dart basics.
4. To learn starter project template and widget tree.
5. To explore Application of Animation to app.

Course Outcomes:

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

1. Install SDK of Flutter, Xcode and Android Emulator.
2. Create Stateless and Stateful Widgets and Packages.
3. Explore Dart basics.
4. Design starter project template and widget tree.
5. Implement Application of Animation to app.

List of Experiments: Students need to implement the following experiments

1. a) Install Flutter and Dart SDK.
   a) Write a simple Dart program to understand the language basics.

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

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

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

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

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

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

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

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

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

TEXT BOOKS:

1. Marco L. Napoli, Beginning Flutter: A Hands-on Guide to App Development, 1st edition, Wrox publisher, 2019.
2. Alessandro Biessek, Thomas Bailey, Flutter for Beginners: An introductory guide to building cross-platform mobile

applications with Flutter and Dart 2, 2nd edition, Packt Publishing Limited, 2021.

REFERENCE BOOKS:

1. Rap Payne, Beginning App Development with Flutter: Create Cross-Platform Mobile Apps, 1st edition, Apress, 2019.
2. Frank Zammetti, Practical Flutter: Improve your Mobile Development with Google's Latest Open -Source SDK, 1st edition, Apress, 2019.

WEB LINKS:

1. https://onlinecourses.nptel.ac.in/noc21_ar05/preview
2. https://www.coursera.org/projects/googlecloud-getting-started-with-flutter-development-guo1q
3. https://www.udemy.com/course/the-complete-flutter-ui-course-build-amazing-mobile-ui/

4.https://iqonic.design/blog/8-best-flutter-ui-kits-free-ui-kits-and-templates

INTELLECTUAL PROPERTY RIGHTS

B.Tech. III Year I Sem.

L	T	P	C
3	0	0	0

Course Code: 22MC508IP

Course Objectives:

1. To explore intellectual property rights.
2. To outline the basics of trademarks.
3. To describe the copyrights and its laws.
4. To understand the nature of trade secrets and unfair competition.
5. To define the new developments of intellectual property rights.

Course Outcomes:

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

1. Interpret intellectual property rights.
2. Identify different types of trademarks and related laws.
3. Explore copyrights and their laws.
4. Differentiate the trade secrets and unfair competition.
5. Impart the rules, laws, and properties of intellectual property rights.

UNIT – I **[9 Lectures]**

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

UNIT – II **[8 Lectures]**

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 **[10 Lectures]**

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 **[9 Lectures]**

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 **[9 Lectures]**

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 BOOKS:

1. Deborah E. Bouchoux, Intellectual Property: The Law Of Trademarks Copyrights Patents And Trade Secrets, 4Th Edn, Cengage India, 2015.
2. Prabuddha ganguli, Intellectual property right – Unleashing the knowledge economy, McGraw Hill Education, 1st edition, 2017.

REFERENCE BOOKS:

1. William M. Landes, Richard A. Posner, The Economic Structure of Intellectual Property Law, Belknap Press, Illustrated edition, 2003.
2. Rami M. Olwan, Intellectual Property and Development: Theory and Practice, Springer-Verlag Berlin and Heidelberg GmbH & Co. K, 2013th edition, 2015.
3. V K Ahuja, Law Relating To Intellectual Property Rights, Lexis Nexis, Third edition, 2017.

WEB LINKS:

1. https://onlinecourses.nptel.ac.in/noc22_hs59/preview
2. https://www.coursera.org/learn/introduction-intellectual-property
3. https://www.udemy.com/topic/intellectual-property/
4. https://allea.org/intellectual-property-rights/
5.https://www.stopfakes.gov/Online-Intellectual-Property-Training-Module

AUTOMATA THEORY AND COMPILER DESIGN

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

Course Code: 22IT601PC

Course Objectives

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

Course Outcomes

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

UNIT – I **[10 Lectures]**

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

UNIT – II **[10 Lectures]**

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

Pumping Lemma for Regular Languages:

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

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

UNIT – III **[10 Lectures]**

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

Turing Machines:

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

Undecidability:

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

UNIT – IV **[9 Lectures]**

Introduction: The structure of a compiler

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

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

UNIT – V **[9 Lectures]**

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

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

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

TEXT BOOKS:

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

REFERENCE BOOKS:

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

WEB LINKS:

1. https://archive.nptel.ac.in/courses/106/106/106106049/
2. https://www.coursera.org/courses?query=theory%20of%20computation
3. https://www.udemy.com/course/formal-languages-and-automata-theory/
4. https://www.classcentral.com/course/coding-stanford-university-automata-theory-376
5. https://link.springer.com/book/10.1007/978-3-031-43973-5
6. https://www.youtube.com/playlist?list=PLYT7YDstBQmHSRKrNApTqquo2FRlMsoHw

INTRODUCTION TO DATA SCIENCE

B.Tech III Year II Sem.

L T P C
3 1 0 4

Subject Code: 22IT602PC
Prerequisites:

1. A Course on "Mathematical and Statistical Foundations".
2. A Course on "Data Structures."

Course Objectives:
1. To Learn concepts, techniques and tools deal with various facets of data science practice, including data collection and integration
2. To understand the basic types of data and basic statistics
3. To demonstrate the use of vectors, matrices, factors, Data frames and Lists
4. To learn about iterative programming in R
5. To identify the importance of data visualization techniques

Course Outcomes: After the completion of the course the student will able to:
1. Discuss a flow process for data science problems.
2. Identify types of data and basic Statistical Description.
3. Create vectors, matrices and list using R
4. Develop R codes using iterative programming
5. Correlate results to the solution approach

UNIT - I **[10 Lectures]**
Introduction: Definition of Data Science, Big Data and Data Science hype, Datafication, Current Landscape of Perspectives, Statistical Inference, Populations and Samples, Statistical Modeling, Probability Distributions, Fitting a Model, Over Fitting.
Basics of R: Introduction, R-Environment Setup, Programming with R, Basic Data Types.

UNIT - II **[10 Lectures]**
Types of Data: Attribute, Types of Attributes, Describing Attributes by the Number of Values, Asymmetric Attributes, Binary Attributes, Nominal Attributes, Ordinal Attributes, Numeric Attributes, Discrete vs. Continuous Attributes.
Basic Statistical Descriptions of Data: Measuring the Central Tendency, Mean, Median, and Mode, Measuring the Dispersion of Data, Range, Quartiles, Variance, Standard Deviation, and Interquartile Range, Graphic Displays of Basic Statistical Descriptions of Data.

UNIT - III **[9 Lectures]**
Vectors: Creating and Naming Vectors, Vector Arithmetic, Vector Sub Setting. **Matrices:** Creating and Naming Matrices, Matrix Sub Setting, Arrays, Class.
Factors and Data Frames: Introduction to Factors, Factor Levels, Summarizing a Factor, Ordered Factors, Comparing Ordered Factors, Introduction to Data Frame, Sub Setting of Data Frames, Extending Data Frames, Sorting Data Frames.
Lists: Introduction, Creating a List, Creating a Named List, Accessing List Elements, Manipulating List Elements, Merging Lists, Converting Lists to Vectors.4

UNIT- IV **[9 Lectures]**
Conditionals and Control Flow: Relational Operators, Relational Operators and Vectors, Logical Operators, Logical Operators and Vectors, Conditional Statements.
Iterative Programming in R: Introduction, While Loop, For Loop, Looping over List. **Functions in R:** Introduction, Writing a Function in R, Nested Functions, Function Scoping, Recursion, Loading an R Package, Mathematical Functions in R.

UNIT- V **[9 Lectures]**
Plotting and Visualization: A Brief Matplotlib API Primer, Figures and Subplots, Colors, Markers, and Line Styles, Ticks, Labels, and Legends, Annotations and Drawing on a Subplot, Saving Plots to File, Plotting Functions in Pandas, Line Plots, Bar Plots, Histograms and Density Plots, Scatter Plots.
Regression: Linear Regression Analysis, Multiple Linear Regression.

TEXT BOOKS:
1. Doing Data Science, Straight Talk from The Frontline. Cathy O'Neil and Rachel Schutt, O'Reilly, 2014.
2. K G Srinivas, G M Siddesh, "Statistical programming in R", Oxford Publications.

REFERENCE BOOKS:

1. Jiawei Han, Micheline Kamber and Jian Pei. Data Mining: Concepts and Techniques, 3rd ed. The Morgan Kaufmann Series in Data Management Systems.
2. Introduction to Data Mining, Pang-Ning Tan, Vipin Kumar, Michael Steinbanch,Pearson Eduation.
3. Brain S. Everitt, “A Handbook of Statistical Analysis Using R”, Second Edition, 4 LLC, 2014.
4. Dalgaard, Peter, “Introductory statistics with R”, Springer Science & Business Media, 2008.
5. Paul Teetor, “R Cookbook”, O’Reilly, 2011.

WEBLINKS:

1. https://www.geeksforgeeks.org/introduction-to-data-science/
2. https://onlinecourses.nptel.ac.in/noc24_cs54/preview

BUSINESS ECONOMICS & FINANCIAL ANALYSIS

B.Tech III Year II Sem. **LT P C**

3 0 0 3

Course Code: 22MB603HS

Prerequisites: None

Course Objective:

1. Understand the basic Business types, the impact of the Economy Businesses and Firms specifically.
2. Learn the market demand and supply.
3. Gain knowledge of the production theories and cost analysis while dealing with the concept of BEA.
4. Analyze the Business from the Financial Perspective.
5. Interpret the performance of the business with the help of ratios.

Course Outcome:

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

1. Understand the various Forms of Business and the impact of economic variables on the Business.
2. Comprehend the demand and supply analysis.
3. Explore the usage of pricing strategies in PLC.
4. Maintaining the financial accounts of a firm or company.
5. Monitoring the business operations through ratios.

UNIT– I [10 Lectures]

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 in 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 [10 Lectures]

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 & Law of Supply.

UNIT-III [10 Lectures]

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 [10 Lectures]

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.

UNIT-V [8 Lectures]

Financial Analysis through Ratios: Concept of Ratio Analysis, Liquidity Ratios, Turnover Ratios, Profitability Ratios, Proprietary Ratios, Solvency, Leverage Ratios (simple problems). Introduction to Fund Flow and Cash Flow Analysis (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 McGraw Hill, 2011.
3. Geethika Ghosh, Piyali Gosh, Purba Roy Choudhury, Managerial Economics, 2e, Tata McGraw Hill Education Pvt. Ltd., 2012.

REFERENCE BOOKS:

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

WEB LINKS:

1. https://nptel.ac.in/courses/110101005
2. https://www.coursera.org/specializations/managerial-economics-business-analysis
3. https://www.udemy.com/course/engineering-economics-take-decision-between-alternatives/
4. https://www.classcentral.com/course/youtube-mefa-managerial-economics-financial-analysis-91420
5.https://www.emerald.com/insight/publication/issn/1569-3759

FULL STACK DEVELOPMENT (Professional Elective – III)

B.Tech. III Year II Sem.

L	T	P	C
3	0	0	3

Course Code: 22IT631PE

Pre-Requisites:

1. A Course on “Object Oriented Programming”.
2. A Course on “Web Technologies”.

Course Objectives:

1. To learn the core concepts of Full Stack Development.
2. To understand the front-end programming like Data, Files, Http Requests and Responses using node.js.
3. To know the storing and processing huge data and connects with NodeJS application using MangoDB.
4. To provide an in-depth study of Angular Services in Web Applications.
5. To explore interactive user interfaces with react structure.

Course Outcomes:

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

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

UNIT-I **[10 Lectures]**

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 **[9 Lectures]**

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 **[8 Lectures]**

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 **[8 Lectures]**

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 **[8 Lectures]**

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, 2018.
3. Jon Duckett, Web Design with HTML, CSS, JavaScript and Jquery Set, Wiley, Pck edition, 2014.
4. Robin Nixon, Learning PHP, MySQL & JavaScript: A Step-by-Step Guide to Creating Dynamic Websites (Learning PHP, MYSQL, Javascript, CSS & HTML5), O'Reilly Media, 6th edition, 2021.
5. Azat Mardan, Full Stack JavaScript: Learn Backbone.js, Node.js, and MongoDB, APress, 2nd ed. edition, 2018.

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.
4. Eric Bush, Full-Stack JavaScript Development: Develop, Test and Deploy with Mongodb, Express, Angular and Node on Aws, Red Sky, 2016.
5. Tomasz Dyl, Kamil Przeorski, Maciej Czarnecki, Mastering Full Stack React Web Development, Packt Publishing Limited, 2016.

WEB LINKS:

1. https://www.fullstacklabs.co/
2. (2) (PDF) Full Stack Web Development with Hands-On Lab (researchgate.net)
3. https://www.udemy.com/course/the-practical-introduction-to-web-development/
4. https://www.coursera.org/learn/the-full-stac

DATA MINING (Professional Elective – III)

B.Tech. III Year II Sem.

L	T	P	C
3	0	0	3

Course Code: 22IT632PE

Pre-Requisites:

1. A course on "Database management systems"
2. Knowledge of probability and statistics

Course Objectives:

1. To understand the data preprocessing methods.
2. To present methods for mining frequent patterns, associations, and correlations.
3. To describe methods for data classification and prediction.
4. To explore the data clustering approaches.
5. To know the various types of data stores such as spatial, textual, multimedia, streams.

Course Outcomes:

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

1. Explore the types of data to be mined and primitives of the data mining system.
2. Extract interesting patterns from large amounts of data.
3. Discover the classification of data mining in various fields.
4. Employ suitable data mining algorithms to clustering applications.
5. Evaluate the accuracy of supervised and unsupervised models and algorithms.

UNIT – I **[10 Lectures]**

Introduction to Data Mining:

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

UNIT – II **[10 Lectures]**

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

UNIT – III **[8 Lectures]**

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

UNIT – IV **[9 Lectures]**

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

UNIT – V **[10 Lectures]**

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

TEXT BOOKS:

1. Jiawei Han, Micheline Kamber and Jion Pei, Data Mining: Concepts and Techniques, 2nd Edition, Elsevier, 2006.
2. Margaret H Dunham, Data Mining Introductory and Advanced topics, Pearson, 1st edition, 2020.

REFERENCE BOOK:

1. Ian H. Witten and Eibe Frank, Data Mining: Practical Machine Learning Tools and Techniques (Second Edition), Morgan Kaufmann, 2005.
2. Arun K.Pujari, Data Mining Techniques, 3rd edition, Universities Press (India) Pvt. Ltd., 2013.

WEB LINKS:

1. https://onlinecourses.nptel.ac.in/noc21_cs06/preview
2. https://www.coursera.org/specializations/data-mining
3. https://www.udemy.com/course/data-mining/
4. https://www.guru99.com/best-data-mining-tools.html
5. https://www.intotheminds.fr/blog/en/data-mining/
6. https://www.datamation.com/big-data/open-source-data-mining-tools/

SCRIPTING LANGUAGES (Professional Elective – III)

B.Tech. III Year II Sem.

L	T	P	C
3	0	0	3

Course Code: 22IT633PE

Prerequisites:

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

Course Objectives:

1. To understand the web service architecture and applications.
2. To introducing Ruby scripting.
3. To understand the basics of PERL programming.
4. To illustrate the advanced programming in PERL.
5. To Learn TCL.

Course Outcomes:

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

1. Comprehend the SOAP architecture and web services.
2. Extend the Ruby scripting language.
3. Apply the basic Perl programming language.
4. Implement the advanced programming in PERL.
5. Develop TCL programming and select an appropriate language for solving a given problem.

UNIT – I **[10 Lectures]**

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 **[8 Lectures]**

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 **[10 Lectures]**

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 **[9 Lectures]**

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

[10 Lectures]

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. David Barron, The World of Scripting Languages, Wiley Publications, 2009.
2. David Flanagan, Yukihiro Matsumoto, The Ruby Programming Language: Everything You Need to Know, O'Reilly Media, 2008.
3. Dave Thomas, Chad Fowler, Andy Hunt, Programming Ruby: The Pragmatic Programmers' Guide, Pragmatic Bookshelf, 2nd edition, 2004.

REFERENCE BOOKS:

1. J. Lee and B. Ware, Open Source Development with LAMP: Using Linux, Apache, MySQL,Perl, and PHP, Pearson Technology Group, 1st edition, 2002.
2. E. Quigley, Perl by Example, Prentice Hall, 5th edition, 2014.
3. Larry Wall, T. Christiansen and J. Orwant, Programming Perl, O'Reilly Media, 3rd edition, 2000.
4. Ousterhout, Tcl and the Tk Tool kit, Pearson Education, 2nd edition, 2010.

WEB LINKS:

1. https://archive.nptel.ac.in/courses/117/106/117106113/
2. https://www.udemy.com/course/scripting-for-beginner/
3. https://freevideolectures.com/blog/free-courses-learn-scripting-language/
4. https://techenclave.com/threads/programming-and-scripting-links-and-resources.1325/
5. https://www.autoitscript.com/site/autoit/
6. https://www.tutorialspoint.com/ruby/ruby_overview.htm

MOBILE APPLICATION DEVELOPMENT (Professional Elective – III)

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

Course Code: 22IT634PE

Prerequisites

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

Course Objectives:

1. To demonstrate their understanding of the fundamentals of Android operating systems.
2. To improve the skills of using Android software development tools.
3. To learn to broadcast and notifications on a mobile platform.
4. To explore persistent storage.
5. To understand database for mobile applications.

Course Outcomes:

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

1. Work on Android OS Practically.
2. Apply the concepts of mobile applications and user interfaces on Android.
3. Develop the broadcast and notifications on mobile applications.
4. Create persistent storage for Android Applications.
5. Design database for mobile Android Applications.

UNIT – I **[10 Lectures]**

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 **[10 Lectures]**

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 **[10 Lectures]**

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 **[9 Lectures]**

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 **[8 Lectures]**

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 BOOKS:

1. Reto Meier, Professional Android 4 Application Development, Wiley India, 2012.
2. James C Sheusi, Android Application Development for Java Programmers, Cengage Learning, 1st edition, 2013.

REFERENCE BOOKS:

1. Wei-Meng Lee, Beginning Android 4 Application Development, Wiley India, 2012.
2. Neil Smyth, Android Studio 4.2 Development Essentials - Java Edition, Payload Media, Inc., 2021.

WEB LINKS:

1. https://onlinecourses.swayam2.ac.in/nou21_ge41/preview
2. https://www.coursera.org/courses?query=mobile%20app%20development
3. https://www.udemy.com/topic/mobile-development/
4. https://www.mygreatlearning.com/mobile-app-development/free-courses
5. https://techreviewer.co/blog/best-free-open-source-app-development-software-solutions

SOFTWARE TESTING METHODOLOGIES (Professional Elective – III)

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

Course Code: 22IT635PE

Prerequisites

1. Software Engineering

Course Objectives

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

Course Outcomes:

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

UNIT – I **[8 Lectures]**

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 **[12 Lectures]**

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 **[11 Lectures]**

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 **[8 Lectures]**

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

UNIT – V **[9 Lectures]**

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. Baris Beizer, Software Testing Techniques, second edition, Dreamtech, 2002.
2. Dr. K. V. K. K. Prasad, Software Testing Tools, Software Testing Tools: Covering WinRunner, Silk Test, LoadRunner, Jmeter and Test Director with Case Studies, Dreamtech Press, 1st edition, 2004.

REFERENCE BOOKS:

1. Brian Marick, The Craft of Software Testing: Subsystems Testing Including Object-Based and Object-Oriented Testing, Pearson India, 1st edition, 2007.
2. Edward Kit, Software Testing In The Real World: Improving The Process, Pearson India, 1st edition, 2002.
3. William E. Perry, Effective Methods for Software Testing, John Wiley & Sons, 3rd ed. edition, 1995.

4. Glenford J. Myers, Corey Sandler, Tom Badgett, The Art of Software Testing, John Wiley & Sons, 3rd edition, 2011.

WEB LINKS:

1. https://onlinecourses.nptel.ac.in/noc22_cs61/preview
2. https://www.coursera.org/courses?query=software%20testing
3. https://www.udemy.com/courses/development/software-testing/
4. https://www.mygreatlearning.com/academy/learn-for-free/courses/software-testing-fundamentals1
5. https://smartbear.com/learn/automated-testing/software-testing-methodologies/

FUNDAMENTALS OF IOT (Open Elective-I)

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

Course Code: 22IT611OE

COURSE OBJECTIVES

1. Understand the definition and significance of the Internet of Things
2. Discuss the architecture, operation, and business benefits of an IoT solution
3. Examine the potential business opportunities that IoT can uncover
4. Explore the relationship between IoT, cloud computing, and big data
5. Identify how IoT differs from traditional data collection systems

COURSE OUTCOMES:

1. Interpret the impact and challenges posed by IoT networks leading to new architectural models.
2. Illustrate the smart objects and the technologies to connect them to network.
3. Compare different Application protocols for IoT.
4. Infer the role of Data Analytics and Security in IoT
5. Identify sensor technologies for sensing real world entities and understand the role of IoT in various domains of Industry.

UNIT–I **[10 Lectures]**

Introduction to Internet of Things, Characteristics of IoT, Physical design of IoT, Functional blocks of IoT, Sensing, Actuation, Basics of Networking, Communication Protocols, Sensor Networks.

UNIT-II **[10 Lectures]**

Machine-to-Machine Communications, Difference between IoT and M2M, Interoperability in IoT, Introduction to Arduino Programming, Integration of Sensors and Actuators with Arduino.

UNIT – III **[10 Lectures]**

Introduction to Python programming, Introduction to RaspberryPi, Interfacing Raspberry Pi with basic peripherals, Implementation of IoT with RaspberryPi.

UNIT-IV **[10 Lectures]**

Implementation of IoT with RaspberryPi, Introduction to Software defined Network (SDN), SDN for IoT, Data Handling and Analytics.

UNIT-V **[10 Lectures]**

Cloud Computing, Sensor-Cloud, Smart Cities and Smart Homes, Connected Vehicles, Smart Grid, Industrial IoT. Case Study: Agriculture, Healthcare, Activity Monitoring.

TEXTBOOKS:

1. "The Internet of Things: Enabling Technologies, Platforms, and Use Cases", by Pethuru Raj and Anupama C.Raman (CRCPress).
2. “Make sensors”:Terokarvinen, kemo, karvinen and villeyval tokari, 1st edition, maker media,2014.
3. "Internet of Things: A Hands-on Approach",by Arshdeep Bahgaand Vijay Madisetti.

REFERENCEBOOKS:

1. Vijay Madisetti, Arshdeep Bahga,“Internet of Things: A Hands-On Approach”.
2. Waltenegus Dargie, Christian Poellabauer,"Fundamentals of Wireless Sensor Networks: Theory and Practice"
3. Beginning Sensor networks with Arduino and Raspberry Pi–Charles Bell, Apress, 2013.

OBJECT ORIENTED PROGRAMMING USING C++ (Open Elective – I)

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

Course Code: 22IT612OE

Course Objectives:

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

Course Outcomes:

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

UNIT – I **[10 Lectures]**

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

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

UNIT – II **[10 Lectures]**

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

UNIT – III **[9 Lectures]**

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

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

UNIT – IV **[8 Lectures]**

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

UNIT – V **[9 Lectures]**

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

TEXT BOOKS:

1. The Complete Reference C++, 4th Edition, Herbert Schildt, Tata McGraw Hill.
2. Problem solving with C++: The Object of Programming, 4th Edition, Walter Savitch, Pearson Education.

REFERENCE BOOKS:

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

COMPILER DESIGN LAB

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

Course Code: 22IT604PC

Corequisite: A Course on “Compiler Design”.

Prerequisites

1. A Course on “Objected Oriented Programming through Java”
2. A course on “C Programming”

Course Objectives:

1. To understand the various phases in the design of a compiler.
2. To learn the design of top-down and bottom-up parsers.
3. To gain the knowledge of syntax directed translation schemes.
4. To describe lex and yacc tools.
5. To explore the structure of symbol table and bnf rules.

Course Outcomes:

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

1. Design, develop, and implement a compiler for any language.
2. Use lex and yacc tools for developing a scanner and a parser.
3. Design and implement LL and LR parsers.
4. Create and convert BNF rules into YACC form to generate various parsers.
5. Implement Symbol table and its operations.

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.
14. Write a YACC program to evaluate algebraic expression.

TEXTBOOKS:

1. Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffry D. Ullman, Compilers: Principles, Techniques and Tools, Second Edition, Pearson, 2012.
2. Douglas Thain, Introduction to Compilers and Language Design, 2nd edition, Independently published, 2020.
3. Allen I. Holub, Compiler Design in C, Prentice-Hall, 2nd ed. edition, 1990.

REFERENCE BOOKS:

1. Doug Brown, John Levine, Tony Mason, lex & yacc,O'Reilly Media, 2nd edition, 1992.
2. Kenneth C. Louden, Compiler Construction: Principles and Practice, Course Technology Inc, International edition, 1997.
3. Andrew W. Appel, Modern Compiler Implementation in C, Cambridge University Press, Revised ed. edition, 2004.
4. Keith Cooper and Linda Torczon, Engineering a compiler, 2nd Edition, Morgan Kaufmann, 2011.

WEB LINKS:

1. https://www.udemy.com/course/creating-a-c-compiler-from-scratch-module-1/
2. https://github.com/cechengannur/Compiler-Design-Lab-S7
3. https://cse.anangpuria.com/labs/compiler-design-lab/
4. https://www.cse.iitd.ac.in/~sbansal/col728/labs.html
5. https://www.geeksforgeeks.org/lex-program-to-identify-the-identifier/
6. https://github.com/Prati5/Lex-Program-examples/blob/master/identifiersKeywords.l

7.https://simple2code.com/c-programs/c-program-for-predictive-parser/

DATA SCIENCE LAB

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

Course Code: 22IT605PC

Course Objectives:

The main objective of the course is to inculcate the basic understanding of Data Scienceand it's practical implementation using Python.

Course Outcomes:

Upon successful completion of the course, the student will be able to:

1. Perform various operations on numpy arrays
2. Importing data from different file formats using pandas
3. Draw different types of charts using matplotlib

List of Experiments

1. Creating a NumPy Array
   a. Basic ndarray
   b. Array of zeros
   c. Array of ones
   d. Random numbers in ndarray
   e. An array of your choice
   f. Imatrix in NumPy
   g. Evenly spaced ndarray
2. The Shape and Reshaping of NumPy Array
   a. Dimensions of NumPy array
   b. Shape of NumPy array
   c. Size of NumPy array
   d. Reshaping a NumPy array
   e. Flattening a NumPy array
   f. Transpose of a NumPy array
3. Expanding and Squeezing a NumPy Array
   a. Expanding a NumPy array
   b. Squeezing a NumPy array
   c. Sorting in NumPy Arrays
4. Indexing and Slicing of NumPy Array
   a. Slicing 1-D NumPy arrays
   b. Slicing 2-D NumPy arrays
   c. Slicing 3-D NumPy arrays
   d. Negative slicing of NumPy arrays
5. Stacking and Concatenating Numpy Arrays
   a. Stacking ndarrays
   b. Concatenating ndarrays
   c. Broadcasting in Numpy Arrays
6. Perform following operations using pandas
   a. Creating dataframe
   b. concat()
   c. Setting conditions
   d. Adding a new column

7. Perform following operations using pandas
    a. Filling NaN with string
    b. Sorting based on column values
    c. groupby()
8. Read the following file formats using pandas
    a. Text files
    b. CSV files
    c. Excel files
    d. JSON files
9. Read the following file formats
    a. Pickle files
    b. Image files using PIL
    c. Multiple files using Glob
    d. Importing data from database
10. Demonstrate web scraping using python
11. Perform following preprocessing techniques on loan prediction dataset
    a. Feature Scaling
    b. Feature Standardization
    c. Label Encoding
    d. One Hot Encoding
12. Perform following visualizations using matplotlib
    a. Bar Graph
    b. Pie Chart
    c. Box Plot
    d. Histogram
    e. Line Chart and Subplots
    f. Scatter Plot

WEB REFERENCES:

1. https://www.analyticsvidhya.com/blog/2020/04/the-ultimate-numpy-tutorial-for-data-science-beginners/
2. https://www.analyticsvidhya.com/blog/2021/07/data-science-with-pandas-2-minutes-guide-to-key-concepts/
3. https://www.analyticsvidhya.com/blog/2020/04/how-to-read-common-file-formats-python/
4. https://www.analyticsvidhya.com/blog/2016/07/practical-guide-data-preprocessing-python-scikit-learn/
5. https://www.analyticsvidhya.com/blog/2020/02/beginner-guide-matplotlib-data-visualization-exploration python/

FULL STACK DEVELOPMENT LAB (Professional Elective – III)

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

Course Code: 22IT631PL

Pre-Requisites:

1. Object Oriented Programming
2. Web Technologies

Corequisite: A Course on “Full Stack Development”.

Course Objectives:

1. To introduce implementation of Forms, inputs and Services using Angular JS.
2. To learn how to develop a simple web application using Nodejs, Angular JS and Express.
3. To know the implementation of data models using MongoDB.
4. To explore react services and applications.
5. To understand creation of real time web applications.

Course Outcomes:

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

1. Design flexible and responsive Web applications using Node JS, React, Express and Angular.
2. Impart the front end and back-end Tools.
3. Perform CRUD operations with MongoDB on huge amount of data.
4. Develop real time applications using react components.
5. Use various full stack modules to handle http requests and responses.

List of Experiments

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

TEXT BOOKS:

1. Brad Dayley, Brendan Dayley, Caleb Dayley., Node.js, MongoDB and Angular Web Development, 2nd Edition, Addison-Wesley,2019.
2. Mark Tielens Thomas., React in Action, 1st Edition, Manning Publications, 2018.
3. Chris Northwood, The Full Stack Developer: Your Essential Guide to the Everyday Skills Expected of a Modern Full Stack Web Developer, APress, 1st ed. edition, 2018.
4. Juha Hinkula, Hands-On Full Stack Development with Spring Boot 2.0 and React: Build modern and scalable full stack applications using the Java-based Spring Framework 5.0 and React, Ingram short title, 2018.

REFERENCE BOOKS:

1. Vasan Subramanian, Pro MERN Stack, Full Stack Web App Development with Mongo,Express, React, and Node, 2nd Edition, Apress,2019.

2. Brad Green& Seshadri. Angular JS. 1st Edition. O'Reilly Media, 2013.

3. Kirupa Chinnathambi, Learning React: A Hands-On Guide to Building Web Applications Using React and Redux, 2nd edition, Addison-Wesley Professional, 2018.

4. Frank Zammetti, Modern Full-Stack Development: Using TypeScript, React, Node.js, Webpack, and Docker, Apress, 1st ed. edition, 2020.

5. Riaz Ahmed, Full Stack Web Development For Beginners: Learn Ecommerce Web Development Using HTML5, CSS3, Bootstrap, JavaScript, MySQL, and PHP, Independently Published, 2021.

WEB LINKS:

1. https://www.fullstacklabs.co/
2. https://www.udemy.com/course/the-practical-introduction-to-web-development/
3. https://www.coursera.org/learn/the-full-stack
4. https://www.udemy.com/course/ultimate-web/
5. https://www.codecademy.com/learn/paths/full-stack-engineer-career-path?periods=year&plan_id=proGoldAnnualV2&utm_source=pepperjam&utm_medium=affiliate&utm_term=229061&clickId=4626128360&pj_creativeid=8-12462&pj_publisherid=229061

DATA MINING LAB (Professional Elective – III)

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

Course Code: 22IT632PL

Prerequisites

- A course on “Database Management System

Course Objectives:

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

Course Outcomes:

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

LIST OF EXPERIMENTS: Experiments using Weka/ Pentaho/Python

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

TEXT BOOKS:

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

REFERENCE BOOK:

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

SCRIPTING LANGUAGES LAB (Professional Elective – III)

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

Course Code: 22IT633PL

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

Corequisite: A course on "Scripting Languages".

Course Objectives:

1. To understand scripting languages for developing web-based projects.
2. To explore Ruby scripting language.
3. To introduce TCL scripting language.
4. To know Perl Scripting language.
5. To learn web-based applications using scripting languages.

Course Outcomes:

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

1. Differentiate between Scripting languages and programming languages.
2. Gain some fluency programming in Ruby, Perl, TCL.
3. Analyze the differences between Scripting languages and programming languages.
4. Create software systems using scripting languages such as Ruby, Perl, TCL.
5. Develop web-based projects using scripting languages.

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. David Barron, The World of Scripting Languages, Wiley Publications, 2009.
2. David Flanagan and Yukihiro Matsumoto, Ruby Programming language, O'Reilly, 1st edition, 2008.
3. Dave Thomas, Chad Fowler, Andy Hunt, Programming Ruby: The Pragmatic Programmers' Guide, Pragmatic Bookshelf, 2nd edition, 2004.
3. Brent B. Welch, Practical Programming in Tcl and Tk, Prentice Hall, 3rd edition, 1999.
4. David Till, Teach Yourself Perl 5 in 21 Days, Sams, 2nd edition, 1996.

REFERENCE BOOKS:

1. J.Lee and B. Ware, Open Source Web Development with LAMP using Linux Apache, MySQL, Perl and PHP, (Addison Wesley) Pearson Education, 1st edition, 2002.
2. E. Quigley, Perl by Example, Pearson Education, 5th edition, 2014.
3. Larry Wall, T. Christiansen and J. Orwant, Programming Perl, O'Reilly, SPD, 3rd edition, 2000.
4. Boris Paskhaver, Learn to Code with Ruby, Packt, 2022.
5. John K. Ousterhout, Tcl and the Tk Toolkit, Addison Wesley, 1994.
6. J. P. Flynt, Perl Power, Cengage Learning, 1st edition, 2006.
7. Tom Christiansen, brian foy, Larry Wall, Jon Orwant, Programming Perl: Unmatched power for text processing and scripting, O'Reilly Media, 4th edition, 2012.

WEB LINKS:

1. https://www.ni.com/en/shop/labview/scripting-languages-and-ni-labview.html
2. https://extendedstudies.ucsd.edu/courses-and-programs/programming-and-scripting-languages

3.https://guides.lib.wayne.edu/information_science_tools/scr

MOBILE APPLICATION DEVELOPMENT LAB (Professional Elective – III)

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

Course Code: 22IT634PL
Corequisite: A course on “Mobile Application Development”.
Course Objectives:
1. To learn how to develop Applications in an android environment.
2. To introduce Layout Management and Multi layout definition techniques to create adaptable User Interface.
3. To know user interface for mobile Application using widgets with event handling.
4. To describe push notifications for incoming messages.
5. To explore applications to the Android marketplace for distribution.

Course Outcomes:
After completion of this course, the students will be able to:
1. Work with Android operating system practically.
2. Configure Android environment and use development tools.
3. Develop rich user Interfaces by using layouts and controls.
4. Implement User Interface components for android application development.
5. Create Android applications using a database and publish it.

LIST OF EXPERIMENTS:

1. (a) 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 sendan 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. Reto Meier, Professional Android 4 Application Development, Wiley India, (Wrox), 2012.
2. James C Sheusi, Android Application Development for Java Programmers, Cengage, 2013.
3. Pradeep Kothari, Kogent Learning Solutions Inc., Android Application Development (With Kitkat Support), Black Book, Dreamtech Press, 2014.
4. Ashok Kumar R, Baskaran p, Mobile Application Development Laboratory, 2017.

REFERENCE BOOKS:

1. Wei-Meng Lee, Beginning Android 4 Application Development, Wiley India (Wrox), 2013.
2. Michael Burton, Android App Development for Dummies, Wiley, Third edition, 2015.
3. Priyadharshini Ashokkumar, VTU-Mobile Application Development Lab Manual-18CSMP68: MAD Lab Manual(18CSMP68), 2021.

WEB LINKS:

1. https://www.udemy.com/course/mobile-application-development-using-android/
2. https://github.com/shubhaam13/Mobile-Application-Development
3. https://www.studocu.com/row/document/university-of-agriculture-faisalabad/mobile-application/mobile-application-development-lab/22189066
4. https://www.codingconnect.net/mobile-application-development-lab/
5. https://www.tutorialspoint.com/android
6. https://www.tutorialspoint.com/android/android_advanced_tutorial.pdf
7. https://developer.android.com

SOFTWARE TESTING METHODOLOGIES LAB (Professional Elective – III)

B.Tech. III Year II Sem.

L	T	P	C
0	0	2	1

Course Code: 22IT635PL
Prerequisites: A course on "Java programming" or "C Programming".
Corequisite: A course on "Software Testing Methodologies".

Course Objectives

1. To demonstrate use of testing tool-WinRunner in applications.
2. To introduce skills in software test automation and management.
3. To learn the recording of tests in analog and context sensitive modes.
4. To know GUI, Bitmap and Database checkpoints for an application.
5. To understand the test cases for windows application and batch testing.

Course Outcome:

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

1. Work with WinRunner user interface and describe the benefits of automated testing.
2. Design and develop the best test strategies in accordance with the development model.
3. Record the test in analog and context sensitive modes.
4. Create various checkpoints like GUI, Bitmap and Database for an application.
5. Perform the 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
1. Database checkpoint for Default check
2. Database checkpoint for custom check
3. Database checkpoint for runtime record check 9.
   a. Data driven test for dynamic test data submission
   b. Data driven test through flat files
   c. Data driven test through front grids
   d. Data driven test through excel test

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

TEXT BOOKS:

1. K.V.K.K. Prasad, Software Testing Tools: Covering WinRunner, SilkTest, LoadRunner, JMeter, TestDirector and QTP with Case Studies, Dreamtech Press, 1st edition, 2007.

2. Mark Harbin, Thomas Messerschmidt, WinRunner 7 In A Hurry! Software Test Automation With WinRunner, Riverside Pr Intl, 2001.

REFERENCE BOOKS:

1. Nageshwar Rao Pusuluri, SOFTWARE TESTING CONCEPTS AND TOOLS, Dreamtech Press, 2006.
2. Naresh Chauhan, Software Testing: principles and practices, Oxford University Press, Second edition, 2016.

WEB LINKS:

1. https://support.microfocus.com/kb/kmdoc.php?id=KM0247&fileName=hp_man_WinRunner-version5.01-WR_TUT_pdf.pdf
2. https://onlinecourses.nptel.ac.in/noc23_cs38/preview
3. https://www.udemy.com/course/software-manual-testing-course/
4. https://www.scribd.com/document/418157684/stm-lab
5. https://itqcr.com/

INDUSTRIAL ORIENTED MINI PROJECT/INTERNSHIP/SKILL DEVELOPMENT COURSE (BIG DATA SPARK)

B.Tech. III Year II Sem.

L	T	P	C
0	0	4	2

Course Code: 22IT606PC

Course Objectives

1. To understand MapReduce programs, usage of Hadoop and Spark for large datasets.
2. To learn Spark Ecosystem and its components for large dataset analysis.
3. To know usage of filter, count, distinct, map, flatMap RDD Operations in Spark.
4. To explore Spark SQL queries, sqoop to import and export data from Hadoop to database.
5. To demonstrate Spark joins on sample datasets.

Course Outcome:

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

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

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. Marko Bonaci and Petar Zecevic, Spark in Action, Wiley India, First Edition, 2016.
2. Raju Kumar Mishra and Sundar Rajan Raman, PySpark SQL Recipes: With HiveQL, Dataframe and Graphframes, Apress Media, 1st edition, 2019.

REFERENCE BOOKS:

1. Bill Chambers, Matei Zaharia, Spark: The Definitive Guide: Big Data Processing Made Simple, O'Reilly Media, 1st edition, 2018.
2. Tom White, Hadoop: The Definitive Guide: Storage and Analysis at Internet Scale, O'Reilly Media, 4th edition, 2015.

WEB LINKS:

1. https://infyspringboard.onwingspan.com/web/en/login?ref=%2Fapp%2Ftoc%2Flex_auth_013301505844518912251%2Foverview
2. https://www.javatpoint.com/spark-big-data
3. https://www.coursera.org/learn/introduction-to-big-data-with-spark-hadoop
4. https://onlinecourses.nptel.ac.in/noc21_cs86/preview

ENVIRONMENTAL SCIENCE

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

Course Code: 22MB607MC

Course Objectives:

1. To recognize the importance of environment, self-sustain eco systems.
2. To understand various natural resources – their importance, over exploitation, effects, and mitigation measures.
3. To create basic knowledge about different biotic resources and their need for conservation.
4. To be aware of the impacts of developmental activities and mitigation measures.
5. To know various environmental policies, legislation, and regulations.

Course Outcomes:

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

1. Explore harmonious co-existence of nature and human beings.
2. Recognize conservation of natural resources particularly alternate sources of energy.
3. Evaluate and develop technologies for conservation of biodiversity in a sustainable manner.
4. Generate ideas and implement technologies to solve environmental problems associated with air, water, and soil.
5. Develop technologies based on ecological principles and environmental regulations which helps in sustainable developments.

UNIT – I **[10 Lectures]**

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 **[9 Lectures]**

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 **[8 Lectures]**

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 **[9 Lectures]**

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

UNIT – V **[10 Lectures]**

Environmental Policy, Legislation & EIA: Environmental Protection act, Legal aspects Air Act- 1981, Water

Act, Forest Act, Wild life Act, Municipal solid waste management and handling rules, biomedical waste management and handling rules, hazardous waste management and handling rules. EIA: EIA structure, methods of baseline data acquisition. Overview on Impacts of air, water, biological and Socio- economical aspects. Strategies for risk assessment, Concepts of Environmental Management Plan
(EMP). **Towards Sustainable Future:** Concept of Sustainable Development, Population and its explosion, Crazy Consumerism, Environmental Education, Urban Sprawl, Human health, Environmental Ethics, Concept of Green Building, Ecological Foot Print, Life Cycle assessment (LCA), Low carbon life style.

TEXT BOOKS:

1. Erach Bharucha, Textbook of Environmental Studies for Undergraduate Courses, Universities Press (India) Private Limited, 2019.
2. R. Rajagopalan, Environmental Studies, Oxford University Press, Third edition, 2015.

REFERENCE BOOKS:

1. Richard T. Wright, Dorothy F. Boorse, Environmental Science: Toward a Sustainable Future, Benjamin -Cummings Pub Co, 13th edition, 2016.
2. Gilbert M. Masters and Wendell P. Ela, Introduction to Environmental Engineering and Science, Pearson, 3rd edition, 2007.
3. Daniel B. Botkin, Edward A. Keller, Environmental Science: Earth as a Living Planet, Wiley, 8th edition, 2010.
4. Anubha Kaushik, Perspectives In Environmental Studies, New Age International Publishers, 2014.
5. Dr. M.Anji Reddy, Textbook of Environmental Science and Technology, BS Publications, 1St ed. edition, 2014.

WEB LINKS:

1. https://onlinecourses.nptel.ac.in/noc23_hs155/preview
2. https://www.coursera.org/courses?query=environmental%20science
3. https://www.udemy.com/topic/environmental-science/
4. https://www.classcentral.com/subject/environmental-science
5. https://www.sciencedaily.com/news/earth_climate/environmental_science/

INFORMATION SECURITY

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

Course Code: 22IT701PC

Prerequisites

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

Course Objectives

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

Course Outcomes

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

UNIT - I

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

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

UNIT - II

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

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

UNIT - III

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

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

UNIT - IV

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

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

UNIT - V

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

TEXT BOOK:

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

REFERENCE BOOKS:

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

CLOUD COMPUTING

B.Tech. IV Year I Sem.

L	T	P	C
3	0	0	3

Course Code: 22IT702PC

Pre-requisites:

1. A course on “Computer Networks”
2. A course on “Operating Systems”

Course Objectives:

1. To introduce an insight into cloud computing.
2. To learn cloud service types, and practice on cloud deployment models and technologies supporting and driving the cloud.
3. To gain knowledge on programming models for cloud and development of software applications that runs the cloud and various services available from major cloud providers.
4. To explore networking for cloud computing.
5. To describe security concerns and issues in cloud computing.

Course Outcomes:

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

1. Practice on different computing paradigms and potential of the paradigms and specifically cloud computing.
2. Identify cloud service types, and practice on cloud deployment models and technologies supporting and driving the cloud.
3. 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.
4. Design networking for cloud computing.
5. Analyze security concerns and issues in cloud computing.

UNIT – I **[10 Lectures]**

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

UNIT – II **[12 Lectures]**

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 **[8 Lectures]**

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

UNIT – IV **[12 Lectures]**

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 **[9 Lectures]**

Security in Cloud Computing, and Advanced Concepts in Cloud Computing

TEXT BOOKS:

1. K. Chandrasekhran, Essentials of cloud Computing, CRC press, 2014.
2. Rajkumar Buyya, James Broberg andAndrzej M. Goscinski, Cloud Computing: Principles and Paradigms, Wiley, 2011.

REFERENCE BOOKS:

1. Kai Hwang, Geoffery C. Fox, Jack J. Dongarra, Distributed and Cloud Computing, Elsevier, 2012.
2. Tim Mather, Subra Kumaraswamy, Shahed Latif, Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance, O'Reilly, SPD, rp2011.

WEB LINKS:

1. https://nptel.ac.in/courses/106105167
2. https://www.coursera.org/browse/information-technology/cloud-computing
3. https://www.udemy.com/topic/cloud-computing/
4. https://www.edx.org/learn/cloud-computing
5. https://www.classcentral.com/course/google-cloud-computing-foundations-cloud-computing-fundamentals-98568

HUMAN COMPUTER INTERACTION (Professional Elective - IV)

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

Course Code: 22IT741PE

Course Objectives:

1. To gain an overview of Human-Computer Interaction (HCI).
2. To understand the alternatives to traditional "keyboard and mouse" computing.
3. To Getting familiarity with the vocabulary associated with sensory and cognitive systems.
4. To explore the basics of software tools.
5. To know the importance of cognitive levels of the user interface.

Course Outcomes:

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

1. Apply HCI and principles to interaction design.
2. Design process of human-computer interaction.
3. Implement the principles of GUI.
4. Develop certain tools for blind or PH people.
5. Create applications of virtual and augmented reality interfaces.

UNIT – I **[10 Lectures]**

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 **[9 Lectures]**

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 **[8 Lectures]**

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 **[11 Lectures]**

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 **[10 Lectures]**

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. Wilbert O Galitz, The Essential Guide to User Interface Design: An Introduction to GUI Design Principles and Techniques, John Wiley & Sons, 2nd edition, 2002.
2. Alan Dix, Janet Fincay, GreGoryd, Abowd, Russell Bealg, Human – Computer Interaction, Pearson Education, 3rd edition, 2003.

REFERENCE BOOKS:

1. Ben Shneidermann, Designing the user interface, 3rd Edition, Pearson Education Asia, 1997.
2. Helen Sharp, Jennifer Preece, Yvonne Rogers, Interaction Design Prece, Wiley, 5th edition, 2019.
3. Soren Lauesen, User Interface Design: A Software Engineering Perspective, Addison-Wesley, 1st edition, 2004.
4. D. R. Olsen, Human –Computer Interaction, Cengage Learning, 1st edition, 2009.
5. Smith - Atakan, Human--Computer Interaction: Basics and Practice, Cengage Learning, 1st edition, 2010.

WEB LINKS:

1. https://nptel.ac.in/courses/106103115
2. https://www.coursera.org/courses?query=human%20computer%20interaction
3. https://www.udemy.com/course/human-computer-interactions/
4. https://www.classcentral.com/course/hciucsd-306
5.https://www.oswego.edu/human-computer-interaction/useful-links

EMBEDDED SYSTEMS (Professional Elective – IV)

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

3 0 0 3

Course Code: 22IT742PE

Pre-requisites:

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

Course Objectives:

1. To provide an overview of principles of Embedded System.
2. To presents fundamental embedded system design architecture, possibility, and challenges, both with respect to software and hardware.
3. To know the formal method, testing, verification, validation and simulation techniques and tools to engineer reliable and safe embedded systems.
4. To understand software and program for an embedded system design.
5. To analyze operating system concept types and RTOS.

Course Outcomes:

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

1. Impart the selection procedure of processers used in embedded systems.
2. Desing the procedure of embedded firmware.
3. Implements code for interfacing various devices.
4. Develop real time applications using hardware and software resources.
5. Evaluate the significance of RTOS, Task communication techniques and devices.

UNIT – I **[10 Lectures]**

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 **[9 Lectures]**

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 **[8 Lectures]**

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 **[8Lectures]**

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 **[9 Lectures]**

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 BOOKS:

1. Shibu K V, Introduction to Embedded Systems, McGraw Hill Education India Private Limited, Second edition, 2017.
2. David E. Simon, An Embedded Software Primer, Addison Wesley, 1999.
3. Tammy Noergaard, Embedded Systems Architecture: A Comprehensive Guide for Engineers and Programmers, Newnes (an imprint of Butterworth-Heinemann Ltd), 2nd edition, 2013.
4. Wayne Wolf, Computers as Components: Principles of Embedded Computing System Design (The Morgan Kaufmann Series in Computer Architecture and Design), Morgan Kaufmann, 2nd edition, 2008.

REFERENCE BOOKS:

1. Rajkamal, Embedded Systems Architecture, Programming and Design, McGraw Hill Education, Third edition, 2017.
2. Frank Vahid, Tony D. Givargis, Embedded System Design: A Unified Hardware / Software Introduction, Wiley, New edition, 2001.
3. Jonathan Valvano, Embedded Microcomputer Systems: Real Time Interfacing, CENGAGE Learning Custom Publishing, 3rd

edition, 2011.
4. Heath, Embedded Systems Design, Elsevier India, Second edition, 2005.

WEB LINKS:

1. https://nptel.ac.in/courses/108102045
2. https://www.coursera.org/learn/introduction-embedded-systems
3. https://www.coursera.org/courses?query=embedded%20systems
4. https://www.udemy.com/topic/embedded-systems/
5. https://www.classcentral.com/subject/embedded-systems
6. https://www.edx.org/learn/embedded-systems
7. https://www.keil.com

ARTIFICIAL INTELLIGENCE (Professional Elective –IV)

B.Tech. IV Year I Sem.

L	T	P	C
3	**0**	**0**	**3**

Course Code: 22IT743PE

Prerequisites:

1. Programming for problem solving, Data Structures.

Course Objectives:

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

Course Outcomes:

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

UNIT – I **[9 Lectures]**

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 **[9 Lectures]**

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 **[9 Lectures]**

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 **[9 Lectures]**

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 **[9 Lectures]**

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

INFORMATION RETRIEVAL SYSTEMS (Professional Elective – IV)

B.Tech. IV Year I Sem.

L	T	P	C
3	**0**	**0**	**3**

Course Code: 22IT744PE

Prerequisites:

1. A course on “Data Structures”.
2. A course on “Data Base Management Systems”.

Course Objectives

1. To learn the concepts of IRS.
2. To understand the structure of information.
3. To design document clustering algorithms.
4. To explore Information Retrieval System for web search tasks.
5. To learn visualization technologies for multimedia data retrieval.

Course Outcomes:

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

1. Apply IR principles to locate relevant information large collections of data.
2. Develop data models using statistical approaches.
3. Implement different automatic document clustering algorithms.
4. Design the Information Retrieval System for web search tasks.
5. Apply visualization tools for multimedia information retrieval.

UNIT – I **[10 Lectures]**

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 **[9 Lectures]**

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 **[8 Lectures]**

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 **[11 Lectures]**

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 **[11 Lectures]**

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 BOOKS:

1. Gerald J. Kowalski, Mark T. May bury, Information Storage and Retrieval Systems – Theory and Implementation, Springer, 2nd edition, 2000.
2. Frakes, W.B., Ricardo Baeza-Yates, Information Retrieval Data Structures and Algorithms, Prentice Hall, 1st edition, 1992.

REFERENCE BOOKS:

1. Robert R. Korfhage, Information Storage & Retrieval, John Wiley & Sons, 1997.
2. Ricardo Baeza-Yates, Berthier Ribeiro-Neto, Modern Information Retrieval: The Concepts and Technology Behind Search, Addison-Wesley Professional, 2nd edition, 2011.

WEB LINKS:

1. Introduction to Information Retrieval Systems || || Information Retrieval Systems || IRS (youtube.com)
2. https://nlp.stanford.edu/IR-book/information-retrieval-book.html
3. https://www.coursera.org/courses?query=information%20retrieval
4. https://www.udemy.com/topic/information-systems/
5. https://link.springer.com/chapter/10.1007/978-0-387-21721-5_15

AD HOC & SENSOR NETWORKS (Professional Elective – IV)

B.Tech. IV Year I Sem.

L	T	P	C
3	0	0	3

Course Code: 22IT745PE

Prerequisites

1. A course on "Computer Networks"
2. A Course on "Distributed Systems"
2. A course on "Mobile Computing"

Course Objectives:

1. To understand the challenges of routing in ad-hoc and sensor networks.
2. To know the MAC and transport protocols for ad-hoc networks.
3. To introduce the protocols TCP and MANET.
4. To describe basics of Wireless sensors, and Lower Layer Issues.
5. To learn basics of Wireless sensors, and Upper Layer Issues of WSN.

Course Outcomes:

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

1. Gain knowledge of Ad Hoc and Wireless Sensor Networks.
2. Compare the MAC and routing protocols for ad hoc networks.
3. Design the solutions for TCP over Ad-hoc sensor networks.
4. Solve the issues in real-time application development based on ASN.
5. Develop the applications in the domain of ASN

UNIT – I **[12 Lectures]**

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 **[10 Lectures]**

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 **[9 Lectures]**

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 **[9 Lectures]**

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 **[9 Lectures]**

Upper Layer Issues of WSN

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

TEXT BOOKS:

1. Carlos Corderio Dharma P. Aggarwal, Ad Hoc and Sensor Networks–Theory and Applications, World Scientific Publications, March 2006.
2. Feng Zhao, Leonidas Guibas, Wireless Sensor Networks: An Information Processing Approach, Elsevier Science, 1st edition, 2005.

REFERENCE BOOKS:

1. C. Siva Ram Murthy, B.S. Manoj, Ad Hoc Wireless Networks: Architectures and Protocols, Pearson Education India, 1st edition, 2006.
2. Taieb Znati Kazem Sohraby, Daniel Minoli, Wireless Sensor Networks: Technology, Protocols and Applications, Wiley, 2010.

WEB LINKS:

1. https://nptel.ac.in/courses/106105160
2. https://www.coursera.org/courses?query=wireless
3. https://www.classcentral.com/course/swayam-wireless-ad-hoc-and-sensor-networks-7888
4. https://www.youtube.com/playlist?list=PLV8vIYTIdSnaoFjclogMhXiBFrHSL2Ar1

INTRUSION DETECTION SYSTEMS (Professional Elective – V)

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

Course Code: 22IT751PE

Prerequisites: Computer Networks, Computer Programming

Course Objectives:

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

Course Outcomes:

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

UNIT – I **[10 Lectures]**

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

UNIT – II **[10 Lectures]**

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

UNIT – III **[9 Lectures]**

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

UNIT – IV **[9 Lectures]**

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

UNIT – V **[10 Lectures]**

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

TEXT BOOKS:

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

REFERENCE BOOKS:

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

WEB LINKS:

1. https://www.intechopen.com/books/intrusion-detection-systems/
2. https://www.sans.org/course/intrusion-detection-in-depth
3. https://www.cybrary.it/skill-certification-course/ids-ips-certification-training-course

REAL TIME SYSTEMS (Professional Elective – V)

B.Tech. IV Year I Sem.

L	T	P	C
3	0	0	3

Course Code: 22IT752PE
Prerequisite: Basic Programming/C/C++ Programming, Computer Organization and Operating System

Course Objectives

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

Course Outcomes:

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

UNIT – I **[9 Lectures]**

Introduction: Introduction to UNIX/LINUX, Overview of Commands, File I/O,(open, create, close, lseek, read, write), Process Control (fork, vfork, exit, wait, waitpid, exec).

UNIT – II **[12 Lectures]**

Real Time Operating Systems: Brief History of OS, Defining RTOS, The Scheduler, Objects, Services, Characteristics of RTOS, Defining a Task, asks States and Scheduling, Task Operations, Structure, Synchronization, Communication and Concurrency. Defining Semaphores, Operations and Use, Defining Message Queue, States, Content, Storage, Operations and Use

UNIT – III **[9 Lectures]**

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

UNIT – IV **[10 Lectures]**

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

UNIT – V **[9 Lectures]**

Case Studies of RTOS: RT Linux, Micro C/OS-II, VxWorks, Embedded Linux, and Tiny OS.

TEXT BOOK:

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

REFERENCE BOOKS:

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

BLOCKCHAIN TECHNOLOGY (Professional Elective – V)

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

Course Code: 22IT753PE

Prerequisites:

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

Course Objectives:

1. To learn the fundamentals of Blockchain and various types of block chain and consensus mechanisms.
2. To understand the public block chain system and smart contracts.
3. To describe Private blockchain system, consortium blockchain and initial coin offering.
4. To know the security issues of blockchain technology.
5. To explore different blockchain application case studies.

Course Outcomes:

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

1. Summarize concepts behind crypto currency.
2. Design applications of smart contracts in decentralized applications and public block chain system.
3. Implement Private blockchain, hybrid blockchain and initial coin offering.
4. Solve the security issues in blockchain and develop applications in different domains.
5. Create blockchain for different application case studies

UNIT-I **[10 Lectures]**

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 **[9 Lectures]**

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 **[12 Lectures]**

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 **[9 Lectures]**

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

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

UNIT-V **[10 Lectures]**

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 BOOKS:

1. Chandramouli Subramanian, Asha A. George, Abhilasj K A and Meena Karthikeyan, Blockchain Technology, Universities Press, 2020.
2. Michael Juntao Yuan, Building Blockchain Apps, Pearson Education, First Edition, 2020.

REFERENCE BOOKS:

1. Melanie Swan, Blockchain: Blueprint for a New Economy, O'reilly, First Edition, 2015.
2. Jai Singh Arun, Jerry Cuomo, Nitin Gaur, Blockchain for Business, Pearson, 1st edition, 2019.

WEB LINKS:

1. https://nptel.ac.in/courses/106104220
2. https://www.coursera.org/courses?query=blockchain
3. https://www.udemy.com/topic/blockchain/
4. https://www.mygreatlearning.com/academy/learn-for-free/courses/blockchain-basics
5. https://www.edx.org/learn/blockchain

DEEP LEARNING (Professional Elective –V)

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

Course Code: 22IT754PE

Course Objectives:
1. To introduce deep Learning algorithms and their applications in real-world data.
2. To understand optimal usage of data for training deep models.
3. To learn CNN models on real-world data.
4. To explore RNN models for real-world data and evaluation of deep models.
5. To describe deep models for real-world problems.

Course Outcomes:
After completion of this course, the students will be able to:
1. Implement deep Learning algorithms and their applications in real-world data.
2. Create optimal usage of data for training deep models.
3. Apply CNN models for real-world data.
4. Create and Evaluate RNN models for real-world data.
5. Develop deep models for real-world problems.

UNIT -I **[10 Lectures]**

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 **[12 Lectures]**

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 **[8 Lectures]**

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 **[10 Lectures]**

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 **[9 Lectures]**

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 BOOKS:

1. Aaron Courville, Ian Goodfellow, Yoshua Bengio, Deep Learning, MIT Press, 2016.
2. Hastie, R. Tibshirani, and J. Friedman, The Elements of Statistical Learning, Springer-Verlag New York Inc., 2nd ed. 2009, Corr. 9th printing 2017 edition, 2017.

REFERENCE BOOKS:

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

WEB LINKS:

1. https://onlinecourses.nptel.ac.in/noc20_cs62/preview
2. https://www.coursera.org/specializations/deep-learning
3. https://www.udemy.com/topic/deep-learning/
4. https://www.mygreatlearning.com/deep-learning/free-courses
5. https://machinelearningmastery.com/deep-learning-courses/

SOFTWARE PROCESS & PROJECT MANAGEMENT (Professional Elective – V)

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

Course Code: 22IT755PE

Course Objectives:

1. To acquire knowledge of software process management and models.
2. To understand the process production stages.
3. To learn the process plan and its checkpoints.
4. To introduce managerial skills for software project development
5. To describe software economics, project policies and its next level adoption.

Course Outcomes:

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

1. Gain knowledge of software process change, assessment, project plans and Quality Standards.
2. Examine the life cycle phases, artifacts, and project organization.
3. Implement workflows and checkpoints of a process and its plan.
4. Design and develop software products using conventional and modern principles of software project management.
5. Identify the new project management process and practices.

UNIT – I **[10 Lectures]**

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 **[10 Lectures]**

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 **[9 Lectures]**

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 **[9 Lectures]**

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 **[9 Lectures]**

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

TEXT BOOKS:

1. Watts S. Humphrey, Managing the Software Process, Pearson Education, 1989.
2. Walker Royce, Software Project Management: A unified framework, Pearson Education, 1st edition, 2002.

REFERENCE BOOKS:

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

WEB LINKS:

1. https://onlinecourses.nptel.ac.in/noc19_cs70/preview
2. https://www.coursera.org/courses?query=software%20project%20management
3. https://www.udemy.com/course/software-project-management-the-complete-course/
4. https://www.classcentral.com/course/swayam-software-project-management-14294
5. https://www.forbes.com/advisor/in/business/software/free-project-management-software

SCRIPTING LANGUAGES (Open Elective – II)

B.Tech. IV Year I Sem.

L	T	P	C
3	0	0	3

Course Code: 22IT721OE

Prerequisites:

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

Course Objectives:

1. To understand the web service architecture and applications.
2. To introducing Ruby scripting.
3. To understand the basics of PERL programming.
4. To illustrate the advanced programming in PERL.
5. To Learn TCL.

Course Outcomes:

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

1. Comprehend the SOAP architecture and web services.
2. Extend the Ruby scripting language.
3. Apply the basic Perl programming language.
4. Implement the advanced programming in PERL.
5. Develop TCL programming and select an appropriate language for solving a given problem.

UNIT – I **[10 Lectures]**

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 **[8 Lectures]**

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 **[9 Lectures]**

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 **[9 Lectures]**

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 **[8 Lectures]**

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. David Barron, The World of Scripting Languages, Wiley Publications, 2009.
2. David Flanagan, Yukihiro Matsumoto, The Ruby Programming Language: Everything You Need to Know, O'Reilly Media, 2008.
3. Dave Thomas, Chad Fowler, Andy Hunt, Programming Ruby: The Pragmatic Programmers' Guide, Pragmatic Bookshelf, 2nd edition, 2004.

REFERENCE BOOKS:

1. J. Lee and B. Ware, Open Source Development with LAMP: Using Linux, Apache, MySQL, Perl, and PHP, Pearson Technology Group, 1st edition, 2002.
2. E. Quigley, Perl by Example, Prentice Hall, 5th edition, 2014.
3. Larry Wall, T. Christiansen and J. Orwant, Programming Perl, O'Reilly Media, 3rd edition, 2000.
4. Ousterhout, Tcl and the Tk Tool kit, Pearson Education, 2nd edition, 2010.

WEB LINKS:

1. https://archive.nptel.ac.in/courses/117/106/117106113/
2. https://www.udemy.com/course/scripting-for-beginner/
3. https://freevideolectures.com/blog/free-courses-learn-scripting-language/
4. https://techenclave.com/threads/programming-and-scripting-links-and-resources.1325/
5. https://www.autoitscript.com/site/autoit/
6. https://www.tutorialspoint.com/ruby/ruby_overview.htm

FULL STACK DEVELOPMENT (Open Elective – II)

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

Course Code: 22IT722OE

Pre-Requisites:

1. Object Oriented Programming
2. Web Technologies

Course Objectives:

To learn the core concepts of Full Stack Development.
2. To understand the front-end programming like Data, Files, Http Requests and Responses using node.js.
3. To know the storing and processing huge data and connects with NodeJS application using MangoDB.
4. To provide an in-depth study of Angular Services in Web Applications.
5. To explore interactive user interfaces with react structure.

Course Outcomes:

After completion of this course, the students will be able to:
1. Develop Full stack components for developing web application.
2. Apply packages of NodeJS to work with Data, Files, Http Requests and Responses.
3. Prepare MongoDB database for storing and processing huge data and connects with NodeJS application.
4. Design faster and effective single page applications using Express and Angular.
5. Create interactive user interfaces with react components.

UNIT-I **[10 Lectures]**

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 Applicati on, Understanding the Node.js Event Model, Adding Work to the Event Queue, Implementing Callbacks

UNIT-II **[8 Lectures]**

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 **[9 Lectures]**

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 **[9 Lectures]**

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 **[10 Lectures]**

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.

INFORMATION SECURITY LAB

B.Tech. IV Year I Sem.

L	T	P	C
0	0	2	1

Course Code: 22IT703PC

Course Objectives

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

Course Outcomes

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

List of Experiments:

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

TEXT BOOK:

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

REFERENCE BOOKS:

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

CLOUD COMPUTING LAB

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

Course Code: 22IT704PC

Course Objectives:

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

Course Outcomes:

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

List of Experiments:

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

TEXT BOOK:

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

REFERENCE BOOKS:

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

ORGANIZATIONAL BEHAVIOUR

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

Course Code: 22MB801HS

Course Objectives:

1. To learn the basic concepts of OB and its applications in contemporary organizations.
2. To understand how individual behavior impacts organizational effectiveness and efficiency.
3. To explore the importance of group behavior in the organization.
4. To empower nature in groups and teams.
5. To appreciate different cultures and diversity in the workplace.

Course Outcomes:

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

1. Impart the conceptual framework of the OB and its practical applications.
2. Evaluate the role of individual behavior in achieving organizational goals.
3. Usage of decision making at group level.
4. Comprehend power and politics in organization.
5. Accept and embrace working with different people from different cultures in the work place.

UNIT - I **[10 Lectures]**

Organizational Behaviour

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

UNIT - II **[10 Lectures]**

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 **[9 Lectures]**

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 **[8 Lectures]**

Leadership and Power

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

UNIT - V **[10 Lectures]**

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.

WEB LINKS:

1. https://onlinecourses.nptel.ac.in/noc20_mg51/preview
2. https://www.coursera.org/learn/managing-people-iese
3. https://www.udemy.com/course/organisational-behaviour/
4. https://www.classcentral.com/course/swayam-organizational-behaviour-17720
5. https://www.mygreatlearning.com/academy/learn-for-free/courses/organizational-behaviour

6.https://open.umn.edu/opentextbooks/textbooks/30

NATURAL LANGUAGE PROCESSING (Professional Elective – VI)

B.Tech. IV Year II Sem.

L	T	P	C
3	0	0	3

Course Code: 22IT861PE

Prerequisites:

1. A Course on "Data structures".
2. A Course on "finite automata and probability theory".

Course Objectives

1. To Introduce the NLP structure of documents.
2. To understand the experimental methodology for empirical syntax analysis.
3. To describe the parsing structure of NLP systems.
4. To design NLP Algorithms.
5. To use language modelling techniques.

Course Outcomes:

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

1. Summarize the NLP structure documents.
2. Use the proper experimental methodology for evaluating NLP systems.
3. Construct statistical models over strings, trees and estimate parameters using supervised and unsupervised training methods.
4. Implement the NLP algorithms.
5. Design different language modelling Techniques

UNIT – I **[10 Lectures]**

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 **[9 Lectures]**

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

UNIT – III **[8 Lectures]**

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

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

UNIT – IV **[7 Lectures]**

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

UNIT – V **[9 Lectures]**

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. Daniel M. Bikel and Imed Zitouni, Multilingual natural Language Processing Applications: From Theory to Practice,
 Pearson Education India, First Edition, 2012.
2. Tanveer Siddiqui, U.S.Tiwary, Natural Language Processing and Information Retrieval, Oxford University Press, 2008.

REFERENCE BOOKS:

1. Daniel Jurafsky & James H Martin, Speech and Language Processing An Introduction to Natural Language Processing,
 Computational Linguistics, and Speech Recognition, PEARSON INDIA, 2nd edition, 2014.
2. Sowmya Vajjala, Bodhisattwa Majumder, Anuj Gupta, Harshit Surana, Practical Natural Language Processing: A Comprehensive Guide to Building Real-World NLP Systems, O'Reilly Media, 1st edition, 2020.

WEB LINKS:

1. https://nptel.ac.in/courses/106105158
2. https://www.coursera.org/specializations/natural-language-processing
3. https://www.coursera.org/courses?query=nlp
4. https://www.udemy.com/topic/natural-language-processing/
5. https://www.kaggle.com/code/kmldas/free-nlp-resources-courses

DISTRIBUTED SYSTEMS (Professional Elective –VI)

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

Course Code: 22IT862PE

Prerequisites

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

Course Objectives

1. To provide an insight into Distributed systems.
2. To introduce operating systems and distributed file systems.
3. To understand Peer to Peer Systems, time and global state, coordination, and agreement.
4. To learn Transactions and Concurrency control, distributed transactions and distributed deadlocks.
5. To describe Replication and Distributed shared memory.

Course Outcomes:

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

1. Gain the knowledge of Distributed systems.
2. Explore operating systems and distributed file systems.
3. Implement Peer to Peer Systems, time and global state, coordination, and agreement.
4. Apply Transactions and Concurrency control, distributed transactions, and distributed deadlocks.
5. Create Replication and Distributed shared memory.

UNIT – I **[10 Lectures]**

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 **[8 Lectures]**

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

Distributed File Systems-Introduction, File Service architecture.

UNIT – III **[11 Lectures]**

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 **[10 Lectures]**

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 **[9 Lectures]**

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. G Coulouris, J Dollimore and T Kindberg, Distributed Systems Concepts and Design, Fourth Edition, Pearson Education, 2008.
2. S.Ghosh, Chapman & Hall/CRC, Distributed Systems, Taylor & Francis Group, 2010.

REFERENCE BOOKS:

1. A.S. Tanenbaum and M.V. Steen, Distributed Systems – Principles and Paradigms, Pearson Education, 2nd edition, 2008.
2. Ajay D. Kshemakalyani and Mukesh Singhal, Distributed Computing, Principles, Algorithms and Systems, Cambridge, rp 2010.

WEB LINKS:

1. https://onlinecourses.nptel.ac.in/noc21_cs87/preview
2. https://www.coursera.org/courses?query=distributed%20systems
3. https://www.udemy.com/topic/distributed-computing/
4. https://github.com/FedericoPonzi/awesome-distributed-systems
5. https://www.classcentral.com/course/swayam-distributed-systems-43583

AUGMENTED REALITY & VIRTUAL REALITY (Professional Elective –VI)

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

Course Code: 22IT863PE

Course Objective:

1. To gain the knowledge of historical and modern overviews and perspectives on virtual reality.
2. To learn the fundamentals of sensation, perception, and perceptual training.
3. To have the scientific, technical, and engineering aspects of augmented and virtual reality systems.
4. To learn the Evaluation of virtual reality from the lens of design.
5. To learn the technology of augmented reality and implement it to have practical knowledge.

Course Outcomes:

1. Identify, examine, and develop software that reflects fundamental techniques for the design and d eployment of VR and AR experiences.
2. Describe how VR and AR systems work.
3. Choose, develop, explain, and defend the use of particular designs for AR and VR experiences.
4. Evaluate the benefits and drawbacks of specific AR and VR techniques on the human body.
5. Identify and examine state-of-the-art AR and VR design problems and solutions from the industry and academia.

UNIT – I **[10 Lectures]**

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

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

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

UNIT – II **[10 Lectures]**

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

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

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

UNIT – III **[9 Lectures]**

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

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

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

UNIT – IV **[10 Lectures]**

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

Visual Perception: Visual Perception - Perception of Depth, Perception of Motion, Perception of Color **Visual Rendering:** Visual Rendering -Ray Tracing and Shading Models, Rasterization, Correcting Optical Distortions, Improving Latency and Frame Rates, Immersive Photos and Videos

UNIT – V **[9 Lectures]**

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

Interaction: Motor Programs and Remapping, Locomotion, Social Interaction

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

TEXT BOOKS:

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

REFERENCE BOOKS:

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

WEB LINKS:

1. https://www.teamviewer.com/en/augmented-reality-ar-vs-virtual-reality-vr/
2. https://www.xmreality.com/blog/augmented-reality-vs-virtual-reality
3. https://www.techtarget.com/whatis/definition/augmented-reality-AR

WEB SECURITY (Professional Elective –VI)

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

Course Code: 22IT864PE

Course Objectives:

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

Course Outcomes: Students should be able to

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

UNIT – I **[10 Lectures]**

The Web Security, The Web Security Problem, Risk Analysis and Best Practices

Cryptography and the Web: Cryptography and Web Security, Working Cryptographic Systems and Protocols, Legal Restrictions on Cryptography, Digital Identification

UNIT – II **[9 Lectures]**

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

UNIT – III **[8 Lectures]**

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

UNIT – IV **[9 Lectures]**

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

UNIT – V **[10 Lectures]**

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

TEXT BOOKS:

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

CYBER FORENSICS (Professional Elective –VI)

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

Course Code: 22IT865PE
Prerequisites: Network Security.

Course Objectives:
1. To understand the types of crimes and response methodology.
2. To learn the objectives of providing digital evidence which is obtained from digital media.
3. To describe the data analysis and validation methods.
4. To explore the modern tools used in an investigation.
5. To retrieve data stored in various storage places

Course Outcomes:

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

1. Summarize the crime types and incident response procedures.
2. Apply Forensic Duplication Tools.
3. Explore data analysis and visualization techniques.
4. Use various forensic tools for a wide variety of investigations.
5. Design principles of data management methods.

UNIT- I **[10 Lectures]**

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 **[10 Lectures]**
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 **[8 Lectures]**
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 **[9 Lectures]**
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 **[9 Lectures]**
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. John R. Vacca, Computer Forensics: Computer Crime Scene Investigation, Laxmi Publications, First Edition, 2015.
3. Nelson, Phillips Enfinger, Steuart, Guide To Computer Forensics and Investigations - Standalone, CENGAGE Learning, 6th edition, 2018.

REFERENCE BOOKS:

1. Keith J. Jones, Richard Bejtiich, Curtis W. Rose, Real Digital Forensics: Computer Security and Incident Response, Addison- Wesley Pearson Education, 1st edition, 2005.
2. Tony Sammes and Brian Jenkinson, Forensic Computing: A Practitioner's Guide, Springer London Ltd, 200 0.

WEB LINKS:

1. https://onlinecourses.swayam2.ac.in/cec21_ge10/preview
2. https://www.coursera.org/courses?query=computer%20forensics
3. https://www.udemy.com/topic/computer-forensics/
4. https://www.mygreatlearning.com/academy/learn-for-free/courses/cyber-forensics
5. https://resources.infosecinstitute.com/topics/digital-forensics/free-open-source-computer-forensics-tools/

DEVOPS (Open Elective – III)

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

Course Code: 22IT831OE

Pre-Requisites:

1. Software Engineering
2. Software Project Management

Course Objectives:

1. To describe the agile relationship between development and IT operations.
2. To know the skill sets and high-functioning teams involved in DevOps and related methods to reach a continuous delivery capability.
3. To explore project management tools.
4. To understand system integration.
5. To impart testing tools, automation, and deployment of the system.

Course Outcomes:

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

1. Explore various components of Devops environment.
2. Identify Software development models and architectures of DevOps.
3. Apply different project management models.
4. Investigate different DevOps Software development models, integration, Collaborate and adopt Devops in r eal-time projects.
5. Assess various Devops practices, testing and code deployment tools.

UNIT- I **[9 Lectures]**

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: **[9 Lectures]**

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 **[9 Lectures]**

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 **[9 Lectures]**

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 **[10 Lectures]**

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, SaltStackand Docker.

TEXT BOOKS:

1. Joakim Verona, Practical Devops, Second Edition, Ingram short title, 2nd edition, 2018.
2. Deepak Gaikwad, Viral Thakkar, DevOps Tools from Practitioner's Viewpoint, Wiley publications, 2019.

REFERENCE BOOKS:

1. Len Bass, Ingo Weber, Liming Zhu, DevOps: A Software Architect's Perspective, Addison-Wesley Professional, 1st edition, 2015.
2. Gene Kim, Jez Humble, Patrick Debois, John Willis, Nicole Forsgren, The Devops Handbook: How to Create World-Class Agility, Reliability, & Security in Technology Organizations, It Revolution Press, 2nd edition, 2021.

WEB LINKS:

1. https://nptel.ac.in/courses/128106012
2. https://elearn.nptel.ac.in/shop/iit-workshops/completed/azure-devops-and-micro-services-azure-kubernetes-deployment-models/
3. https://www.coursera.org/learn/intro-to-devops
4. https://www.udemy.com/course/devops-fundamentals-for-beginners/
5. https://dev.to/javinpaul/my-favorite-free-courses-to-learn-devops-in-2023-4h6m
6. https://www.udemy.com/topic/DevOps/?utm_source=adwords&utm_medium=udemyads&utm_campaign =Brand-Topic_la.EN_cc.INDIA&utm_content=deal4584&utm_term=
7. https://www.techtarget.com/whatis/feature/9-best-free-DevOps-certifications-and-training-courses

BIG DATA TECHNOLOGIES (Open Elective –III)

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

Course Code: 22IT832OE

Course Objectives

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

Courses Outcomes

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

UNIT – I **[10 Lectures]**

Getting an Overview of Big Data

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

Technologies for Handling Big Data

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

UNIT – II **[9 Lectures]**

Understanding Hadoop Ecosystem

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

Understanding MapReduce Fundamentals and HBase

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

UNIT – III **[8 Lectures]**

Exploring Hive

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

Analyzing Data with Pig

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

UNIT – IV **[10 Lectures]**

Using Oozie

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

NoSQL Data Management

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

UNIT – V **[10 Lectures]**

ZooKeeper: Installing and Running ZooKeeper, An Example, Group Membership in ZooKeeper, Creating the Group, Joining a Group, Listing Members in a Group, The ZooKeeper Service, Data Model, Operations, Implementation, Consistency, Sessions, Building Applications with ZooKeeper, A Configuration, Service, The Resilient ZooKeeper Application, A Lock Service, More Distributed Data Structures and Protocols, ZooKeeper in Production

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

TEXT BOOKS:

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

REFERENCE BOOKS:

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