# JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY KAKINADA
## KAKINADA – 533 003, Andhra Pradesh, India
### CSE (DS) (R23-IInd YEAR COURSE STRUCTURE & SYLLABUS)

**B. Tech – II Year I Semester**

| S.No. | Category | Title | L | T | P | Credits |
|---|---|---|---|---|---|---|
| 1 | BS&H | Discrete Mathematics & Graph Theory | 3 | 0 | 0 | 3 |
| 2 | BS&H | Universal human values – understanding harmony and Ethical human conduct | 2 | 1 | 0 | 3 |
| 3 | Engineering Science | Introduction to Data Science | 3 | 0 | 0 | 3 |
| 4 | Professional Core | Advanced Data Structures &Algorithms Analysis | 3 | 0 | 0 | 3 |
| 5 | Professional Core | Object-Oriented Programming Through JAVA | 3 | 0 | 0 | 3 |
| 6 | Professional Core | Data Science Lab | 0 | 0 | 3 | 1.5 |
| 7 | Professional Core | Object-Oriented Programming Through JAVA Lab | 0 | 0 | 3 | 1.5 |
| 8 | Skill Enhancement course | Python programming | 0 | 1 | 2 | 2 |
| 9 | Audit Course | Environmental Science | 2 | 0 | 0 | - |
| **Total** | | | **15** | **2** | **10** | **20** |

**B. Tech– II Year II Semester**

| S.No. | Category | Title | L | T | P | Credits |
|---|---|---|---|---|---|---|
| 1 | Management Course- I | Optimization Techniques | 2 | 0 | 0 | 2 |
| 2 | Engineering Science/ Basic Science | Statistical methods for Data science | 3 | 0 | 0 | 3 |
| 3 | Professional Core | Data Engineering | 3 | 0 | 0 | 3 |
| 4 | Professional Core | DBMS | 3 | 0 | 0 | 3 |
| 5 | Professional Core | Digital Logic and Computer Organization | 3 | 0 | 0 | 3 |
| 7 | Professional Core | Data Engineering Lab | 0 | 0 | 3 | 1.5 |
| 8 | Professional Core | DBMS Lab | 0 | 0 | 3 | 1.5 |
| 9 | Skill Enhancement course | Exploratory Data Analysis with Python | 0 | 1 | 2 | 2 |
| 10 | BS&H | Design Thinking &Innovation | 1 | 0 | 2 | 2 |
| **Total** | | | **15** | **1** | **10** | **21** |
| Mandatory Community Service Project Internship of 08 weeks duration during summer Vacation | | | | | | |

**II Year I Semester**

| L | T | P | C |
|---|---|---|---|
| 3 | 0 | 0 | 3 |

## DISCRETE MATHEMATICS AND GRAPH THEORY

**Course Objectives:**

- To introduce the students to the topics and techniques of discrete methods and combinatorial reasoning.
- To introduce a wide variety of applications. The algorithmic approach to the solution of problems is fundamental in discrete mathematics, and this approach reinforces the close ties between this discipline and the area of computer science.

**Course Outcomes:** At the end of the course students will be able to
1. Build skills in solving mathematical problems (L3)
2. Comprehend mathematical principles and logic (L4)
3. Demonstrate knowledge of mathematical modeling and proficiency in using mathematical software (L6)
4. Manipulate and analyze data numerically and/or graphicallysing appropriate Software (L3)
5. How to communicate effectively mathematical ideas/results verbally or in writing (L1)

### UNIT–I: Mathematical Logic:

Propositional Calculus: Statements and Notations, Connectives, Well Formed Formulas, Truth Tables, Tautologies, Equivalence of Formulas, Duality Law, Tautological Implications, Normal Forms, Theory of Inference for Statement Calculus, Consistency of Premises, Indirect Method ofProof, Predicate Calculus: Predicates, Predicative Logic, Statement Functions, Variables and Quantifiers, Free and Bound Variables, Inference Theory for Predicate Calculus.

### UNIT-II: Set Theory:
Sets: Operations on Sets, Principle of Inclusion-Exclusion, Relations: Properties, Operations, Partition and Covering, Transitive Closure, Equivalence, Compatibility and Partial Ordering, Hasse Diagrams, Functions: Bijective, Composition, Inverse, Permutation, and Recursive Functions, Lattice and its Properties.

### UNIT-III: Combinatorics and Recurrence Relations:
Basis of Counting, Permutations, Permutations withRepetitions, Circular and Restricted Permutations, Combinations, RestrictedCombinations, Binomial and Multinomial Coefficients and Theorems.

### Recurrence Relations:
Generating Functions, Function of Sequences, Partial Fractions, Calculating Coefficient of Generating Functions, Recurrence Relations, Formulation as Recurrence Relations, Solving Recurrence Relations by Substitution and Generating Functions, Method of Characteristic

Roots, Solving Inhomogeneous Recurrence Relations

## UNIT-IV: Graph Theory:

Basic Concepts, Graph Theory and its Applications, Subgraphs, Graph Representations: Adjacency and Incidence Matrices, Isomorphic Graphs, Paths and Circuits, Eulerian and Hamiltonian Graphs,

## Unit-V: Multi Graphs

Multigraphs, Bipartite and Planar Graphs, Euler's Theorem, Graph Colouring and Covering, Chromatic Number, Spanning Trees, Prim's and Kruskal's Algorithms, BFS and DFS Spanning Trees.

## TEXT BOOKS:

1. Discrete Mathematical Structures with Applications to Computer Science, J. P. Tremblay and P. Manohar, Tata McGraw Hill.
2. Elements of Discrete Mathematics-A Computer Oriented Approach, C. L.Liu and D. P. Mohapatra, 3rd Edition, Tata McGraw Hill.
3. Theory and Problems of Discrete Mathematics, Schaum's Outline Series, Seymour Lipschutz and Marc Lars Lipson, 3rd Edition, McGraw Hill.

## REFERENCE BOOKS:

1. Discrete Mathematics for Computer Scientists and Mathematicians, J. L.Mott, A. Kandel and T. P. Baker, 2nd Edition, Prentice Hall of India.
2. Discrete Mathematical Structures, Bernand Kolman, Robert C. Busby andSharon Cutler Ross, PHI.
3. Discrete Mathematics, S. K. Chakraborthy and B.K. Sarkar, Oxford, 2011.
4. Discrete Mathematics and its Applications with Combinatorics and GraphTheory, K. H. Rosen, 7th Edition, Tata McGraw Hill.

**II Year I Semester**

| L | T | P | C |
|---|---|---|---|
| 2 | 1 | 0 | 3 |

## UNIVERSAL HUMAN VALUES – UNDERSTANDING HARMONY AND ETHICAL HUMAN CONDUCT

**Course Objectives:**
- To help the students appreciate the essential complementary between 'VALUES' and 'SKILLS' to ensure sustained happiness and prosperity which are the core aspirations of all human beings.
- To facilitate the development of a Holistic perspective among students towards life and profession as well as towards happiness and prosperity based on a correct understanding of the Human reality and the rest of existence. Such holistic perspective forms the basis of Universal Human Values and movement towards value-based living in a natural way.
- To highlight plausible implications of such a Holistic understanding in terms of ethical human conduct, trustful and mutually fulfilling human behaviour and mutually enriching interaction with Nature.

**Course Outcomes:**
- Define the terms like Natural Acceptance, Happiness and Prosperity (L1, L2)
- Identify one's self, and one's surroundings (family, society nature) (L1, L2)
- Apply what they have learnt to their own self in different day-to-day settings in real life (L3)
- Relate human values with human relationship and human society. (L4)
- Justify the need for universal human values and harmonious existence (L5)
- Develop as socially and ecologically responsible engineers (L3, L6)

**Course Topics**

The course has 28 lectures and 14 tutorials in 5 modules. The lectures and tutorials are of 1-hour duration. Tutorial sessions are to be used to explore and practice what has been proposed during the lecture sessions.

The Teacher's Manual provides the outline for lectures as well as practice sessions. The teacher is expected to present the issues to be discussed as propositions and encourage the students to have a dialogue.

**UNIT I** Introduction to Value Education (6 lectures and 3 tutorials for practice session)

Lecture 1: Right Understanding, Relationship and Physical Facility (Holistic Development and the Role of Education)
Lecture 2: Understanding Value Education
Tutorial 1: Practice Session PS1 Sharing about Oneself

Lecture 3: self-exploration as the Process for Value Education

Lecture4: Continuous Happiness and Prosperity – the Basic Human Aspirations

Tutorial 2: Practice Session PS2 Exploring Human Consciousness

Lecture 5: Happiness and Prosperity – Current Scenario

Lecture 6: Method to Fulfill the Basic Human Aspirations

Tutorial 3: Practice Session PS3 Exploring Natural Acceptance

**UNIT II**  Harmony in the Human Being (6 lectures and 3 tutorials for practice session)

Lecture 7: Understanding Human being as the Co-existence of the self and the body.

Lecture 8: Distinguishing between the Needs of the self and the body

Tutorial 4: Practice Session PS4 Exploring the difference of Needs of self and body.

Lecture 9: The body as an Instrument of the self

Lecture 10: Understanding Harmony in the self

Tutorial 5: Practice Session PS5 Exploring Sources of Imagination in the self

Lecture 11: Harmony of the self with the body

Lecture 12: Programme to ensure self-regulation and Health

Tutorial 6: Practice Session PS6 Exploring Harmony of self with the body

**UNIT III**  Harmony in the Family and Society (6 lectures and 3 tutorials for practice session)

Lecture 13: Harmony in the Family – the Basic Unit of Human Interaction

Lecture 14: 'Trust' – the Foundational Value in Relationship

Tutorial 7: Practice Session PS7 Exploring the Feeling of Trust

Lecture 15: 'Respect' – as the Right Evaluation

Tutorial 8: Practice Session PS8 Exploring the Feeling of Respect

Lecture 16: Other Feelings, Justice in Human-to-Human Relationship

Lecture 17: Understanding Harmony in the Society

Lecture 18: Vision for the Universal Human Order

Tutorial 9: Practice Session PS9 Exploring Systems to fulfil Human Goal

**UNIT IV**  Harmony in the Nature/Existence (4 lectures and 2 tutorials for practice session)

Lecture 19: Understanding Harmony in the Nature

Lecture 20: Interconnectedness, self-regulation and Mutual Fulfilment among the Four Orders of Nature

Tutorial 10: Practice Session PS10 Exploring the Four Orders of Nature

Lecture 21: Realizing Existence as Co-existence at All Levels

Lecture 22: The Holistic Perception of Harmony in Existence

Tutorial 11: Practice Session PS11 Exploring Co-existence in Existence.

**UNIT V**    Implications of the Holistic Understanding – a Look at Professional Ethics (6 lectures and 3 tutorials for practice session)
Lecture 23: Natural Acceptance of Human Values
Lecture 24: Definitiveness of (Ethical) Human Conduct
Tutorial 12: Practice Session PS12 Exploring Ethical Human Conduct
Lecture 25: A Basis for Humanistic Education, Humanistic Constitution and Universal Human Order
Lecture 26: Competence in Professional Ethics
Tutorial 13: Practice Session PS13 Exploring Humanistic Models in Education
Lecture 27: Holistic Technologies, Production Systems and Management Models-Typical Case Studies
Lecture 28: Strategies for Transition towards Value-based Life and Profession
Tutorial 14: Practice Session PS14 Exploring Steps of Transition towards Universal Human Order

Practice Sessions for UNIT I – Introduction to Value Education
PS1 Sharing about Oneself
PS2 Exploring Human Consciousness
PS3 Exploring Natural Acceptance

Practice Sessions for UNIT II – Harmony in the Human Being
PS4 Exploring the difference of Needs of self and body
PS5 Exploring Sources of Imagination in the self
PS6 Exploring Harmony of self with the body

Practice Sessions for UNIT III – Harmony in the Family and Society
PS7 Exploring the Feeling of Trust
PS8 Exploring the Feeling of Respect
PS9 Exploring Systems to fulfil Human Goal

Practice Sessions for UNIT IV – Harmony in the Nature (Existence)
PS10 Exploring the Four Orders of Nature
PS11 Exploring Co-existence in Existence

Practice Sessions for UNIT V – Implications of the Holistic Understanding – a Look at Professional Ethics
PS12 Exploring Ethical Human Conduct
PS13 Exploring Humanistic Models in Education
PS14 Exploring Steps of Transition towards Universal Human Order

**READINGS:**

**Textbook and Teachers Manual**

a. The Textbook

R R Gaur, R Asthana, G P Bagaria, *A Foundation Course in Human Values and Professional Ethics*, 2nd Revised Edition, Excel Books, New Delhi, 2019. ISBN 978-93-87034-47-1

b. The Teacher's Manual

R R Gaur, R Asthana, G P Bagaria,*Teachers' Manual for A Foundation Course in Human Values and Professional Ethics,* 2nd Revised Edition, Excel Books, New Delhi, 2019. ISBN 978-93-87034-53-2

**Reference Books**

1. *JeevanVidya: EkParichaya,* A Nagaraj, JeevanVidyaPrakashan, Amarkantak, 1999.
2. *Human Values*, A.N. Tripathi, New Age Intl. Publishers, New Delhi, 2004.
3. *The Story of Stuff* (Book).
4. *The Story of My Experiments with Truth* - by Mohandas Karamchand Gandhi
5. *Small is Beautiful* - E. F Schumacher.
6. *Slow is Beautifu*l - Cecile Andrews
7. *Economy of Permanence* - J C Kumarappa
8. *Bharat Mein Angreji Raj* – PanditSunderlal
9. *Rediscovering India* - by Dharampal
10. *Hind Swaraj or Indian Home Rule* - by Mohandas K. Gandhi
11. *India Wins Freedom* - Maulana Abdul Kalam Azad
12. *Vivekananda* - Romain Rolland (English)
13. *Gandhi* - Romain Rolland (English)

**Mode of Conduct:**

Lecture hours are to be used for interactive discussion, placing the proposals about the topics at hand and motivating students to reflect, explore and verify them.

Tutorial hours are to be used for practice sessions.

While analyzing and discussing the topic, the faculty mentor's role is in pointing to essential elements to help in sorting them out from the surface elements. In other words, help the students explore the important or critical elements.

In the discussions, particularly during practice sessions (tutorials), the mentor encourages the student to connect with one's own self and do self-observation, self-reflection and self-exploration.

Scenarios may be used to initiate discussion. The student is encouraged to take up "ordinary" situations rather than" extra-ordinary" situations. Such observations and their analyses are shared and discussed with other students and faculty mentor, in a group sitting.

Tutorials (experiments or practical) are important for the course. The difference is that the laboratory is everyday life, and practical are how you behave and work in real life. Depending on the nature of topics, worksheets, home assignment and/or activity are included. The practice sessions (tutorials) would also provide support to a student in performing actions

commensurate to his/her beliefs. It is intended that this would lead to development of commitment, namely behaving and working based on basic human values.

It is recommended that this content be placed before the student as it is, in the form of a basic foundation course, without including anything else or excluding any part of this content. Additional content may be offered in separate, higher courses. This course is to be taught by faculty from every teaching department, not exclusively by any one department.

Teacher preparation with a minimum exposure to at least one 8-day Faculty Development Program on Universal Human Values is deemed essential.

**Online Resources:**

1. https://fdp-si.aicte-india.org/UHV-II%20Class%20Notes%20&%20Handouts/UHV%20Handout%201-Introduction%20to%20Value%20Education.pdf

2. https://fdp-si.aicte-india.org/UHV-II%20Class%20Notes%20&%20Handouts/UHV%20Handout%202-Harmony%20in%20the%20Human%20Being.pdf

3. https://fdp-si.aicte-india.org/UHV-II%20Class%20Notes%20&%20Handouts/UHV%20Handout%203-Harmony%20in%20the%20Family.pdf

4. https://fdp-si.aicte-india.org/UHV%201%20Teaching%20Material/D3-S2%20Respect%20July%202023.pdf

5. https://fdp-si.aicte-india.org/UHV-II%20Class%20Notes%20&%20Handouts/UHV%20Handout%205-Harmony%20in%20the%20Nature%20and%20Existence.pdf

6. https://fdp-si.aicte-india.org/download/FDPTeachingMaterial/3-days%20FDP-SI%20UHV%20Teaching%20Material/Day%203%20Handouts/UHV%203D%20D3-S2A%20Und%20Nature-Existence.pdf

7. https://fdp-si.aicte-india.org/UHV%20II%20Teaching%20Material/UHV%20II%20Lecture%202023-25%20Ethics%20v1.pdf

8. https://www.studocu.com/in/document/kiet-group-of-institutions/universal-human-values/chapter-5-holistic-understanding-of-harmony-on-professional-ethics/62490385

9. https://onlinecourses.swayam2.ac.in/aic22_ge23/preview

**II Year I Semester**

| L | T | P | C |
|---|---|---|---|
| 3 | 0 | 0 | 3 |

## INTRODUCTION TO DATA SCIENCE

**COURSE OBJECTIVES**: From the course the student will learn

1. Knowledge and expertise to become a data scientist.
2. Essential concepts of statistics and machine learning that are vital for data science;
3. Significance of exploratory data analysis (EDA) in data science.
4. Critically evaluate data visualizations presented on the dashboards
5. Suitability and limitations of tools and techniques related to data science process

**UNIT I:** Introduction to Data science, benefits and uses, facets of data, data science process in brief, big data ecosystem and data science

**Data Science process:** Overview, defining goals and creating project charter, retrieving data, cleansing, integrating and transforming data, exploratory analysis, model building, presenting findings and building applications on top of them

**Unit II:** Applications of machine learning in Data science, role of ML in DS, Python tools like sklearn, modelling process for feature engineering, model selection, validation and prediction, types of ML, semi-supervised learning

**Handling large data:** problems and general techniques for handling large data, programming tips for dealing large data, case studies on DS projects for predicting malicious URLs, for building recommender systems

**UNIT III: NoSQL movement for handling Bigdata:** Distributing data storage and processing with Hadoop framework, case study on risk assessment for loan sanctioning, ACID principle of relational databases, CAP theorem, base principle of NoSQL databases, types of NoSQL databases, case study on disease diagnosis and profiling

**UNIT IV: Tools and Applications of Data Science:** Introducing **Neo4j** for dealing with graph databases, graph query language **Cypher**, Applications graph databases, Python libraries like nltk and SQLite for handling Text mining and analytics, case study on classifying Reddit posts

**UNIT V: Data Visualization and Prototype Application Development:** Data Visualization options, Crossfilter, the JavaScript MapReduce library, Creating an interactive dashboard with dc.js, Dashboard development tools.

Applying the Data Science process for real world problem solving scenarios as a detailed case study.

**Textbook:**

1) Davy Cielen, Arno D.B.Meysman, and Mohamed Ali, "Introducing to Data Science using Python tools", Manning Publications Co, Dreamtech press, 2016
2) Prateek Gupta, "Data Science with Jupyter" BPB publishers, 2019 for basics

**Reference Books**:

1) Joel Grus, "Data Science From Scratch", OReilly, 2019
2) Doing Data Science: Straight Talk From The Frontline, 1 st Edition, Cathy O'Neil and Rachel Schutt, O'Reilly, 2013

| L | T | P | C |
|---|---|---|---|
| 3 | 0 | 0 | 3 |

**II Year I Semester**

## ADVANCED DATA STRUCTURES & ALGORITHM ANALYSIS

**Course Objectives:**
The main objectives of the course is to
- provide knowledge on advance data structures frequently used in Computer Science domain
- Develop skills in algorithm design techniques popularly used
- Understand the use of various data structures in the algorithm design

**UNIT – I:**
Introduction to Algorithm Analysis, Space and Time Complexity analysis, Asymptotic Notations.
AVL Trees – Creation, Insertion, Deletion operations and Applications
B-Trees – Creation, Insertion, Deletion operations and Applications

**UNIT – II:**
Heap Trees (Priority Queues) – Min and Max Heaps, Operations and Applications
Graphs – Terminology, Representations, Basic Search and Traversals, Connected Components and Biconnected Components, applications
Divide and Conquer: The General Method, Quick Sort, Merge Sort, Strassen's matrix multiplication, Convex Hull

**UNIT – III:**
Greedy Method: General Method, Job Sequencing with deadlines, Knapsack Problem, Minimum cost spanning trees, Single Source Shortest Paths
Dynamic Programming: General Method, All pairs shortest paths, Single Source Shortest Paths – General Weights (Bellman Ford Algorithm), Optimal Binary Search Trees, 0/1 Knapsack, String Editing, Travelling Salesperson problem

**UNIT – IV:**
Backtracking: General Method, 8-Queens Problem, Sum of Subsets problem, Graph Coloring, 0/1 Knapsack Problem
Branch and Bound: The General Method, 0/1 Knapsack Problem, Travelling Salesperson problem

**UNIT – V:**
NP Hard and NP Complete Problems: Basic Concepts, Cook's theorem
NP Hard Graph Problems: Clique Decision Problem (CDP), Chromatic Number Decision Problem (CNDP), Traveling Salesperson Decision Problem (TSP)
NP Hard Scheduling Problems: Scheduling Identical Processors, Job Shop Scheduling

**Textbooks:**
1. Fundamentals of Data Structures in C++, Horowitz, Ellis; Sahni, Sartaj; Mehta, Dinesh 2nd Edition Universities Press
2. Computer Algorithms/C++ Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran 2nd Edition University Press

**Reference Books:**
1. Data Structures and program design in C, Robert Kruse, Pearson Education Asia
2. An introduction to Data Structures with applications, Trembley & Sorenson, McGraw Hill
3. The Art of Computer Programming, Vol.1: Fundamental Algorithms, Donald E Knuth, Addison-Wesley, 1997.
4. Data Structures using C & C++: Langsam, Augenstein & Tanenbaum, Pearson, 1995
5. Algorithms + Data Structures & Programs:, N.Wirth, PHI
6. Fundamentals of Data Structures in C++: Horowitz Sahni & Mehta, Galgottia Pub.
7. Data structures in Java:, Thomas Standish, Pearson Education Asia

**Online Learning Resources:**
1. https://www.tutorialspoint.com/advanced_data_structures/index.asp
2. http://peterindia.net/Algorithms.html
3. Abdul Bari, 1. Introduction to Algorithms (youtube.com)

**II Year I Semester**

| L | T | P | C |
|---|---|---|---|
| 3 | 0 | 0 | 3 |

## OBJECT-ORIENTED PROGRAMMING THROUGH JAVA

**Course Objectives:**

The learning objectives of this course are to:

- Identify Java language components and how they work together in applications
- Learn the fundamentals of object-oriented programming in Java, including defining classes, invoking methods, using class libraries.
- Learn how to extend Java classes with inheritance and dynamic binding and how to use exception handling in Java applications
- Understand how to design applications with threads in Java
- Understand how to use Java APIs for program development

**UNIT I: Object Oriented Programming:** Basic concepts, Principles, Program Structure in Java: Introduction, Writing Simple Java Programs, Elements or Tokens in Java Programs, Java Statements, Command Line Arguments, User Input to Programs, Escape Sequences Comments, Programming Style.

**Data Types**, Variables, and Operators :Introduction, Data Types in Java, Declaration of Variables, Data Types, Type Casting, Scope of Variable Identifier, Literal Constants, Symbolic Constants, Formatted Output with printf() Method, Static Variables and Methods, Attribute Final,  **Introduction to Operators**, Precedence and Associativity of Operators, Assignment Operator ( = ), Basic Arithmetic Operators, Increment (++) and Decrement (- -) Operators, Ternary Operator, Relational Operators, Boolean Logical Operators, Bitwise Logical Operators.

**Control Statements**: Introduction, if Expression, Nested if Expressions, if–else Expressions, Ternary Operator?:, Switch Statement, Iteration Statements, while Expression, do–while Loop, for Loop, Nested for Loop, For–Each for Loop, Break Statement, Continue Statement.

**UNIT II: Classes and Objects:** Introduction, Class Declaration and Modifiers, Class Members, Declaration of Class Objects, Assigning One Object to Another, Access Control for Class Members, Accessing Private Members of Class, Constructor Methods for Class, Overloaded Constructor Methods, Nested Classes, Final Class and Methods, Passing Arguments by Value and by Reference, Keyword this.

**Methods:** Introduction, Defining Methods, Overloaded Methods, Overloaded Constructor Methods, Class Objects as Parameters in Methods, Access Control, Recursive Methods, Nesting of Methods, Overriding Methods, Attributes Final and Static.

**UNIT III: Arrays:**Introduction, Declaration and Initialization of Arrays, Storage of Array in Computer Memory, Accessing Elements of Arrays, Operations on Array Elements, Assigning Array to Another Array, Dynamic Change of Array Size, Sorting of Arrays, Search for

Values in Arrays, Class Arrays, Two-dimensional Arrays, Arrays of Varying Lengths, Three-dimensional Arrays, Arrays as Vectors.

**Inheritance:** Introduction, Process of Inheritance, Types of Inheritances, Universal Super Class-Object Class, Inhibiting Inheritance of Class Using Final, Access Control and Inheritance, Multilevel Inheritance, Application of Keyword Super, Constructor Method and Inheritance, Method Overriding, Dynamic Method Dispatch, Abstract Classes, Interfaces and Inheritance.

**Interfaces:** Introduction, Declaration of Interface, Implementation of Interface, Multiple Interfaces, Nested Interfaces, Inheritance of Interfaces, Default Methods in Interfaces, Static Methods in Interface, Functional Interfaces, Annotations.

**UNIT IV: Packages and Java Library:** Introduction, Defining Package, Importing Packages and Classes into Programs, Path and Class Path, Access Control, Packages in Java SE, Java.lang Package and its Classes, Class Object, Enumeration, class Math, Wrapper Classes, Auto-boxing and Auto-unboxing, Java util Classes and Interfaces, Formatter Class, Random Class, Time Package, Class Instant (java.time.Instant), Formatting for Date/Time in Java, Temporal Adjusters Class, Temporal Adjusters Class.

**Exception Handling:** Introduction, Hierarchy of Standard Exception Classes, Keywords throws and throw, try, catch, and finally Blocks, Multiple Catch Clauses, Class Throwable, Unchecked Exceptions, Checked Exceptions.

**Java I/O and File:** Java I/O API, standard I/O streams, types, Byte streams, Character streams, Scanner class, Files in Java(Text Book 2)

**UNIT V: String Handling in Java:** Introduction, Interface Char Sequence, Class String, Methods for Extracting Characters from Strings, Comparison, Modifying, Searching; Class String Buffer.

**Multithreaded Programming:** Introduction, Need for Multiple Threads Multithreaded Programming for Multi-core Processor, Thread Class, Main Thread-Creation of New Threads, Thread States, Thread Priority-Synchronization, Deadlock and Race Situations, Inter-thread Communication - Suspending, Resuming, and Stopping of Threads.

**Java Database Connectivity:** Introduction, JDBC Architecture, Installing My SQL and My SQL Connector/J, JDBC Environment Setup, Establishing JDBC Database Connections, Result Set Interface

**Java FX GUI:** Java FX Scene Builder, Java FX App Window Structure, displaying text and image, event handling, laying out nodes in scene graph, mouse events (Text Book 3)

**Text Books:**

1. JAVA one step ahead, Anitha Seth, B.L.Juneja, Oxford.
2. Joy with JAVA, Fundamentals of Object Oriented Programming, DebasisSamanta, MonalisaSarma, Cambridge, 2023.
3. JAVA 9 for Programmers, Paul Deitel, Harvey Deitel, 4th Edition, Pearson.

**References Books:**

1. The complete Reference Java, 11<sup>th</sup>edition, Herbert Schildt,TMH
2. Introduction to Java programming, 7<sup>th</sup> Edition, Y Daniel Liang, Pearson

**Online Resources:**

1. https://nptel.ac.in/courses/106/105/106105191/
2. https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_012880464547618816347_shared/overview

**II Year I Semester**

| L | T | P | C |
|---|---|---|---|
| 0 | 0 | 3 | 1.5 |

## DATA SCIENCE LAB

**Course Objectives:**

- The main objective of the course is to inculcate the basic understanding of Data Science and its practical implementation using Python.

**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

13. Getting started with NLTK, install NLTK using PIP

14. Python program to implement with Python Sci Kit-Learn & NLTK

15. Python program to implement with Python NLTK/Spicy/Py NLPI.

**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/6.
6. https://www.nltk.org/book/ch01.html

**II Year I Semester**

| L | T | P | C |
|---|---|---|---|
| 0 | 0 | 3 | 1.5 |

### OBJECT-ORIENTED PROGRAMMING THROUGH JAVA LAB

**Course Objectives:**
The aim of this course is to
- Practice object oriented programming in the Java programming language
- implement Classes, Objects, Methods, Inheritance, Exception, Runtime Polymorphism, User defined Exception handling mechanism
- Illustrate inheritance, Exception handling mechanism, JDBC connectivity
- Construct Threads, Event Handling, implement packages, Java FX GUI

**Experiments covering the Topics:**
- Object Oriented Programming fundamentals- data types, control structures
- Classes, methods, objects, Inheritance, polymorphism,
- Exception handling, Threads, Packages, Interfaces
- Files, I/O streams, JavaFX GUI

**Sample Experiments:**
**Exercise – 1**:
a) Write a JAVA program to display default value of all primitive data type of JAVA
b) Write a java program that display the roots of a quadratic equation $ax^2+bx=0$. Calculate the discriminate D and basing on value of D, describe the nature of root.

**Exercise - 2**
a) Write a JAVA program to search for an element in a given list of elements using binary search mechanism.
b) Write a JAVA program to sort for an element in a given list of elements using bubble sort
c) Write a JAVA program using StringBuffer to delete, remove character.

**Exercise - 3**
a) Write a JAVA program to implement class mechanism. Create a class, methods and invoke them inside main method.
b) Write a JAVA program implement method overloading.
c) Write a JAVA program to implement constructor.
d) Write a JAVA program to implement constructor overloading.

**Exercise - 4**
a) Write a JAVA program to implement Single Inheritance
b) Write a JAVA program to implement multi level Inheritance
c) Write a JAVA program for abstract class to find areas of different shapes

**Exercise - 5**

a) Write a JAVA program give example for "super" keyword.

b) Write a JAVA program to implement Interface. What kind of Inheritance can be achieved?

c) Write a JAVA program that implements Runtime polymorphism

**Exercise - 6**

a) Write a JAVA program that describes exception handling mechanism

b) Write a JAVA program Illustrating Multiple catch clauses

● Write a JAVA program for creation of Java Built-in Exceptions

● Write a JAVA program for creation of User Defined Exception

**Exercise - 7**

a) Write a JAVA program that creates threads by extending Thread class. First thread display "Good Morning "every 1 sec, the second thread displays "Hello "every 2 seconds and the third display "Welcome" every 3 seconds, (Repeat the same by implementing Runnable)

b) Write a program illustrating **is Alive** and **join ()**

c) Write a Program illustrating Daemon Threads.

d) Write a JAVA program Producer Consumer Problem

**Exercise – 8**

1. Write a JAVA program that import and use the user defined packages

2. Without writing any code, build a GUI that display text in label and image in an ImageView (use JavaFX)

3. Build a Tip Calculator app using several JavaFX components and learn how to respond to user interactions with the GUI

**Exercise – 9**

1. Write a java program that connects to a database using JDBC

b) Write a java program to connect to a database using JDBC and insert values into it.

c) Write a java program to connect to a database using JDBC and delete values from it

| L | T | P | C |
|---|---|---|---|
| 0 | 1 | 2 | 2 |

**II Year I Semester**

# PYTHON PROGRAMMING

**(Skill Enhancement Course)**

**Course Objectives:**

The main objectives of the course are to

a. Introduce core programming concepts of Python programming language.
b. Demonstrate about Python data structures like Lists, Tuples, Sets and dictionaries
c. Implement Functions, Modules and Regular Expressions in Python Programming and to create practical and contemporary applications using these

**UNTI-I:** History of Python Programming Language, Thrust Areas of Python, Installing Anaconda Python Distribution, Installing and Using Jupyter Notebook.
**Parts of Python Programming Language:** Identifiers, Keywords, Statements and Expressions, Variables, Operators, Precedence and Associativity, Data Types, Indentation, Comments, Reading Input, Print Output, Type Conversions, the type () Function and Is Operator, Dynamic and Strongly Typed Language.
**Control Flow Statements:** if statement, if-else statement, if...elif…else, Nested if statement, while Loop, for Loop, continue and break Statements, Catching Exceptions Using try and except Statement.

**Sample Experiments:**
1. Write a program to find the largest element among three Numbers.
2. Write a Program to display all prime numbers within an interval
3. Write a program to swap two numbers without using a temporary variable.
4. Demonstrate the following Operators in Python with suitable examples.
  i) Arithmetic Operators ii) Relational Operators iii) Assignment Operatorsiv) Logical Operators v) Bit wise Operators vi) Ternary Operator vii) Membership Operators viii) Identity Operators
5. Write a program to add and multiply complex numbers
6. Write a program to print multiplication table of a given number.

**UNIT-II: Functions:** Built-In Functions, Commonly Used Modules, Function Definition and Calling the function, return Statement and void Function, Scope and Lifetime of Variables, Default Parameters, Keyword Arguments, *args and **kwargs, Command Line Arguments.
**Strings:** Creating and Storing Strings, Basic String Operations, Accessing Characters in String by Index Number, String Slicing and Joining, String Methods, Formatting Strings.
**Lists**: Creating Lists, Basic List Operations, Indexing and Slicing in Lists, Built-In Functions Used on Lists, List Methods, del Statement.

**Sample Experiments:**
1. Write a program to define a function with multiple return values.
2. Write a program to define a function using default arguments.
3. Write a program to find the length of the string without using any library functions.
4. Write a program to check if the substring is present in a given string or not.
5. Write a program to perform the given operations on a list:
         i. Addition      ii. Insertion      iii. slicing
6. Write a program to perform any 5 built-in functions by taking any list.

**UNIT-III: Dictionaries:** Creating Dictionary, Accessing and Modifying key:value Pairs in Dictionaries, Built-In Functions Used on Dictionaries, Dictionary Methods, del Statement.
**Tuples and Sets:** Creating Tuples, Basic Tuple Operations, tuple() Function, Indexing and Slicing in Tuples, Built-In Functions Used on Tuples, Relation between Tuples and Lists, Relation between Tuples and Dictionaries, Using zip() Function, Sets, Set Methods, Frozenset.

**Sample Experiments:**
1. Write a program to create tuples (name, age, address, college) for at least two members and concatenate the tuples and print the concatenated tuples.
2. Write a program to count the number of vowels in a string (No control flow allowed).
3. Write a program to check if a given key exists in a dictionary or not.
4. Write a program to add a new key-value pair to an existing dictionary.
5. Write a program to sum all the items in a given dictionary.

**UNIT-IV:Files**: Types of Files, Creating and Reading Text Data, File Methods to Read and Write Data, Reading and Writing Binary Files, Pickle Module, Reading and Writing CSV Files, Python os and os.path Modules.
**Object-Oriented Programming:** Classes and Objects, Creating Classes in Python, Creating Objects in Python, Constructor Method, Classes with Multiple Objects, Class Attributes Vs Data Attributes, Encapsulation, Inheritance, Polymorphism.

**Sample Experiments:**
1. Write a program to sort words in a file and put them in another file. The output file should have only lower-case words, so any upper-case words from source must be lowered.
2. Python program to print each line of a file in reverse order.
3. Python program to compute the number of characters, words and lines in a file.
4. Write a program to create, display, append, insert and reverse the order of the items in the array.
5. Write a program to add, transpose and multiply two matrices.

6.  Write a Python program to create a class that represents a shape. Include methods to calculate its area and perimeter. Implement subclasses for different shapes like circle, triangle, and square.

**UNIT-V: Introduction to Data Science:** Functional Programming, JSON and XML in Python, NumPy with Python, Pandas.

**Sample Experiments:**
1.  Python program to check whether a JSON string contains complex object or not.
2.  Python Program to demonstrate NumPy arrays creation using array () function.
3.  Python program to demonstrate use of ndim, shape, size, dtype.
4.  Python program to demonstrate basic slicing, integer and Boolean indexing.
5.  Python program to find min, max, sum, cumulative sum of array
6.  Create a dictionary with at least five keys and each key represent value as a list where this list contains at least ten values and convert this dictionary as a pandas data frame and explore the data through the data frame as follows:
    a)  Apply head () function to the pandas data frame
    b)  Perform various data selection operations on Data Frame
7.  Select any two columns from the above data frame, and observe the change in one attribute with respect to other attribute with scatter and plot operations in matplotlib

**Reference Books:**
1.  Gowrishankar S, Veena A., Introduction to Python Programming, CRC Press.
2.  Python Programming, S Sridhar, J Indumathi, V M Hariharan, 2nd Edition, Pearson, 2024
3.  Introduction to Programming Using Python, Y. Daniel Liang, Pearson.

**Online Learning Resources/Virtual Labs:**

1.  https://www.coursera.org/learn/python-for-applied-data-science-ai
2.  https://www.coursera.org/learn/python?specialization=python#syllabus

**II Year I Semester**

| L | T | P | C |
|---|---|---|---|
| 2 | 0 | 0 | 0 |

**ENVIRONMENTAL SCIENCE**

**Course Objectives:**
- To make the students to ge tawareness on environment
- To understand the importance of protecting natural resources, ecosystems for future generations and pollution causes due to the day to day activities of human life
- To save earth from the inventions by the engineers.

**Course Outcomes:**
- Grasp  multidisciplinary nature of environmental studies and various renewable and non-renewable resources.
- Understand flow and bio-geo-chemical cycles and ecological pyramids.
- Understand various causes of pollution and solid waste management and related preventive measures.
- About the rainwater harvesting, watershed management, ozone layer depletion and waste landreclamation.
- Casus of population explosion, value education and welfare programmes.

**UNIT–I**

Multidisciplinary Nature Of Environmental Studies: – Definition, Scope and Importance – Need forPublicAwareness.

Natural Resources:Renewable and non-renewable resources – Natural resources and associated problems – Forest resources – Use and over – exploitation, deforestation, case studies – Timberextraction – Mining, dams and other effects on forest and tribal people – Water resources – Use andover utilization of surface and ground water – Floods, drought, conflicts over water, dams – benefits and problems–Mineral resources:Use and exploitation, environmental effects of extracting and using mineral resources, case studies– Food resources:World food problems, changes caused by agriculture and overgrazing, effects of modern agriculture, fertilizer-pesticide problems, water logging, salinity,casestudies.–Energy resources:

**UNIT–II**

Ecosystems: Concep to fan ecosystem.–Structure and function of an ecosystem–Producers, consumers and decomposers – Energy flow in the ecosystem – Ecological succession – Food chains, food webs and ecological pyramids–Introduction, types, characteristic features, structure and function of the following ecosystem:
  a. Forest ecosystem.
  b. Grassl and ecosystem
  c. Desert ecosystem
  d. Aquatic ecosystems (ponds, streams, lakes, rivers, oceans, estuaries)
Biodiversity And Its Conservation: Introduction Definition: genetic, species and ecosystem

diversity–Bio-geographical classification of India–Value of biodiversity: consumptive use, Productive use, social, ethical, aesthetic and option values – Biodiversity at global, National and locallevels – India as a mega-diversity nation – Hot-sports of biodiversity – Threats to biodiversity: habitat loss,poaching of wildlife, man-wildlife conflicts– Endangered and endemic species of India –Conservation of biodiversity:In-situand Ex-situ conservation of biodiversity.

**UNIT–III**

Environmental Pollution:Definition, Cause, effects and control measures of:

a. Air Pollution.
b. Water pollution
c. Soil pollution
d. Marine pollution
e. Noise pollution
f. Thermal pollution
g. Nuclear hazards

Solid Waste Management: Causes, effects and control measures of urban and industrial wastes – Roleof an individual in prevention of pollution – Pollution case studies – Disaster management: floods,earthquake,cycloneandlandslides.

**UNIT–IV**

Social Issues and the Environment: From Unsustainable to Sustainable development– Urban problems related to energy – Water conservation, rain water harvesting, watershed management –Resettlement and rehabilitation of people; its problems and concerns. Case studies – Environmental ethics: Issues and possible solutions–Climate change, global warming, acidrain, ozone layer depletion, nuclear accidents and holocaust. Case Studies – Wastel and reclamation. – Consumerism and waste products. – Environment Protection Act. – Air (Prevention and Control of Pollution) Act. –Water (Prevention and control of Pollution) Act–Wild life Protection Act–Forest Conservation Act–Issues involved inenforcement of environment allegislation–Public awareness.

**UNIT–V**

Human Population And The Environment: Population growth, variation among nations. Population explosion – Family Welfare Programmes. – Environment and human health – Human Rights – Value Education–HIV/AIDS–Women and Child Welfare–Role of information Technology in Environment and human health–Case studies. Field Work:Visit to a local area to document environmental assets River/ forest grassland/ hill/ mountain – Visit to a local polluted site-Urban/Rural/Industrial/Agricultural Study of common plants,insects, and birds–river, hills lopes,etc..

**Text books:**

1. Text book of Environmental Studies for Undergraduate Courses ErachBharucha for University Grants Commission,Universities Press.

2. Palaniswamy,"Environmental Studies",Pearson education
3. S.AzeemUnnisa,"Environmental Studies"Academic Publishing Company
4. K.RaghavanNambiar,"Text book of Environmental Studies for Undergraduate Courses as per UGC model syllabus", Scitech Publications (India), Pvt.Ltd.

**Reference Books:**
1. DeekshaDaveandE.SaiBabaReddy, "Text book of Environmental Science",Cengage Publications.
2. M.AnjiReddy,"Text book of Environmental Sciences and Technology",BSPublication.
3. J.P.Sharma,Comprehensive Environmental studies,Laxmi publications.
4. J.GlynnHenryandGaryW.Heinke,"Environmental Sciences and Engineering",Prentice Hall of India Private limited
5. G.R.Chatwal,"A Text Book of Environmental Studies"Himalaya Publishing House
6. Gilbert M.Masters and WendellP.Ela,"Introduction to Environmental Engineering and Science,Prentice Hall of India Private limited.

**II Year II Semester**

| L | T | P | C |
|---|---|---|---|
| 3 | 0 | 0 | 3 |

## OPTIMIZATION TECHNIQUES

**Pre-requisite:**

**Course Objectives:**
1. To define an objective function and constraint functions in terms of design variables, and then state the optimization problem.
2. To state single variable and multi variable optimization problems, without and with constraints.
3. To explain linear programming technique to an optimization problem, define slack and surplus variables, by using Simplex method.
4. To state transportation and assignment problem as a linear programming problem to determine Simplex method.
5. To study and explain nonlinear programming techniques, unconstrained or constrained, and define exterior and interior penalty functions for optimization problems.

**Course Outcomes:** At the end of the course, student will be able to
- State and formulate the optimization problem, without and with constraints, by using design variables from an engineering design problem.
- Apply classical optimization techniques to minimize or maximize a multi-variable objective function, without or with constraints, and arrive at an optimal solution.
- Apply and Solve transportation and assignment problem by using Linear programming Simplex method.
- Apply gradient and non-gradient methods to nonlinear optimization problems and use interior or exterior penalty functions for the constraints to derive the optimal solutions
- Formulate and apply Dynamic programming technique to inventory control, production planning, engineering design problems etc. to reach a final optimal solution from the current optimal solution.

**UNIT I: Introduction and Classical Optimization Techniques:**
Statement of an Optimization problem, design vector, design constraints, constraint surface, objective function, objective function surfaces, classification of Optimization problems.
**Classical Optimization Techniques:** Single variable Optimization, multi variable Optimization without constraints, necessary and sufficient conditions for minimum/maximum, multivariable Optimization with equality constraints. Solution by method of Lagrange multipliers, multivariable Optimization with inequality constraints, Kuhn – Tucker conditions

**UNIT II: Linear Programming :**
Standard form of a linear programming problem, geometry of linear programming problems, definitions and theorems,  solution of a system of linear simultaneous equations, pivotal reduction of a general system of equations, motivation to the simplex method, simplex algorithm.

**UNIT III: Transportation Problem:**

Finding initial basic feasible solution by north – west corner rule, least cost method and Vogel's approximation method, testing for optimality of balanced transportation problems, Special cases in transportation problem.

**UNIT IV: Nonlinear Programming:**
Unconstrained cases, One – dimensional minimization methods: Classification, Fibonacci method, Univariate method, steepest descent method. Constrained cases– Characteristics of a constrained problem, Classification, Basic approach of PenaltyFunction method, Basic approaches of Interior and Exterior penalty function methods,

**UNIT V: Dynamic Programming:**
Dynamic programming multistage decision processes, types, concept of sub optimization and the principle of optimality,  computational procedure in dynamic programming, examples illustrating the calculus method of solution, examples illustrating the tabular method of solution.

**Textbooks:**
1. "Engineering optimization: Theory and practice", S. S.Rao, New Age International (P) Limited, 3rd  edition, 1998.
2.  "Introductory Operations Research", H.S. Kasene& K.D. Kumar, Springer (India), Pvt.LTd.

**Reference Books:**
1. "Optimization Methods in Operations Research and systems Analysis",  by K.V. Mital and C. Mohan, New Age International (P) Limited, Publishers, 3rd edition, 1996.
2. Operations Research, Dr.S.D.Sharma, Kedarnath, Ramnath& Co

**II Year II Semester**

| L | T | P | C |
|---|---|---|---|
| 3 | 0 | 0 | 3 |

## PROBABILITY AND STATISTICS

**Course Objectives:**
- To familiarize the students with the foundations of probability and statistical methods
- To impart probability concepts and statistical methods in various applications Engineering

**Course Outcomes:** Upon successful completion of this course, the student should be able to
1. Classify the concepts of data science and its importance (L2)
2. Interpret the association of characteristics and through correlation and regression tools (L4)
3. Apply discrete and continuous probability distributions (L3)
4. Design the components of a classical hypothesis test (L6)
5. Infer the statistical inferential methods based on small and large sampling tests (L4)

**Unit – I: Descriptive statistics and methods for data science:**
Data science – Statistics Introduction – Population vs Sample –Collection of data – primary and secondary data – Type of variable: dependent and independent Categorical and Continuous variables – Data visualization – Measures of Central tendency – Measures of Variability – Skewness – Kurtosis.

**UNIT – II: Correlation and Regression:**

Correlation – Correlation coefficient – Rank correlation.
Linear Regression: Straight line – Multiple Linear Regression - Regression coefficients and properties – Curvilinear Regression: Parabola – Exponential – Power curves.

**UNIT – III: Probability and Distributions:**

Probability– Conditional probability and Baye's theorem – Random variables – Discrete and Continuous random variables – Distribution functions – Probability mass function, Probability density function and Cumulative distribution functions – Mathematical Expectation and Variance – Binomial, Poisson, Uniform and Normal distributions.

**UNIT – IV: Sampling Theory:**

Introduction – Population and Samples – Sampling distribution of Means and Variance (definition only) – Point and Interval estimations – Maximum error of estimate – Central limit theorem (without proof) – Estimation using t, $\chi^2$ and F-distributions.

**UNIT – V: Tests of Hypothesis:**

Introduction – Hypothesis – Null and Alternative Hypothesis – Type I and Type II errors – Level of significance – One tail and two-tail tests – Test of significance for large samples and Small Samples: Single and difference means – Single and two proportions – Student's t- test, F-test, $\chi^2$ -test.

**Text Books:**

- **Miller and Freund's**, Probability and Statistics for Engineers,7/e, Pearson, 2008.

- **S. C. Gupta and V.K. Kapoor**, Fundamentals of Mathematical Statistics, 11/e, Sultan Chand & Sons Publications, 2012.

**Reference Books:**
- **Shron L. Myers, Keying Ye, Ronald E Walpole,** Probability and Statistics Engineers and the Scientists,8th Edition, Pearson 2007.
- **Jay l. Devore,** Probability and Statistics for Engineering and the Sciences, 8th Edition, Cengage.
- **Sheldon M. Ross**, Introduction to probability and statistics Engineers and the Scientists, 4th Edition, Academic Foundation, 2011.
- **Johannes Ledolter and Robert V. Hogg**, Applied statistics for Engineers and Physical Scientists, 3rd Edition, Pearson, 2010.

**II Year II Semester**

| L | T | P | C |
|---|---|---|---|
| 3 | 0 | 0 | 3 |

## DATA ENGINEERING

**Course Objectives:**

- Explain basic concepts of Data Engineering
- Discuss bout Data Engineering Life Cycle
- How to design Good Data Architecture

**UNIT-I: Introduction to Data Engineering:** Definition, Data Engineering Life Cycle, Evolution of Data Engineer, Data Engineering Versus Data Science, Data Engineering Skills and Activities,

Data Maturity, Data Maturity Model, Skills of a Data Engineer, Business Responsibilities, Technical Responsibilities, Data Engineers and Other Technical Roles.

**UNIT-II: Data Engineering Life Cycle:** Data Life Cycle Versus Data Engineering Life Cycle, Generation: Source System, Storage, Ingestion, Transformation, Serving Data.

**Major undercurrents across the Data Engineering Life Cycle:** Security, Data Management, DataOps, Data Architecture, Orchestration, Software Engineering.

**UNIT-III: Designing Good Data Architecture:** Enterprise Architecture, Data Architecture, Principles of Good Data Architecture, Major Architecture Concepts.

**Data Generation in Source Systems:** Sources of Data, Files and Unstructured Data, APIs, Application Databases (OLTP), OLAP, Change Data Capture, Logs, Database Logs, CRUD, Source System Practical Details.

**UNIT-IV: Storage:** Raw Ingredients of Data Storage, Data Storage Systems, Data Engineering Storage Abstractions, Data warehouse, Data Lake, Data Lakehouse.

**Ingestion:** Data Ingestion, Key Engineering considerations for the Ingestion Phase, Batch Ingestion Considerations, Message and Stream Ingestion Considerations, Ways to Ingest Data

**UNIT-V: Queries, Modeling and Transformation:** Queries, Life of a Query, Query Optimizer, Queries on Streaming Data, Data Modelling, Modeling Streaming Data, Transformations, Streaming Transformations and Processing.

**Serving Data for Analytics, Machine Learning and Reverse ETL**: General Considerations for serving Data, Business Analytics, Operational Analytics, Embedded Analytics, Ways to serve data for analytics and ML, Reverse ETL.

**Text Books:**

1. Joe Reis, Matt Housley, Fundamentals of Data Engineering, O'Reilly Media, Inc.,June 2022,ISBN: 9781098108304

**Reference Books:**

1. Paul Crickard , Data Engineering with Python,Packt Publishing, October 2020.
2. Ralph Kimball, Margy Ross, The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling,  Wiley, 3rd Edition, 2013
3. James Densmore, Data Pipelines Pocket Reference: Moving and Processing Data for Analytics,  O'Reilly Media, 1st Edition, 2021

**II Year II Semester**

| L | T | P | C |
|---|---|---|---|
| 3 | 0 | 0 | 3 |

## DATABASE MANAGEMENT SYSTEMS

**Course Objectives:**

The main objectives of the course is to

- Introduce database management systems and to give a good formal foundation on the relational model of data and usage of Relational Algebra
- Introduce the concepts of basic SQL as a universal Database language
- Demonstrate the principles behind systematic database design approaches by covering conceptual design, logical design through normalization
- Provide an overview of physical design of a database system, by discussing Database indexing techniques and storage techniques

**UNIT I: Introduction:** Database system, Characteristics (Database Vs File System), Database Users, Advantages of Database systems, Database applications. Brief introduction of different Data Models; Concepts of Schema, Instance and data independence; Three tier schema architecture for data independence; Database system structure, environment, Centralized and Client Server architecture for the database.

**Entity Relationship Model:** Introduction, Representation of entities, attributes, entity set, relationship, relationship set, constraints, sub classes, super class, inheritance, specialization, generalization using ER Diagrams.

**Unit II: Relational Model:** Introduction to relational model, concepts of domain, attribute, tuple, relation, importance of null values, constraints (Domain, Key constraints, integrity constraints) and their importance, Relational Algebra, Relational Calculus. BASIC SQL:Simple Database schema, data types, table definitions (create, alter), different DML operations (insert, delete, update).

**UNIT III: SQL:** Basic SQL querying (select and project) using where clause, arithmetic & logical operations, SQL functions(Date and Time, Numeric, String conversion).Creating tables with relationship, implementation of key and integrity constraints, nested queries, sub queries, grouping, aggregation, ordering, implementation of different types of joins, view(updatable and non-updatable), relational set operations.

**UNIT IV: Schema Refinement (Normalization):** Purpose of Normalization or schema refinement, concept of functional dependency, normal forms based on functional dependencyLossless join and dependency preserving decomposition, (1NF, 2NF and 3 NF), concept of surrogate key, Boyce-Codd normal form(BCNF), MVD, Fourth normal form(4NF), Fifth Normal Form (5NF).

**UNIT V: Transaction Concept:** Transaction State, ACID properties, Concurrent Executions, Serializability, Recoverability, Implementation of Isolation, Testing for

Serializability, lock based, time stamp based, optimistic, concurrency protocols, Deadlocks, Failure Classification, Storage, Recovery and Atomicity, Recovery algorithm.

**Introduction to Indexing Techniques:** B+ Trees, operations on B+Trees, Hash Based Indexing:

**Text Books:**

1) Database Management Systems, 3rd edition, Raghurama Krishnan, Johannes Gehrke, TMH (For Chapters 2, 3, 4)
2) Database System Concepts,5th edition, Silberschatz, Korth, Sudarsan,TMH (For Chapter 1 and Chapter 5)

**Reference Books:**

1. Introduction to Database Systems, 8thedition, C J Date, Pearson.
2. Database Management System, 6th edition, RamezElmasri, Shamkant B. Navathe, Pearson
3. Database Principles Fundamentals of Design Implementation and Management, Corlos Coronel, Steven Morris, Peter Robb, Cengage Learning.

**Web-Resources:**

1. https://nptel.ac.in/courses/106/105/106105175/
2. https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01275806667282022456_shared/overview

**II Year II Semester**

| L | T | P | C |
|---|---|---|---|
| 3 | 0 | 0 | 3 |

## COMPUTER ORGANIZATION AND ARCHITECTURE

**Course Objectives:**

The purpose of the course is

- Discuss about principles of computer organization and the basic architectural concepts.
- Explain in depth understanding of basic organization, design, programming of a simple digital computer, computer arithmetic, instruction set design, micro programmed control unit, pipelining and vector processing, memory organization and I/O systems.

**UNIT I: Digital Computers and Data Representation:** Introduction, Numbering Systems, Decimal to Binary Conversion, Binary Coded Decimal Numbers, Weighted Codes, Self-Complementing Codes, Cyclic Codes, Error Detecting Codes, Error Correcting Codes, Hamming Code for Error Correction, Alphanumeric Codes, ASCI Code
**Data Representation**: Data types, Complements, Fixed Point Representation, Floating Point Representation.
**Boolean Algebra and Logical gates:** Boolean Algebra :Theorems and properties, Boolean functions, canonical and standard forms , minimization of Boolean functions using algebraic identities; Karnaugh map representation and minimization using two and three variable Maps ;Logical gates ,universal gates and Two- level realizations using gates : AND-OR, OR-AND, NAND-NAND and NOR-NOR structures

**UNIT II: Digital logic circuits:** Combinatorial Circuits: Introduction, Combinatorial Circuit Design Procedure, Implementation using universal gates, Multi-bit adder, Multiplexers, De-multiplexers, Decoders
Sequential Switching Circuits: Latches and Flip-Flops, Ripple counters using T flip-flops;
**Synchronous counters**: Shift Registers; Ring counters

**UNIT III: Computer Arithmetic:** Addition and subtraction, multiplication Algorithms, Booth multiplication algorithm, Division Algorithms, Floating – point Arithmetic Operations.
**Register Transfer language and microinstructions:** Bus memory transfer, arithmetic and logical micro-operations, shift and rotate micro-operations
Basic Computer Organization and Design: Stored program concept, computer Registers, common bus system, Computer instructions, Timing and Control, Instruction cycle, Memory Reference Instructions, Input–Output configuration and program Interrupt.

**UNIT IV: Microprogrammed Control:** Control memory, Address sequencing, microprogram example, design of control unit.

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

**Program Control:** conditional Flags and Branching

**UNITV: 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.

**Text Books:**

1. Digital Logic and Computer Design, Moriss Mano, 11th Edition, Pearson.
2. Computer System Architecture,3rd Edition, M.Morris Mano, PHI

**Reference Books:**

1. Digital Logic and Computer Organization, Rajaraman, Radha krishnan, PHI, 2006
2. Computer Organization, 5Th Edition, Hamacher, Vranesic, Zaky, TMH, 2002
3. Computer Organization & Architecture: Designing for Performance, 7th Edition, William Stallings, PHI, 2006

**II Year II Semester**

| L | T | P | C |
|---|---|---|---|
| 0 | 0 | 3 | 1.5 |

## DATA ENGINEERING LAB

**Course Objective:**

- The main objective of this course is to teach how build data engineering infrastructure and data pipelines.

**Course Outcomes:**

At the end of the course student will be able to:

1. Build our Data Engineering Infrastructure
2. Demonstrate Reading and Writing files
3. Build Data Pipelines and integrate with Dashboard
4. Deploy the Data Pipeline in production

**Experiments:**

1. Installing and configuring Apache NiFi, Apache Airflow
2. Installing and configuring Elasticsearch, Kibana, PostgreSQL, pgAdmin 4
3. Reading and Writing files
    a. Reading and writing files in Python
    b. Processing files in Airflow
    c. NiFi processors for handling files
    d. Reading and writing data to databases in Python
    e. Databases in Airflow
    f. Database processors in NiFi
4. Working with Databases
    a. Inserting and extracting relational data in Python
    b. Inserting and extracting NoSQL database data in Python
    c. Building database pipelines in Airflow
    d. Building database pipelines in NiFi
5. Cleaning, Transforming and Enriching Data
    a. Performing exploratory data analysis in Python
    b. Handling common data issues using pandas
    c. Cleaning data using Airflow
6. Building the Data Pipeline
7. Building a Kibana Dash Board
8. Perform the following operations
    a. Staging and validating data
    b. Building idempotent data pipelines
    c. Building atomic data pipelines
9. Version Control with the NiFi Registry

    a.  Installing and configuring the NiFi Registry

    b.  Using the Registry in NiFi

    c.  Versioning your data pipelines

    d.  Using git-persistence with the NiFi Registry

10. Monitoring Data Pipelines

    a.  Monitoring NiFi in the GUI

    b.  Monitoring NiFi using processors

    c.  Monitoring NiFi with Python and the REST API

11. Deploying Data Pipelines

    a.  Finalizing your data pipelines for production

    b.  Using the NiFi variable registry

    c.  Deploying your data pipelines

12. Building a Production Data Pipeline

    a.  Creating a test and production environment

    b.  Building a production data pipeline

    c.  Deploying a data pipeline in production

**Reference Books:**

1. Paul Crickard , Data Engineering with Python,Packt Publishing, October 2020.

**II Year II Semester**

| L | T | P | C |
|---|---|---|---|
| 0 | 0 | 3 | 1.5 |

## DATABASE MANAGEMENT SYSTEMS LAB

**Course Objectives:**

This Course will enable students to

- Populate and query a database using SQL DDL/DML Commands
- Declare and enforce integrity constraints on a database
- Writing Queries using advanced concepts of SQL
- Programming PL/SQL including procedures, functions, cursors and triggers,

**Experiments covering the topics:**

- DDL, DML, DCL commands
- Queries, nested queries, built-in functions,
- PL/SQL programming- control structures
- Procedures, Functions, Cursors, Triggers,
- Database connectivity- ODBC/JDBC

**Sample Experiments:**

1. Creation, altering and droping of tables and inserting rows into a table (use constraints while creating tables) examples using SELECT command.
2. Queries (along with sub Queries) using ANY, ALL, IN, EXISTS, NOTEXISTS, UNION, INTERSET, Constraints. Example:- Select the roll number and name of the student who secured fourth rank in the class.
3. Queries using Aggregate functions (COUNT, SUM, AVG, MAX and MIN), GROUP BY, HAVING and Creation and dropping of Views.
4. Queries using Conversion functions (to_char, to_number and to_date), string functions (Concatenation, lpad, rpad, ltrim, rtrim, lower, upper, initcap, length, substr and instr), date functions (Sysdate, next_day, add_months, last_day, months_between, least, greatest, trunc, round, to_char, to_date)
5. 
   i. Create a simple PL/SQL program which includes declaration section, executable section and exception –Handling section (Ex. Student marks can be selected from the table and printed for those who secured first class and an exception can be raised if no records were found)
   ii. Insert data into student table and use COMMIT, ROLLBACK and SAVEPOINT in PL/SQL block.

6. Develop a program that includes the features NESTED IF, CASE and CASE expression. The program can be extended using the NULLIF and COALESCE functions.

7. Program development using WHILE LOOPS, numeric FOR LOOPS, nested loops using ERROR Handling, BUILT –IN Exceptions, USE defined Exceptions, RAISE-APPLICATION ERROR.

8. Programs development using creation of procedures, passing parameters IN and OUT of PROCEDURES.

9. Program development using creation of stored functions, invoke functions in SQL Statements and write complex functions.

10. Develop programs using features parameters in a CURSOR, FOR UPDATE CURSOR, WHERE CURRENT of clause and CURSOR variables.

11. Develop Programs using BEFORE and AFTER Triggers, Row and Statement Triggers and INSTEAD OF Triggers

12. Create a table and perform the search operation on table using indexing and non-indexing techniques.

13. Write a Java program that connects to a database using JDBC

14. Write a Java program to connect to a database using JDBC and insert values into it

15. Write a Java program to connect to a database using JDBC and delete values from it

**Text Books/Suggested Reading:**

1. Oracle: The Complete Reference by Oracle Press
2. Nilesh Shah, "Database Systems Using Oracle", PHI, 2007
3. Rick F Vander Lans, "Introduction to SQL", Fourth Edition, Pearson Education, 2007

**II Year II Semester**

| L | T | P | C |
|---|---|---|---|
| 0 | 1 | 2 | 2 |

## EXPLORATORY DATA ANALYSIS USING PYTHON
## (SKILL DEVELOPMENT COURSE)

**Course Objectives:**
1. This course introduces the fundamentals of Exploratory Data Analysis
2. It covers essential exploratory techniques for understanding multivariate data by summarizing it through statistical methods and graphical methods.

**Course Outcomes:**
1. Enumerate the fundamentals of Exploratory Data Analysis.
2. Visualize the data using basic graphs and plots.
3. Apply different Data Transformation Techniques.
4. Summarize the data using descriptive statistics.
5. Evaluate the Models and select the best model

**UNIT-I**
**Exploratory Data Analysis Fundamentals:** Understanding data science, The significance of EDA, Steps in EDA, Making sense of data, Numerical data, Categorical data, Measurement scales, Comparing EDA with classical and Bayesian analysis, Software tools available for EDA, Getting started with EDA.

**Sample Experiments:**
1. a) Download Dataset from Kaggle using the following link :
   https://www.kaggle.com/datasets/sukhmanibedi/cars4u
   b) Install python libraries required for Exploratory Data Analysis (numpy, pandas, matplotlib,seaborn)
2. Perform Numpy Array basic operations and Explore Numpy Built-in functions.
3. Loading Dataset into pandas dataframe
4. Selecting rows and columns in the dataframe

**UNIT-II**
**Visual Aids for EDA:** Technical requirements, Line chart, Bar charts, Scatter plot using seaborn, Polar chart, Histogram, Choosing the best chart
**Case Study:** EDA with Personal Email, Technical requirements, Loading the dataset, Data transformation, Data cleansing, Applying descriptive statistics, Data refactoring, Data analysis.

**Sample Experiments:**
1. Apply different visualization techniques using sample dataset
a) Line Chart  b) Bar Chart c) Scatter Plots d)Bubble Plot
2. Generate Scatter Plot using seaborn library for iris dataset
3. Apply following visualization Techniques for a sample dataset
a) Area Plot  b) Stacked Plot c) Pie chart d) Table Chart

4. Generate the following charts for a dataset.
a) Polar Chart b)Histogram c)Lollipop chart
5. Case Study: Perform Exploratory Data Analysis with Personal Email Data

**UNIT-III**

**Data Transformation:**Merging database-style dataframes, Concatenating along with an axis, Merging on index, Reshaping and pivoting, Transformation techniques, Handling missing data, Mathematical operations with NaN, Filling missing values, Discretization and binning, Outlier detection and filtering, Permutation and random sampling, Benefits of data transformation, Challenges.

**Sample Experiments:**

1. Perform the following operations
   a) Merging Dataframes
   b) Reshaping with Hierarchical Indexing
   c) Data Deduplication
   d) Replacing Values

2. Apply different Missing Data handling techniques
   a) NaN values in mathematical Operations
   b) Filling in missing data
   c) Forward and Backward filling of missing values
   d) Filling with index values
   e) Interpolation of missing values

3. Apply different data transformation techniques
   a) Renaming axis indexes
   b)Discretization and Binning
   c) Permutation and Random Sampling
   d) Dummy variables

**UNIT-IV**

**Descriptive Statistics:** Distribution function, Measures of central tendency, Measures of dispersion, Types of kurtosis, Calculating percentiles, Quartiles, Grouping Datasets, Correlation, Understanding univariate, bivariate, multivariate analysis, Time Series Analysis

**Sample Experiments:**

1. Study the following Distribution Techniques on a sample data
   a) Uniform Distribution
   b) Normal Distribution
   c) Gamma Distribution
   d) Exponential Distribution
   e) Poisson Distribution
   f) Binomial Distribution

2. Perform Data Cleaning on a sample dataset.

3. Compute measure of Central Tendency on a sample dataset
   a) Mean  b)Median  c)Mode

4. Explore Measures of Dispersion on a sample dataset

a) Variance  b) Standard Deviation  c) Skewness   d) Kurtosis

5.  a) Calculating percentiles on sample dataset

b) Calculate Inter Quartile Range(IQR) and Visualize using Box Plots

6. Perform the following analysis on automobile dataset.

a) Bivariate analysis  b)Multivariate analysis

7. Perform Time Series Analysis on Open Power systems dataset

**UNIT-V**

**Model Development and Evaluation:** Unified machine learning workflow, Data preprocessing, Data preparation, Training sets and corpus creation, Model creation and training, Model evaluation, Best model selection and evaluation, Model deployment

**Case Study:**EDA on Wine Quality Data Analysis

**Sample Experiments:**

1. Perform hypothesis testing using statsmodels library

a) Z-Test     b)T-Test

2. Develop model and Perform Model Evaluation using different metrics such as prediction score, R2 Score, MAE Score, MSE Score.

22. Case Study: Perform Exploratory Data Analysis with Wine Quality Dataset

**Text Book:**

1. Suresh Kumar Mukhiya, Usman Ahmed, Hands-On Exploratory Data Analysis with Python, Packt Publishing, 2020.

**References:**

1. Ronald K. Pearson, Exploratory Data Analysis Using R, CRC Press, 2020
2. Radhika Datar,Harish Garg, Hands-On Exploratory Data Analysis with R: Become an expert in exploratory data analysis using R packages, Ist Edition, Packt Publishing, 2019

**Web References:**

1. https://github.com/PacktPublishing/Hands-on-Exploratory-Data-Analysis-with-Python
2. https://www.analyticsvidhya.com/blog/2022/07/step-by-step-exploratory-data-analysis-eda-using-python/#h-conclusion
3. https://github.com/PacktPublishing/Exploratory-Data-Analysis-with-Python-Cookbook

**II Year II Semester**

| L | T | P | C |
|---|---|---|---|
| 1 | 0 | 2 | 2 |

## DESIGN THINKING & INNOVATION

**Course Objectives:** The objectives of the course are to

- Bring awareness on innovative design and new product development.
- Explain the basics of design thinking.
- Familiarize the role of reverse engineering in product development.
- Train how to identify the needs of society and convert into demand.
- Introduce product planning and product development process.

### UNIT – I     Introduction to Design Thinking

Introduction to elements and principles of Design, basics of design-dot, line, shape, form as fundamental design components. Principles of design. Introduction to design thinking, history of Design Thinking, New materials in Industry.

### UNIT - II     Design Thinking Process

Design thinking process (empathize, analyze, idea & prototype), implementing the process in driving inventions, design thinking in social innovations. Tools of design thinking - person, costumer, journey map, brainstorming, product development

**Activity:** Every student presents their idea in three minutes, Every student can present design process in the form of flow diagram or flow chart etc. Every student should explain about product development.

### UNIT - III     Innovation

Art of innovation, Difference between innovation and creativity, role of creativity and innovation in organizations. Creativity to Innovation. Teams for innovation, Measuring the impact and value of creativity.

**Activity:** Debate on innovation and creativity, Flow and planning from idea to innovation, Debate on value-based innovation.

### UNIT - IV     Product Design

Problem formation, introduction to product design, Product strategies, Product value, Product planning, product specifications. Innovation towards product design Case studies.

**Activity:** Importance of modeling, how to set specifications, Explaining their own product design.

### UNIT – V     Design Thinking in Business Processes

Design Thinking applied in Business & Strategic Innovation, Design Thinking principles that redefine business – Business challenges: Growth, Predictability, Change, Maintaining Relevance, Extreme competition, Standardization. Design thinking to meet corporate needs. Design thinking for Startups. Defining and testing Business Models and Business Cases. Developing & testing prototypes.

**Activity:** How to market our own product, about maintenance, Reliability and plan for startup.

**Textbooks:**

1. Tim Brown, Change by design, 1/e, Harper Bollins, 2009.
2. Idris Mootee, Design Thinking for Strategic Innovation, 1/e, Adams Media, 2014.

**Reference Books:**

1. David Lee, Design Thinking in the Classroom, Ulysses press, 2018.
2. Shrrutin N Shetty, Design the Future, 1/e, Norton Press, 2018.
3. William lidwell, Kritinaholden, &Jill butter, Universal principles of design, 2/e, Rockport Publishers, 2010.
4. Chesbrough.H, The era of open innovation, 2003.

**Online Learning Resources:**

- https://nptel.ac.in/courses/110/106/110106124/
- https://nptel.ac.in/courses/109/104/109104109/
- https://swayam.gov.in/nd1_noc19_mg60/preview
- https://onlinecourses.nptel.ac.in/noc22_de16/preview

**Course Outcomes:**

| COs | Statements | Blooms Level |
|-----|------------|--------------|
| CO1 | Define the concepts related to design thinking. | L1 |
| CO2 | Explain the fundamentals of Design Thinking and innovation. | L2 |
| CO3 | Apply the design thinking techniques for solving problems in various sectors. | L3 |
| CO4 | Analyse to work in a multidisciplinary environment. | L4 |
| CO5 | Evaluate the value of creativity. | L5 |

### B.Tech. – III Year I Semester

| S.No. | Category | Title | L | T | P | Credits |
|---|---|---|---|---|---|---|
| 1 | Professional Core | Machine Learning | 3 | 0 | 0 | 3 |
| 2 | Professional Core | Computer Networks | 3 | 0 | 0 | 3 |
| 3 | Professional Core | Software Engineering | 3 | 0 | 0 | 3 |
| 4 | Professional Elective-I | 1. Automata Theory & Compiler Design 2. Object Oriented Analysis and Design 3. Soft computing 4. Internet of Things 5. Any of the 12-Week SWAYAM /NPTEL Course suggested by the BoS | 3 | 0 | 0 | 3 |
| 5 | Open Elective- I | OR Entrepreneurship Development & Venture Creation | 3 | 0 | 0 | 3 |
| 6 | Professional Core | Machine Learning Lab | 0 | 0 | 3 | 1.5 |
| 7 | Professional Core | Computer Networks Lab | 0 | 0 | 3 | 1.5 |
| 8 | Skill Enhancement course | Full Stack development -1 / SWAYAM Plus – Data Engineer / AI Engineer | 0 | 1 | 2 | 2 |
| 9 | ES | Tinkering Lab *(User Interface Design using Flutter)*/ SWAYAM Plus - Android Application Development (with Flutter) | 0 | 0 | 2 | 1 |
| 10 | Evaluation of Community Service Project Internship | | - | - | - | 2 |
| | **Total** | | **15** | **1** | **10** | **23** |
| MC | Student may select from the Same Minor Pool | | 3 | 0 | 3 | 4.5 |
| MC | Minor Course through SWAYAM / NPTEL (Minimum 12 Week, 3 credit course) | | 3 | 0 | 0 | 3 |
| HC | Student may select from the Same Honors Pool | | 3 | 0 | 0 | 3 |
| HC | Student may select from the Same Honors Pool | | 3 | 0 | 0 | 3 |

**B. Tech– III Year II Semester**

| S.No | Category | Title | L | T | P | C |
|------|----------|-------|---|---|---|---|
| 1 | Professional Core | Deep Learning | 3 | 0 | 0 | 3 |
| 2 | Professional Core | Operating Systems | 3 | 0 | 0 | 3 |
| 3 | Professional Core | Data Visualization | 3 | 0 | 0 | 3 |
| 4 | Professional Elective-II | 1. Social Media Analytics<br>2. Cryptography & Network Security<br>3. Recommender Systems<br>4. Cloud Computing<br>5.  Sensor Networks | 3 | 0 | 0 | 3 |
| 5 | Professional Elective-III | 1. Software Project Management<br>2. Quantum Computing<br>3. Computer Vision<br>4. NoSQL databases<br>5. Any of the 12-Week SWAYAM /NPTEL Course suggested by the BoS | 3 | 0 | 0 | 3 |
| 6 | Open Elective – II | | 3 | 0 | 0 | 3 |
| 7 | Professional Core | Deep Learning Lab | 0 | 0 | 3 | 1.5 |
| 8 | Professional Core |  Data Visualization Lab | 0 | 0 | 3 | 1.5 |
| 9 | Skill Enhancement course | Soft skills | 0 | 1 | 2 | 2 |
| 10 | Audit Course | Technical Paper Writing & IPR | 2 | 0 | 0 | - |
| | | **Total** | **20** | **01** | **08** | **23** |
| | *Mandatory Industry Internship / Mini Project of 08 weeks duration during summer vacation | | | | | |
| MC | Student may select from the same minors pool | | 3 | 0 | 3 | 4.5 |
| MC | Minor Course<br>(Student may select from the same specialized minors pool) | | 3 | 0 | 0 | 3 |
| | Student may select from the same honors pool | | 3 | 0 | 0 | 3 |
| HC | Honors Course<br>( Student may select from the honors pool) | | 3 | 0 | 0 | 3 |

*** Under Industry Internship interested students can pursue SWAYAM Plus courses viz.,
*Hands-on Masterclass on Data Analytics OR Artificial Intelligence for Real-World
Application**

*Note : Student need to do at least ONE MOOC Course (3 credits out of 160 credits) to meet the mandatory requirement (11th criteria, as per R23 Regulations)*

**Open Electives, offered to other department students:**
Open Elective I: Operating Systems / Computer Organization and Architecture
Open Elective II: Database Management Systems
Open Elective III: Object Oriented Programming Through Java
Open Elective IV: Computer Networks / Software Engineering / IOT Based Smart Systems

## Minor Engineering

*Note:*
1. *To obtain Minor Engineering, student needs to obtain 18 credits by successfully completing any of the following courses in the concern stream.*
2. *During Minor/Honors Course selection, there should not be any overlapping with Regular/Major/OPEN Electives*

**Minor in Data Science**
1. Introduction to Data Science                        3-0-3-4.5 (II-II)
2. Operating Systems                                   3-0-0-3 (III-I)
3. Data Engineering                                    3-0-3-4.5 (III-II)
4. Deep Learning                                       3-0-0-3 (IV-I)
   **Any of the following 12 Week 3 credit NPTEL MOOC Courses**
5. Introduction to Database Systems
6. Artificial Intelligence: Knowledge Representation and Reasoning
7. Computer Networks and Internet Protocol
8. Fundamentals of Object Oriented Programming
9. Discrete Mathematics for CS
10. Software Engineering

**Suggested MOOC Courses for Honors Degree in Data Science**

**Note:** *To obtain Honor's degree, student needs to obtain 18 credits by successfully completing any of the following courses in the concern stream (without duplication).*

**Mandatory Course(s):**
1. Deep Learning for Natural Language Processing - 12 Week 3 Credit Course, MOOCS
2. Applied Time-Series Analysis                 12 Week 3 Credit Course, MOOCS

**Any of the following for remaining 12 Credits**
3. Social Network Analysis               12 Week 3 Credit Course, MOOCS
4. Privacy and Security in Online Social Media 12 Week 3 Credit Course, MOOCS
5. Reinforcement Learning                12 Week 3 Credit Course, MOOCS
6. Algorithms in Computational Biology and Sequence Analysis     12 Week 3 Credit Course, MOOCS
7. GPU Architecture and Programming      12 Week 3 Credit Course, MOOCS
8. Quantum Algorithms and Cryptography    12 Week 3 Credit Course, MOOCS
9. Affective Computing                  12 Week 3 Credit Course, MOOCS
10. Unmanned Arial Systems & Robotics     12 Week 3 Credit Course, MOOCS

**B. Tech III Year – I Semester**

| III Year I Semester | MACHINE LEARNING | L | T | P | C |
|---|---|---|---|---|---|
| | | 3 | 0 | 0 | 3 |

**Course Objectives:**

The objectives of the course are to

- Define machine learning and its different types (supervised and unsupervised) and understand their applications.
- Apply supervised learning algorithms including decision trees and k-nearest neighbors (k-NN).
- Implement unsupervised learning techniques, such as K-means clustering.

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

- Enumerate the Fundamentals of Machine Learning
- Build Nearest neighbor-based models
- Apply Models based on decision trees and Bayes rule
- Make use of Linear discriminants for machine Learning
- Choose appropriate clustering technique

**UNIT - I: Introduction to Machine Learning:**

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

**UNIT - II: Nearest Neighbor-Based Models:**

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

**UNIT - III: Models Based on Decision Trees**:

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

**UNIT - IV: Linear Discriminants for Machine Learning**:

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

**UNIT - V: Clustering:** Introduction to Clustering, Partitioning of Data, Matrix Factorization | Clustering of Patterns, Divisive Clustering, Agglomerative Clustering, Partitional Clustering, K-Means Clustering, Soft Partitioning, Soft Clustering, Fuzzy C-Means Clustering, Rough Clustering, Rough K-Means Clustering Algorithm, Expectation Maximization-Based Clustering, Spectral Clustering.

**Text Books:**

1."Machine Learning Theory and Practice", M N Murthy, V S Ananthanarayana, Universities Press (India), 2024

**Reference Books:**

1."Machine Learning", Tom M. Mitchell, McGraw-Hill Publication, 2017
2."Machine Learning in Action", Peter Harrington, Dream Tech
3."Introduction to Data Mining", Pang-Ning Tan, Michel Stenbach, Vipin Kumar, 7th Edition, 2019.

| III Year I Semester | COMPUTER NETWORKS | L | T | P | C |
|---|---|---|---|---|---|
| | | 3 | 0 | 0 | 3 |

**Course Objectives:**

The course is designed to

- To understand the different types of networks

- To develop an understanding the principles of computer networks.

- To familiarize with Reference model OSI and TCP/IP

- To understand various layers of Reference models functions

- To explore network protocols

**Course Outcomes (CO):**

After completion of the course, students will be able to

- Understand the reference models and network protocols
- Describe data transmission media and data link layer.
- Understand the network layer design issues and Network layer Protocols.
- Evaluate transport layer services and its protocols.
- Understand application layer protocols and their uses

**UNIT - I: Introduction**
Types of Computer Networks, Reference Models- The OSI Reference Model, The TCP/IP Reference Model, A Critique of the OSI Model and Protocols, A Critique of the TCP/IP Reference Model. History of Internet.

**UNIT -II: The Data Link Layer**
Transmission Media, Guided and Un-guided media,
Data Link Layer Design Issues, Services Provided to the Network Layer, Error detecting and Error Correcting codes, Elementary Data Link Protocols, Sliding Window Protocols, HDLC, PPP. Multiple Access Protocols Wired Lans: Ethernet, Fast Ethernet, Gigabit Ethernet

**UNIT - III: The Network Layer**
Network Layer Design Issues, Routing Algorithms, Congestion, Congestion control algorithms. The Network Layer in the Internet, The IP Version 4 Protocol, IP Addresses-Classful, CIDR, NAT, IP Version 6 Protocol, Transition from IPV4 to IPV6

**UNIT -IV: TheTransportLayer**
The Transport Layer Services, Transport Layer Protocols: UDP, TCP and SCTP

**UNIT -V: The Application Layer**

The World Wide Web, HTTP, Domain Name Space, Remote Loging, Electronic Mail and File Transfer

**Textbooks:**

1. "Computer Networks", Andrew S Tanenbaum, David J Wetherall, 5<sup>th</sup> Edition, Pearson
2. "Data Communications and Networking", Behrouz A Forouzan, 4<sup>th</sup> Edition, Tata McGraw Hill Education

**Reference Books**:

1. "Data and Computer Communication", William Stallings, Pearson
2. "TCP/IP Protocol Suite", Behrouz Forouzan, McGraw Hill.

| III Year I Semester | **SOFTWARE ENGINEERING** | **L** | **T** | **P** | **C** |
|---|---|---|---|---|---|
| | | **3** | **0** | **0** | **3** |

**Course Objectives:**

The objectives of this course are to introduce

- Software lifecycle models, Software requirements and SRS document.
- Project Planning, quality control and ensuring good quality software.
- Software Testing strategies, use of CASE tools, Implementation issues, validation &verification procedures.

**UNIT - I:**

**Introduction:** Evolution, Software development projects, Exploratory style of software developments, Emergence of software engineering, Notable changes in software development practices, Computer system engineering.

**Software Life Cycle Models:** Basic concepts, Waterfall model and its extensions, Rapid application development, Agile development model, Spiral model.

**UNIT - II:**

**Software Project Management:** Software project management complexities, Responsibilities of a software project manager, Metrics for project size estimation, Project estimation techniques, Empirical Estimation techniques, COCOMO, Halstead's software science, risk management.

**Requirements Analysis and Specification:** Requirements gathering and analysis, Software Requirements Specification (SRS), Formal system specification, Axiomatic specification, Algebraic specification, Executable specification and 4GL.

**UNIT - III:**

**Software Design:** Overview of the design process, How to characterize a good software design? Layered arrangement of modules, Cohesion and Coupling. approaches to software design.

**Agility:** Agility and the Cost of Change, Agile Process, Extreme Programming (XP), Other Agile Process Models, Tool Set for the Agile Process (Text Book 2)

**Function-Oriented Software Design:** Overview of SA/SD methodology, Structured analysis, Developing the DFD model of a system, Structured design, Detailed design, and Design Review.

**User Interface Design:** Characteristics of a good user interface, Basic concepts, Types of user interfaces, Fundamentals of component-based GUI development, and user interface design methodology.

**UNIT - IV:**

**Coding And Testing:** Coding, Code review, Software documentation, Testing, Black-box

testing, White-Box testing, Debugging, Program analysis tools, Integration testing, testing object-oriented programs, Smoke testing, and some general issues associated with testing.

**Software Reliability and Quality Management:** Software reliability. Statistical testing, Software quality, Software quality management system, ISO9000. SEI Capability maturity model. Fewother important quality standards, and Six Sigma.

**UNIT - V:**
**Computer-Aided Software Engineering (Case):** CASE and its scope, CASE environment, CASE support in the software life cycle, other characteristics of CASE tools, Towards second generation CASE Tool, and Architecture of a CASE Environment.

**Software Maintenance:** Characteristics of software maintenance, Software reverse engineering, Software maintenance process models and Estimation of maintenance cost.

**Software Reuse:** Reuse-definition, Introduction, Reason behind no reuses of ar, Basic issues in any reuse program, A reuse approach, and Reuse at organization level.

**Text Books:**

1. Fundamentals of Software Engineering, RajibMall,5ᵗʰEdition, PHI.
2. Software Engineering a Practitioner's Approach, Roger S. Pressman, 9ᵗʰEdition, McGraw Hill International Edition.

**Reference Books:**

1. Software Engineering, IanSommerville,10ᵗʰEdition, Pearson.
2. Software Engineering, Principles and Practices, Deepak Jain, Oxford University Press.

**Resources:**

1. https://nptel.ac.in/courses/106/105/106105182/
2. https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01260589506387148827_shared/overview
3. https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01338269041100390473 5_shared/overview

| III Year I Semester | AUTOMATA THEORY AND COMPILER DESIGN (PROFESSIONAL ELECTIVE -I) | L | T | P | C |
|---|---|---|---|---|---|
| | | 3 | 0 | 0 | 3 |

**Course Outcomes:** After completion of this course

- Understand and apply formal language theory.
- Design and implement parsers.
- Understand the phases of a compiler.
- Apply semantic analysis and error handling.
- Optimize intermediate and target code.

**UNIT – I:**
**Introduction to Finite Automata:** Structural Representations, Automata and Complexity, the Central Concepts of Automata Theory – Alphabets, Strings, Languages, Problems. Nondeterministic Finite Automata: Formal Definition, an application, Text Search, Finite Automata with Epsilon-Transitions. Deterministic Finite Automata: Definition of DFA, How A DFA Process Strings, The language of DFA, Conversion of NFA with €-transitions to NFA without €-transitions. Conversion of NFA to DFA.

**UNIT – II:**
**Regular Expressions:** Finite Automata and Regular Expressions, Applications of Regular Expressions, Algebraic Laws for Regular Expressions, Conversion of Finite Automata to Regular Expressions. Pumping Lemma for Regular Languages: Statement of the pumping lemma, Applications of the Pumping Lemma. Context-Free Grammars: Definition of Context-Free Grammars, Derivations Using a Grammar, Leftmost and Rightmost Derivations, the Language of a Grammar, Parse Trees, Ambiguity in Grammars and Languages.

**UNIT – III:**
**Push Down Automata:** Definition of the Pushdown Automaton, the Languages of a PDA, Equivalence of PDA's and CFG's, Acceptance by final state Turing Machines: Introduction to Turing Machine, Formal Description, Instantaneous description, The language of a Turing machine Undecidability: Undecidability, A Language that is Not Recursively Enumerable, An Undecidable Problem That is RE, Undecidable Problems about Turing Machines

**UNIT – IV:**
**Introduction:** The structure of a compiler, Lexical Analysis: The Role of the Lexical Analyzer, Input Buffering, Recognition of Tokens, The Lexical- Analyzer Generator Lex.
**Syntax Analysis:** Introduction, Context-Free Grammars, Writing a Grammar, Top-Down Parsing, Bottom- Up Parsing.
**Introduction to LR Parsing:** Simple LR, More Powerful LR Parsers R18 B.Tech. CS&D Syllabus JNTU Hyderabad.

**UNIT – V:**

**Syntax-Directed Translation:** Syntax-Directed Definitions, Evaluation Orders for SDD's, Syntax Directed Translation Schemes, Implementing L-Attributed SDD's. Intermediate-Code Generation: Variants of Syntax Trees, Three-Address Code Run-Time Environments: Stack Allocation of Space, Access to Nonlocal Data on the Stack, Heap Management

**Text Books:**

1. Introduction to Automata Theory, Languages, and Computation, 3nd Edition, John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman, Pearson Education.
2. Compilers: Principles, Techniques and Tools, Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffry D. Ullman, 2nd Edition, Pearson.
3. Theory of Computer Science – Automata languages and computation, Mishra and Chandrashekaran, 2nd Edition, PHI.

**Reference Books:**

1. Introduction to Formal languages Automata Theory and Computation, Kamala Krithivasan, Rama R, Pearson.
2. Introduction to Languages and The Theory of Computation, John C Martin, TMH.
3. Lex &Yacc – John R. Levine, Tony Mason, Doug Brown, O'reilly
4. Compiler Construction, Kenneth C. Louden, Thomson. Course Technology.

| III Year I Semester | **OBJECT ORIENTED ANALYSIS AND DESIGN (PROFESSIONAL ELECTIVE -I)** | L | T | P | C |
|---|---|---|---|---|---|
| | | 3 | 0 | 0 | 3 |

**Course Objectives:**

The main objective is the students to

- Become familiar with all phases of OOAD.
- Master the main features of the UML.
- Master the main concepts of Object Technologies and how to apply them at work and develop the ability to analyze and solve challenging problem in various domains.
- Learn the Object design Principles and understand how to apply them towards Implementation.

**UNIT - I:**

**Introduction:** The Structure of Complex systems, The Inherent Complexity of Software, Attributes of Complex System, Organized and Disorganized Complexity, Bringing Order to Chaos, Designing Complex Systems.
**Case Study:** System Architecture: Satellite-Based Navigation

**UNIT - II:**

**Introduction to UML:** Importance of modeling, principles of modeling, object-oriented modeling, conceptual model of the UML, Architecture, and Software Development Life Cycle.
**Basic Structural Modeling:** Classes, Relationships, common Mechanisms, and diagrams.
**Case Study:** Control System: Traffic Management.

**UNIT - III:**

**Class & Object Diagrams:** Terms, concepts, modeling techniques for Class & Object Diagrams.
**Advanced Structural Modeling:** Advanced classes, advanced relationships, Interfaces, Types and Roles, Packages. **Case Study:** AI: Cryptanalysis.

**UNIT - IV:**

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

**UNIT-V:**

**Advanced Behavioral Modeling:** Events and signals, state machines, processes and Threads, time and space, state chart diagrams.

**Architectural Modeling:** Component, Deployment, Component diagrams and Deployment diagrams

**Case Study:** Weather Forecasting

**Text Books:**

1. Grady BOOCH, Robert A. Maksimchuk, Michael W. ENGLE, Bobbi J. Young, Jim Conallen, Kellia Houston, "Object- Oriented Analysis and Design with Applications", 3rd edition,2013, PEARSON.

2. Grady Booch, James Rumbaugh, Ivar Jacobson: The Unified Modeling Language User Guide, Pearson Education.

**Reference Books:**

1. Meilir Page-Jones: Fundamentals of Object-Oriented DesigninUML,PearsonEducation.

2. PascalRoques:ModelingSoftwareSystemsUsingUML2,WILEY- DreamtechIndiaPvt.Ltd.

3. Atul Kahate: Object Oriented Analysis &Design, The McGraw-Hill Companies.

Appling UML and Patterns: An introduction to Object–Oriented Analysis and Design and Unified Process, Craig Larman, Pearson Education.

| III Year I Semester | SOFT COMPUTING (PROFESSIONAL ELECTIVE -I) | L | T | P | C |
|---|---|---|---|---|---|
| | | 3 | 0 | 0 | 3 |

**Course Objectives:**

To introduce the concepts in Soft Computing such as Artificial Neural Networks, Fuzzy logic-based systems, genetic algorithm-based systems and their hybrids.

**Course Outcomes:** The students will be able to

- Learn soft computing techniques and their applications.
- Analyze various neural network architectures.
- Define the fuzzy systems.
- Understand the genetic algorithm concepts and their applications.
- Identify and select a suitable Soft Computing technology to solve the problem; construct a solution and implement a Soft Computing solution

**UNIT - I:**
Introduction to Soft Computing, Artificial neural networks, biological neurons, Basic models of artificial neural networks, Connections, Learning, Activation Functions, McCulloch and Pitts Neuron, Hebb network.

**UNIT - II:**
Perceptron networks, learning rule, Training and testing algorithm, Adaptive Linear Neuron, Back propagation Network, Architecture, Training algorithm

**UNIT - III:**
Fuzzy logic, fuzzy sets, properties, operations on fuzzy sets, fuzzy relations, operations on fuzzy relations, Fuzzy membership functions, fuzzification, Methods of membership, value assignments, intuition, inference, rank ordering, Lambda –cuts for fuzzysets, Defuzzification methods.

**UNIT - IV:**
Truth values and Tables in Fuzzy Logic, Fuzzy propositions, Formation of fuzzy rules, Decomposition of rules, Aggregation of rules, Fuzzy Inference Systems, Mamdani and Sugeno types, Neuro-fuzzy hybrid systems, characteristics, classification.

**UNIT - V:**
Introduction to genetic algorithm, operators in genetic algorithm, coding, selection, crossover, mutation, stopping condition for genetic algorithm flow, Genetic-neuro hybrid systems, Genetic Fuzzy rule-based system

**Text Books:**

1. S. N. Sivanandamand S. N. Deepa, Principles of soft computing–John Wiley & Sons,2007.
2. Timothy J. Ross, Fuzzy Logic with engineering applications, John Wiley& Sons, 2016.

**Reference Books:**

1. N. K. Sinhaand M. M. Gupta, Soft Computing & Intelligent Systems: Theory & Applications-Academic Press /Elsevier. 2009.
2. Simon Haykin, Neural Network- A Comprehensive Foundation-Prentice Hall International, Inc.1998
3. R. Eberhart and Y. Shi, ComputationalIntelligence: Concepts to Implementation, Morgan Kaufman/Elsevier, 2007.
4. Driankov D., Hellendoorn H.and Reinfrank M., An Introduction to Fuzzy Control Narosa Pub., 2001.
5. BartKosko, Neural NetworkandFuzzySystems-PrenticeHall,Inc.,Englewood Cliffs, 1992
6. Goldberg D.E.,Genetic Algorithms in Search, Optimization, and Machine Learning Addison Wesley, 1989

| III Year I Semester | INTERNET OF THINGS (PROFESSIONAL ELECTIVE -I) | L | T | P | C |
|---|---|---|---|---|---|
| | | 3 | 0 | 0 | 3 |

**Course Objectives:**

- Vision and Introduction to Internet of Things (IoT).
- Understand IoT Market perspective.
- Data and Knowledge Management and use of Devices in IoT Technology.
- Understand State of the Art – IoT Architecture.
- Understand Real World IoT Design Constraints, Industrial Automation and Commercial.

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

- Explain in a concise manner how the general Internet as well as Internet of Things work.
- Understand constraints and opportunities of wireless and mobile networks for Internet of Things.
- Use basic sensing and measurement and tools to determine the real-time performance of network of devices.
- Develop prototype models for various applications using IoT technology.

**UNIT-I:**
**The Internet of Things:** An Overview of Internet of things, Internet of Things Technology, behind IoTs Sources of the IoTs, M2M Communication, Examples of IoTs, Design Principles for Connected Devices Internet Connectivity Principles, Internet connectivity, Application Layer Protocols: HTTP, HTTPS, FTP, Telnet.

**UNIT-II:**
Business Models for Business Processes in the Internet of Things, IoT/M2M systems LAYERS AND designs standardizations, Modified OSI Stack for the IoT/M2M Systems, ETSI M2M domains and High- level capabilities, Communication Technologies, Data Enrichment and Consolidation and Device Management Gateway Ease of designing and affordability

**UNIT-III:**
Design Principles for the Web Connectivity for connected-Devices, Web Communication protocols for Connected Devices, Message Communication protocols for Connected Devices, Web Connectivity for connected-Devices.

**UNIT-IV:**
Data Acquiring, Organizing and Analytics in IoT/M2M, Applications /Services /Business Processes, IOT/M2M Data Acquiring and Storage, Business Models for Business Processes in the Internet of Things, Organizing Data, Transactions, Business Processes, Integration and Enterprise Systems.

**UNIT-V:**
Data Collection, Storage and Computing Using a Cloud Platform for IoT/M2M Applications/Services, Data Collection, Storage and Computing Using cloud platform Everything as a service and Cloud Service Models, IOT cloud-based services using the Xively (Pachube/COSM), Nimbits and other platforms Sensor, Participatory Sensing, Actuator, Radio Frequency Identification, and Wireless, Sensor Network Technology, Sensors Technology, Sensing the World.

**Text Books:**
1. Internet of Things: Architecture, Design Principles and Applications, Rajkamal, McGraw Hill Higher Education
2. Internet of Things, A. Bahgya and V. Madisetti, University Press, 2015

**Reference Books:**
1. Designing the Internet of Things, Adrian McEwen and Hakim Cassimally, Wiley
Getting Started with the Internet of Things, Cuno Pfister, Oreilly

| III Year I Semester | **MACHINE LEARNING LAB** | L | T | P | C |
|---|---|---|---|---|---|
| | | 0 | 0 | 3 | 1.5 |

**Course Objectives:**

- To learn about computing central tendency measures and Data preprocessing techniques
- To learn about classification and regression algorithms
- To apply different clustering algorithms for a problem.

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

- Develop program for computing central tendency measures and Apply Data Preprocessing techniques
- Build Classifiers using KNN, Decision Tree, Random Forest algorithms
- Implement classification algorithms such as Naïve Bayes, SVM, Multi-Layer Perceptron
- Apply clustering algorithms such as K-Means, Fuzzy C-Means and Expectation Maximization for a problem

**Software's Required: Python/R/Weka**

**List of Experiments:**

1. Compute Central Tendency Measures: Mean, Median, Mode Measure of Dispersion: Variance, Standard Deviation.
2. Apply the following Pre-processing techniques for a given dataset.
    a. Attribute selection
    b. Handling Missing Values
    c. Discretization
    d. Elimination of Outliers
3. Apply KNN algorithm for classification and regression
4. Demonstrate decision tree algorithm for a classification problem and perform parameter tuning for better results
5. Demonstrate decision tree algorithm for a regression problem
6. Apply Random Forest algorithm for classification and regression
7. Demonstrate Naïve Bayes Classification algorithm.
8. Apply Support Vector algorithm for classification
9. Demonstrate simple linear regression algorithm for a regression problem
10. Apply Logistic regression algorithm for a classification problem
11. Demonstrate Multi-layer Perceptron algorithm for a classification problem
12. Implement the K-means algorithm and apply it to the data you selected. Evaluate performance by measuring the sum of the Euclidean distance of each example from its class center. Test the performance of the algorithm as a function of the parameters K.

13.    Demonstrate the use of Fuzzy C-Means Clustering

14. Demonstrate the use of Expectation Maximization based clustering algorithm

| III Year I Semester | COMPUTER NETWORKS LAB | L | T | P | C |
|---|---|---|---|---|---|
| | | 0 | 0 | 3 | 1.5 |

**List of Experiments:**

1. Study of Network devices in detail and connect the computers in Local Area Network.

2. Write a Program to implement the data link layer farming methods such as

    i) Character stuffing ii) bit stuffing.

3. Write a Program to implement data link layer farming method checksum.

4. Write a program for Hamming Code generation for error detection and correction.

5. Write a Program to implement on a data set of characters the three CRC polynomials – CRC 12, CRC 16 and CRC CCIP.

6. Write a Program to implement Sliding window protocol for Goback N.

7. Write a Program to implement Sliding window protocol for Selective repeat.

8. Write a Program to implement Stop and Wait Protocol.

9. Write a program for congestion control using leaky bucket algorithm

10. Write a Program to implement Dijkstra's algorithm to compute the Shortest path through a graph.

11. Write a Program to implement Distance vector routing algorithm by obtaining routing table at each node (Take an example subnet graph with weights indicating delay between nodes).

12. Write a Program to implement Broadcast tree by taking subnet of hosts.

13. Wireshark

    i. Packet Capture Using Wire shark

    ii. Starting Wire shark

    iii. Viewing Captured Traffic

    iv. Analysis and Statistics & Filters.

14. How to run Nmap scan

15. Operating System Detection using Nmap

16. Do the following using NS2 Simulator

    i. NS2 Simulator-Introduction

    ii. Simulate to Find the Number of Packets Dropped

    iii. Simulate to Find the Number of Packets Dropped by TCP/UDP

iv. Simulate to Find the Number of Packets Dropped due to Congestion

v. Simulate to Compare Data Rate& Throughput.

| **III Year I Semester** | **FULL STACK DEVELOPMENT – I** **(SKILL ENHANCEMENT COURSE)** | **L** | **T** | **P** | **C** |
|---|---|---|---|---|---|
| | | **0** | **1** | **2** | **2** |

**Course Objectives:**

The main objectives of the course are to

1. Make use of HTML elements and their attributes for designing static webpages.

2. Build a webpage by applying appropriate CSS styles to HTML elements.

3. Experiment with Java Script to develop dynamic webpages and validate forms.

**Experiments covering the Topics:**
- Lists, Links and Images
- HTML Tables, Forms and Frames
- HTML5 and Cascading Style Sheets, Types of CSS
- Selector forms
- CSS with Color, Background, Font, Text and CSS Box Model
- Applying Java Script-internal and external, I/O, Type Conversion
- JavaScript Conditional Statements and Loops, Pre-defined and User-defined Objects
- JavaScript Functions and Events
- Node.js

*Sample Experiments:*

**1. Lists, Links and Images**

a. Write a HTML program, to explain the working of lists.

Note: It should have an ordered list, unordered list, nested list and ordered list in an unordered list and definition lists.

b. Write a HTML program, to explain the working of hyperlinks using<a>tag and href, target Attributes.

c. Create a HTML document that has your image and your friend's image with a specific height and width. Also, when clicked on the images it should navigate to their respective profiles.

d. Write a HTML program, in such a way that, rather than placing large images on a page, the preferred technique istousethumbnailsbysettingtheheightandwidthparameterstosomething like to 100*100 pixels. Each thumbnail image is also a link to a full-sized version of the image. Create an image gallery using this technique

**2. HTML Tables, Forms and Frames**
- Write a HTML program, to explain the working of tables. (use tags: <table>, <tr>, <th>, <td> and attributes: border, rowspan, colspan)
- Write a HTML program, to explain the working of tables by preparing a timetable.
- (Note: Use <caption>tag to set the caption to the table & also use cell spacing, cell

padding, border, rowspan, colspan etc.).

- Write a HTML program, to explain the working of forms by designing Registration form. (Note: Include text field, password field, number field, date of birth field, checkboxes, radio buttons, list boxes using <select>&<option> tags,<text area>and two buttons i.e. submit and reset. Use tables to provide a better view).

- Write a HTML program, to explain the working of frames, such that page is to be divided into3parts on either direction. (Note: first frame image, second frame paragraph, third frame hyperlink. And also make sure of using "no frame" attribute such that frames to be fixed).

### 3. HTML5 and Cascading Style Sheets, Types of CSS
a. Write a HTML program, that makes use of<article>,<aside>,<figure>,<figcaption>,<footer>, <header>,<main>,<nav>,<section>,<div>,<span>tags.
b. Write a HTML program, to embed audio and video into HTML web page.
c. Write a program to apply different types (of levels of styles or style specification formats) - inline, internal, external styles to HTML elements. (identify selector, property and value).

### 4. Selector forms
a. Write a program to apply different types of selector forms
  - Simple selector (element, id, class, group, universal)
  - Combinator selector (descendant, child, adjacent sibling, general sibling)
  - Pseudo-class selector
  - Pseudo-element selector
  - Attribute selector

### 5. CSS with Color, Background, Font, Text and CSS Box Model
a. Write a program to demonstrate the various ways you can reference a color in CSS.
b. Write a CSS rule that places a background image half way down the page, tilting it horizontally. The image should remain in place when the user scrolls up or down.
c. Write a program using the following terms related to CSS font and text:
    i.font-sizeii.font-weightiii.font-styleiv.text-decorationv.text-transformation vi.text-alignment
d. Write a program, to explain the importance of CSS Box model using
    i. Content ii. Border iii. Margin iv. Padding

### 6. Applying JavaScript-internal and external, I/O, Type Conversion
a. Write a program to embed internal and external Java Script in a web page.
b. Write a program to explain the different ways for displaying output.
c. Write a program to explain the different ways for taking input.

d. Create a web page which uses prompt dialogue box to ask a voter for his name and age. Displaythe information in table format along with either the voter can vote or not

## 7. Java Script Pre-defined and User-defined Objects
a. Write a program using document object properties and methods
b. Write a program using window object properties and methods.
c. Write a program using array object properties and methods.
d. Write a program using math object properties and methods.
e. Write a program using string object properties and methods.
f. Write a program using regex object properties and methods.
g. Write a program using date object properties and methods.
h. Write a program to explain user-defined object by using properties, methods, accessors, constructors and display.

## 8. Java Script Conditional Statements and Loops
a. Write a program which asks the user to enter three integers, obtains the numbers from the user and outputs HTML text that displays the larger number followed by the words "LARGERNUMBER" in an information message dialog. If the numbers are equal, output HTML text as "EQUAL NUMBERS".
b. Write a program to display weekdays using switch case.
c. Write a program to print1to10 numbers using for, while and do-while loops.
d. Write a program to print data in object using for-in, for-each and for-of loops
e. Develop a program to determine whether a given number is an 'ARMSTRONGNUMBER' or not. [Ex: 153 is an Armstrong number, since sum of the cube of the digits is equal to the number i.e.,13 + 53+33 = 153]
f. Write a program to display the denomination of the amount deposited in the bank in terms of 100's, 50's,20's,10's,5's,2's&1's.(Ex: If deposited amount is Rs.163, the output shouldbe1-100's,1- 50's, 1- 10's, 1-2's & 1-1's)

## 9. Java Script Functions and Events
a. Design a appropriate function should be called to display
   - Factorial of that number
   - Fibonacci series up to that number
   - Prime numbers up to that number
   - Is it palindrome or not
b. Design a HTML having a textbox and four buttons named Factorial, Fibonacci, Prime, and Palindrome. When a button is pressed an appropriate function should be called to display
   - Factorial of that number
   - Fibonacci series up to that number
   - Prime numbers up to that number
   - Is it palindrome or not
c. Write a program to validate the following fields in a registration page

i. Name (start with alphabet and followed by alphanumeric and the length should not be less than 6 characters)
ii. Mobile (onlynumbersandlength10 digits)
iii. E-mail (should contain format like xxxxxxx@xxxxxx.xxx)

**Text Books:**

1. ProgrammingtheWorldWideWeb,7<sup>th</sup> Edition, Robet W Sebesta, Pearson, 2013.

2. WebProgrammingwithHTML5, CSS and JavaScript, John Dean, Jones & Bartlett Learning, 2019 (Chapters 1-11).

3. ProMERNStack:FullStackWebAppDevelopmentwithMongo,Express,React,and Node, Vasan Subramanian, 2nd edition, A Press, O'Reilly

**Web Links:**
Infosys spring board*

1. https://www.w3schools.com/html
2. https://www.w3schools.com/css
3. https://www.w3schools.com/js/
4. https://www.w3schools.com/nodejs
5. https://www.w3schools.com/typescript

| **III Year I Semester** | **TINKERING LAB (USER INTERFACE DESIGN USING FLUTTER)** | **L** | **T** | **P** | **C** |
|---|---|---|---|---|---|
| | | 0 | 0 | 2 | 1 |

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

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

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

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

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

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

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

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

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

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

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

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

**Text Book:**
1. Marco L. Napoli, Beginning Flutter: A Hands-on Guide to App Development.
2. Rap Payne, Beginning App Development with Flutter: Create Cross-Platform Mobile Apps 1stEdition, Apres.

**B. Tech III Year – II Semester**

| III Year II Semester | DEEP LEARNING | L | T | P | C |
|---|---|---|---|---|---|
| | | 3 | 0 | 0 | 3 |

**Course Objectives:**
The objective of this course is to cover the fundamentals of neural networks as well as some advanced topics such as recurrent neural networks, long short-term memory cells and convolution neural networks.

**Course Outcomes:**
After completion of course, students would be able to:
- Explore feed forward networks and Deep Neural networks
- Mathematically understand the deep learning approaches and paradigms
- Apply the deep learning techniques for various applications

**UNIT-I:**
Basics- Biological Neuron, Idea of computational units, McCulloch–Pitts unit and Thres holding logic, Linear Perceptron, Perceptron Learning Algorithm, Linear separability, Convergence theorem for Perceptron Learning Algorithm.

**UNIT-II:**
Feed forward Networks- Multilayer Perceptron, Gradient Descent, Backpropagation, Empirical Risk Minimization, regularization, autoencoders.
Deep Neural Networks: Difficulty of training deep neural networks, Greedy layer wise training.

**UNIT-III:**
Better Training of Neural Networks –Newer optimization methods for neural networks (Adagrad, adadelta, rmsprop, adam, NAG), second order methods for training, Saddle point proble minneural networks, Regularization methods (dropout, drop connect, batch normalization).

**UNITIV:**
Recurrent Neural Networks- Back propagation through time, Long Short-Term Memory, Gated Recurrent Units, Bidirectional LSTMs, Bidirectional RNNs.
Convolutional Neural Networks: LeNet, AlexNet. Generative models: Restrictive Boltzmann Machines (RBMs), Introduction to MCMC and Gibbs Sampling, gradient computations in RBMs, Deep Boltzmann Machines.

**UNITV:**
Recent trends- Variational Autoencoders, Transformers, GPT Applications: Vision, NLP, Speech

**TextBooks:**
1. Deep Learning, Ian Good fellow and Yoshua Bengio and Aaron Courville, MIT Press, 2016

**Reference Books:**
1. Neural Networks: A Systematic Introduction, RaúlRojas,1996
2. Pattern Recognition and Machine Learning, Christopher Bishop,2007
3. Deep Learning with Python, François Chollet, Manning Publications,2017

| **III Year II Semester** | **OPERATING SYSTEMS** | **L** | **T** | **P** | **C** |
|---|---|---|---|---|---|
| | | **3** | **0** | **0** | **3** |

**Course Objectives:**

The main objectives of the course are to make student

- Understand the basic concepts and principles of operating systems, including process management, memory management, file systems, and Protection
- Make use of process scheduling algorithms and synchronization techniques to achieve better performance of a computer system.
- Illustrate different conditions for deadlock and their possible solutions.

**UNIT – I:**
**Operating Systems Overview:** Introduction, Operating system functions, Operating systems operations, Computing environments, Free and Open-Source Operating Systems System **Structures:** Operating System Services, User and Operating-System Interface, system calls, Types of System Calls, system programs, Operating system Design and Implementation, Operating system structure, Building and Booting an Operating System, Operating system debugging

**UNIT - II**
**Processes:** Process Concept, Process scheduling, Operations on processes, Inter-process communication. Threads and Concurrency: Multithreading models, Thread libraries, Threading issues. CPU Scheduling: Basic concepts, Scheduling criteria, Scheduling algorithms, Multiple processor scheduling.

**UNIT – III**
**Synchronization Tools:** The Critical Section Problem, Peterson's Solution, Mutex Locks, semaphores, Monitors, Classic problems of Synchronization. Deadlocks: system Model, Deadlock characterization, Methods for handling Deadlocks, Deadlock prevention, Deadlock avoidance, Deadlock detection, Recovery from Deadlock.

**UNIT - IV**
**Memory- Management Strategies:** Introduction, Contiguous memory allocation, Paging, Structure of the Page Table, Swapping. Virtual Memory Management: Introduction, Demand paging, Copy-on-write, Page replacement Allocation of frames, Thrashing Storage Management: Overview of Mass Storage Structure, HDD Scheduling.

**UNIT - V**
**File System:** File System Interface: File concept, Access methods, Directory Structure; Filesystem Implementation: File-system structure, File-system Operations, Directory

implementation, Allocation method, Free space management; File-System Internals: File-System Mounting, Partitions and Mounting, File Sharing. Protection: Goals of protection, Principles of protection, Protection Rings, Domain of protection, Access matrix

**Text Books:**
1. Operating System Concepts, Silberschatz A, GalvinPB, GagneG,10thEdition, Wiley, 2018.
2. Modern Operating Systems, Tanenbaum AS,4th Edition, Pearson ,2016

**Reference Books:**
1. Operating Systems -Internals and Design Principles, Stallings W, 9thedition, Pearson, 2018
2. Operating Systems: A Concept Based Approach, D. M Dhamdhere, 3rd Edition, McGraw-Hill, 2013

**Online Learning Resources:**

1. https://nptel.ac.in/courses/106/106/106106144/
2. http://peterindia.net/OperatingSystems.html

| III Year II Semester | DATA VISUALIZATION | L | T | P | C |
|---|---|---|---|---|---|
| | | 3 | 0 | 0 | 3 |

**Pre-Requisites: Computer Graphics, Image Processing**

**Course Objective:**

- Familiarize students with the basic and advanced techniques of information visualization and scientific visualization
- Learn key techniques of the visualization process
- A detailed view of visual perception, the visualized data and the actual visualization, interaction and distorting techniques

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| CO1 | **Explain** Visualization and representation of data | K6 |
| CO2 | **Creating** visual representations and visualization reference model of applications | K3 |
| CO3 | **Classify** the visualization systems in a data representation | K4 |
| CO4 | **Identify** Visualization of groups and trees | K3 |
| CO5 | **Determine** the visualization of volumetric different data sets in applications | K6 |

**#Based on suggested Revised BTL**

**SYLLABUS:**

**UNIT-I:**
**Introduction**: What Is Visualization? History of Visualization, Relationship between Visualization and Other Fields The Visualization Process, Introduction of visual perception, visual representation of data, Gestalt principles, information overloads.

**UNIT-II:**
Creating visual representations, visualization reference model, visual mapping, visual analytics, Design of visualization applications

**UNIT-III:**
Classification of visualization systems, Interaction and visualization techniques misleading, Visualization of one, two and multi-dimensional data, text and text documents.

**UNIT-IV:**

Visualization of groups, trees, graphs, clusters, networks, software, Metaphorical visualization

**UNIT-V:**
Visualization of volumetric data, vector fields, processes and simulations, Visualization of maps, geographic information, GIS systems, collaborative visualizations, Evaluating visualizations

**Recent trends** in various perception techniques, various visualization techniques, data structures used in data visualization.

**Text Books:**
1. WARD, GRINSTEIN, KEIM.I nteractive Data Visualization: Foundations, Techniques, and Applications. Natick: A K Peters, Ltd.
2. E. Tufte, The Visual Display of Quantitative Information, Graphics Press.

**Resources:**
https://kdd.cs.ksu.edu/Courses/CIS536/Lectures/Slides/Lecture-34-Main_6up.pdf

| III Year II Semester | SOCIAL MEDIA ANALYTICS (PROFESSIONAL ELECTIVE -II) | L | T | P | C |
|---|---|---|---|---|---|
| | | 3 | 0 | 0 | 3 |

**Course Objectives:**
Knowledge on social media and its analytics Course

**Course Outcomes:**
1. Understanding characteristic sand types of social media
2. Knowledge on layers of social media analytics
3. Apply text analysis tools on social media data
4. Understand the significance of action analytics
5. Detect viral topics on social media (YouTube)

**UNIT - I:**
Introduction to Social Media, World Wide Web, Web 1.0, Web 2.0, Web 3.0, Social Media, jCore Characteristics of Social Media, Types of Social Media, Social Networking Sites, Using Facebook for Business Purposes, Content Communities

**UNIT - II:**
Social Media Analytics Overview, Purpose of Social Media Analytics, social media Vs. Traditional Business Analytics, Seven Layers of Social Media Analytics, Types of Social Media Analytics, Social Media Analytics Cycle, Challenges to Social Media Analytics, social media Analytics Tools. Case Study: The Underground Campaign That Scored Big

**UNIT - III:**
Social Media Text Analytics, Types of Social Media Text, Purpose of Text Analytics, Steps in Text Analytics, Social Media Text Analysis Tools. CaseStudy: Tapping Into Online Customer Opinions

**UNIT - IV:**
Social Media Actions Analytics, Introduction to Actions Analytics, Common Social Media Actions, Actions Analytics Tools. Case Study: Cover-More Group

**Unit - V:**
Social Media Hyperlink Analytics Types of Hyperlinks, Hyperlink Analytics, Types of Hyperlink Analytics, Hyperlink Analytics Tools. Case Study: Hyperlinks And Viral YouTube Videos

**Text Books:**
1. Seven Layers of Social Media Analytics Mining Business Insights from Social Media Text, Actions, Networks, Hyperlinks, Apps, Search Engine, And Location Data by Gohar F. Khan Isbn: 1507823207, Isbn-13: 9781507823200

**Reference Books:**
1. Social Media Analytics: Techniques And Insights for Extracting Business Value Out of Social Media by Matthew Ganis, Avinash Kohirkar, Pearson Education.

2. Social Media Analytics: Effective Tools for Building, Interpreting, and Using Metrics, Marshall Sponder, MGH.
3. Big Data and Analytics, Seema Acharya, Subhasinin Chellappan, Wiley Publications.
4. Big Data, Black Booktm, DreamtechPress,2015Edition.

| III Year II Semester | CRYPTOGRAPHY & NETWORK SECURITY (PROFESSIONAL ELECTIVE -II) | L | T | P | C |
|---|---|---|---|---|---|
| | | 3 | 0 | 0 | 3 |

**Course Objectives:**

- Explain the objectives of information security
- Explain the importance and application of each of confidentiality, integrity, authentication and availability
- Understand the basic categories of threats to computers and networks
- Discusses the Mathematics of Cryptography
- Discuss the fundamental ideas of Symmetric and Asymmetric cryptographic Algorithms
- Discusses the Network layer, Transport Layer and Application layer Protocols Enhanced security mechanisms

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| CO1 | Student will be able to understand security issues related to computer networks and learn different symmetric key techniques | K2 |
| CO2 | Students will be able learn mathematic of cryptography for symmetric and Asymmetric algorithms and apply this knowledge to understand the Cryptographic algorithms | K3 |
| CO3 | Students will be able learn different types of symmetric and Asymmetric algorithms | K3 |
| CO4 | Students will be able learn different algorithms of Hash functions, message authentication and digital signature and their importance to the security | K4 |
| CO5 | Students will be able learn different Enhanced security protocols of Application Layer, Transport Layer and Network layer | K4 |

**#Based on suggested Revised BTL**

**SYLLABUS:**
**UNIT – I: Security Concepts:** Introduction, The need for security, Security approaches, Principles of security, Types of Security attacks, Security services, Security Mechanisms, A model for Network Security Cryptography. Classical Encryption Techniques-symmetric cipher model, Substitution techniques, Transposition techniques, Rotor Machines, Steganography.

**UNIT – II: Introduction to Symmetric Cryptography: Algebraic Structures**-Groups, Rings, Fields, GF($2^n$) fields, Polynomials.**Mathematics of Asymmetric cryptography:** Primes, Checking For Primness, Eulers phi-functions, Fermat's Little Theorem, Euler's Theorem, Generating Primes, Primality Testing, Factorization, Chinese Remainder Theorem, Quadratic Congruence, Exponentiation And Logarithm.

**UNIT – III: Symmetric key Ciphers:** Block Cipher principles, DES, AES, Blow fish, IDEA, Block cipher operation, Stream ciphers: RC4, RC5
**Asymmetric key Ciphers:** Principles of public key cryptosystems, RSA algorithm, Diffie-

Hellman Key Exchange, Elgamal Cryptographic system, Elliptic Curve Arithmetic, Elliptic Curve Cryptography.

**UNIT – IV: Cryptographic Hash Functions:** Applications of Cryptographic Hash Functions, Two Simple Hash Functions, Requirements and Security, Hash Functions Based on Cipher Block Chaining, Secure Hash Algorithms (SHA)
**Message Authentication Codes:** Message Authentication Requirements, Message Authentication Functions, Requirements for Message Authentication Codes, Security of MAC'S, MAC'S Based on Hash Functions: HMAC, MAC'S Based On Block Ciphers: DAA And CMAC
**Digital Signatures:** Digital Signatures, Elgamal Digital Signature Scheme, Elliptic Curve Digital Signature Algorithm, RSA-PSS Digital Signature Algorithm.

**UNIT – V: Network and Internet Security: Transport-Level Security:** Web Security Considerations, Transport Level Security, HTTPS, SSH.
**IP Security:** IP Security Overview, IP Security Policy, Encapsulating Security Payload, Authentication Header Protocol.
**Electronic-Mail Security:** Internet-mail Security, Email Format, Email Threats and Comprehensive Email Security, S/MIME, PGP.

**Text Books:**
1. Cryptography and Network Security - Principles and Practice: William Stallings, Pearson Education, 7th Edition, 2017
2. Cryptography and Network Security: Behrouz A. Forouzan Debdeep, Mc Graw Hill, 3rd Edition, 2015

**Reference Books:**
1. Cryptography and Network Security: Atul Kahate, Mc Graw Hill, 3rd Edition
2. Introduction to Cryptography with Coding Theory: Wade Trappe, Lawrence C. Washington, Pearson.

3. Modern Cryptography: Theory and Practice by Wenbo Mao. Pearson

| **III Year II Semester** | **RECOMMENDER SYSTEMS** **(PROFESSIONAL ELECTIVE -II)** | **L** | **T** | **P** | **C** |
|---|---|---|---|---|---|
| | | 3 | 0 | 0 | 3 |

**Course Objectives:**
This course covers the basic concepts of recommender systems, including personalization algorithms, evaluation tools, and user experiences

**Course Outcomes:**
- Describe basic concepts behind recommender systems
- Explain a variety of approaches for building recommender systems
- Describe system evaluation methods from both algorithmic and users' perspectives
- Describe applications of recommender systems in various domains

**UNIT - I:**
**Introduction:** Recommender system functions, Linear Algebra notation: Matrix addition, Multiplication, transposition, and inverses, covariance matrices, Understanding ratings, Applications of recommendation systems, Issues with recommender system.

**UNIT - II:**
**Collaborative Filtering**: User-based nearest neighbor recommendation, Item-based nearest neighbor recommendation, Model based and pre-processing based approaches, Attacks on collaborative recommender systems.

**UNIT - III:**
**Content-based recommendation:** High level architecture of content-based systems, Advantages and drawbacks of content-based filtering, Item profiles, Discovering features of documents, Obtaining item features from tags, Representing item profiles, Methods for learning user profiles, Similarity based retrieval, Classification algorithms.
**Knowledge based recommendation:** Knowledge representation and reasoning, Constraint based recommenders, Case based recommenders.

**UNIT - IV:**
**Hybrid approaches:** Opportunities for hybridization, Monolithic hybridization design: Feature combination, Feature augmentation, Parallelized hybridization design: Weighted, Switching, Mixed, Pipelined hybridization design: Cascade Meta-level, Limitations of hybridization strategies.

**UNIT - V:**
**Evaluating Recommender System:** Introduction, General properties of evaluation research, Evaluation designs, Evaluation on historical datasets, Error metrics, Decision-Support metrics, User-Centred metrics.
**Recommender Systems and communities:** Communities, collaboration and recommender systems in personalized web search, social tagging recommender systems, Trust and recommendations

**Text Books:**
1. Jannach D., Zanker M. and FelFering A., Recommender Systems: An Introduction, Cambridge University Press (2011), 1st ed.
2. Ricci F., Rokach L., Shapira D., Kantor B.P., Recommender Systems Handbook, Springer (2011), 1st ed.

**References:**
1. Manouselis N., Drachsler H., Verbert K., Duval E., Recommender Systems for Learning, Springer (2013), 1st ed.

| III Year II Semester | **CLOUD COMPUTING** **(PROFESSIONAL ELECTIVE -II)** | L | T | P | C |
|---|---|---|---|---|---|
| | | 3 | 0 | 0 | 3 |

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

**UNIT - I**: **Introduction to Cloud Computing Fundamentals**
Cloud computing at a glance, defining a cloud, cloud computing reference model, types of services (IaaS, PaaS, SaaS), cloud deployment models (public, private, hybrid), utility computing, cloud computing characteristics and benefits, cloud service providers (Amazon Web Services, Microsoft Azure, Google App Engine).

**UNIT - II**: **Cloud Enabling Technologies**
Ubiquitous Internet, parallel and distributed computing, elements of parallel computing, hardware architectures for parallel computing (SISD, SIMD, MISD, MIMD), elements of distributed computing, Inter-process communication, technologies for distributed computing, remote procedure calls (RPC), service-oriented architecture (SOA), Web services, virtualization.

**UNIT - III**: **Virtualization and Containers**
Characteristics of virtualized environments, taxonomy of virtualization techniques, virtualization and cloud Computing, pros and cons of virtualization, technology examples (XEN, VMware), building blocks of containers, container platforms (LXC, Docker), container orchestration, Docker Swarm and Kubernetes, public cloud VM (e.g. Amazon EC2) and container (e.g. Amazon Elastic Container Service) offerings.

**UNIT-IV**: **Cloud computing challenges**
Economics of the cloud, cloud interoperability and standards, scalability and fault tolerance, energy efficiency in clouds, federated clouds, cloud computing security, fundamentals of computer security, cloud security architecture, cloud shared responsibility model, security in cloud deployment models.

**UNIT -V**: **Advanced concepts in cloud computing**
Serverless computing, Function-as-a-Service, serverless computing architecture, public cloud (e.g. AWS Lambda) and open-source (e.g. OpenFaaS) serverless platforms, Internet of Things (IoT), applications, cloud-centric IoT and layers, edge and fog computing, DevOps, infrastructure-as-code, quantum cloud computing.

**Text Books:**

1. Mastering Cloud Computing, 2nd edition, Rajkumar Buyya, Christian Vecchiola, Thamarai Selvi, Shivananda Poojara, Satish N. Srirama, Mc Graw Hill, 2024.

2. Distributed and Cloud Computing, Kai Hwang, Geoffery C. Fox, Jack J. Dongarra, Elsevier, 2012.

**Reference Books:**

1. Cloud Computing, Theory and Practice, Dan C Marinescu, 2nd edition, MK Elsevier, 2018.

2. Essentials of cloud Computing, K. Chandrasekhran, CRC press, 2014.

3. Online documentation and tutorials from cloud service providers (e.g., AWS, Azure, GCP)

| III Year II Semester | SENSOR NETWORKS (PROFESSIONAL ELECTIVE -II) | L | T | P | C |
|---|---|---|---|---|---|
| | | 2 | 0 | 0 | 3 |

**Course Outcomes:**

- To provide an overview about sensor networks and emerging technologies.
- To study about the node and network architecture of sensor nodes and its execution environment.
- To understand the concepts of communication, MAC, routing protocols and also study about the naming and addressing in WSN.
- To learn about topology control and clustering in networks with timing synchronization for localization services with sensor tasking and control.
- To study about sensor node hardware and software platforms and understand the simulation and programming techniques.

**UNIT-I: Introduction and Overview:**

Overview of wireless networks, types, infrastructure-based and infrastructure-less, introduction to MANETs (Mobile Ad-hoc Networks), characterise, reactive and proactive routing protocols with examples, introduction to sensor networks, commonalities and differences with MANETs, constraints and challenges, advantages, applications, enabling technologies for WSNs.

**UNIT-II: Architectures:**

Single-node architecture - hardware components, design constraints, energy consumption of sensor nodes, operating systems and execution environments, examples of sensor nodes, sensor network scenarios, types of sources and sinks - single hop vs. multi hop networks, multiple sources and sinks - mobility, optimization goals and figures of merit, gateway concepts, design principles for WNs, service interfaces for WSNs.

**UNIT- III: Communication Protocols**:

Physical layer and transceiver design considerations, MAC protocols for wireless sensor networks, low duty cycle protocols and wakeup concepts - S-MAC, the mediation device protocol, wakeup radio concepts, address and name management, assignment of MAC addresses, routing protocols- classification, gossiping, flooding, energy-efficient routin inicast protocols, multi-path routing, data-centric routing, data aggregation, SPIN, LEACH, Directed-Diffusion, geographic routing.

**UNIT- IV: Infrastructure Establishment:**

Topology control, flat network topologies, hierarchical networks by clustering, time synchronization, properties, protocols based on sender-receiver and receiver-receiver synchronization, LTS, TPSN, RBS, HRTS, localization and positioning, properties and

approaches, single-hop localization, positioning in multi-hop environment, range-based localization algorithms - location services, sensor tasking and control.

**UNIT-V: Sensor Network Platforms and Tools**:

Sensor node hardware, Berkeley motes, programming challenges, node-level software platforms, node-level simulators, state-centric programming, Tiny OS, nesC components, NS2 simulator, TOSSIM.

**Text Books:**

1. Holger Karl & Andreas Willig, "Protocols and Architectures for Wireless Sensor Networks", John Wiley, 2005.

2. Feng Zhao & Leonidas J. Guibas, "Wireless Sensor Networks- An Information Processing Approach", Elsevier, 2007.

**Reference Books:**

1. Kazem Sohraby, Daniel Minoli, & Taieb Znati, "Wireless Sensor Networks- Technology, Protocols, and Applications", John Wiley, 2007.

2. Anna Hac, "Wireless Sensor Network Designs", John Wiley, 2003.

3. Thomas Haenselmann, "Sensor Networks", available online for free, 2008.

4. Edgar Callaway, "Wireless Sensor Networks: Architectures and Protocols", Auerbach, 2003.

| III Year II Semester | SOFTWARE PROJECT MANAGEMENT (PROFESSIONAL ELECTIVE -III) | L | T | P | C |
|---|---|---|---|---|---|
| | | 3 | 0 | 0 | 3 |

**Course Objectives:**

At the end of the course, the student shall be able to:

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

**UNIT-I:**

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

**Evolution of Software Economics:** Software Economics, pragmatic software cost estimation.

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

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

**UNIT-II:**

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

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

**UNIT- III:**

**Model based software architectures:** A Management perspective and technical perspective.

**Work Flows of the process:** Software process workflows, Iteration workflows.

**Checkpoints of the process: Major** mile stones, Minor Milestones, Periodic status assessments.

**Iterative Process Planning:** Work breakdown structures, planning guidelines, cost and schedule estimating, Iteration planning process, Pragmatic planning.

**UNIT- IV:**

**Project Organizations and Responsibilities:** Line-of-Business Organizations, Project Organizations, evolution of Organizations.

**Process Automation:** Automation Building blocks, The Project Environment.

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

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

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

**Reference Books:**
1. Software Project Management, Bob Hughes,3/e, Mike Cotterell, TM
2. Software Project Management, Joel Henry, PEA
3. Software Project Management in practice, Pankaj Jalote, PEA, 2005.
4. Effective Software Project Management, Robert K.Wysocki, Wiley,2006.
5. Project Management in IT, Kathy Schwalbe, Cengage

| III Year II Semester | **QUANTUM COMPUTING** **(PROFESSIONAL ELECTIVE -III)** | L | T | P | C |
|---|---|---|---|---|---|
| | | 3 | 0 | 0 | 3 |

**Course Objectives:**

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

**UNIT - I**

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

**UNIT - II**

**Background Mathematics:** Basics of Linear Algebra, Hilbert space, Probabilities and measurements.

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

**UNIT - III**

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

**UNIT - IV**

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

**UNIT - V**

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

**Text Books:**

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

**Reference Books:**

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

| III Year II Semester | COMPUTER VISION (PROFESSIONAL ELECTIVE -III) | L | T | P | C |
|---|---|---|---|---|---|
| | | 3 | 0 | 0 | 3 |

**Course Objectives:**
- To understand the Fundamental Concepts related to sources, shadows and shading
- To understand the Geometry of Multiple Views

**Course Outcomes:**
- Implement fundamental image processing techniques required for computer vision
- Implement boundary tracking techniques
- Apply chain codes and other region descriptors, Hough Transform for line, circle, and ellipse detections.
- Apply 3D vision techniques and Implement motion related techniques.
- Develop applications using computer vision techniques.

**UNIT – I:**
**CAMERAS:** Pinhole Cameras Radiometry–Measuring Light: Light in Space, Light Surfaces, Important Special Cases Sources, Shadows, And Shading: Qualitative Radiometry, Sources and Their Effects, Local Shading Models, Application: Photometric Stereo, Interreflections: Global Shading Models Color: The Physics of Color, Human Color Perception, Representing Color, A Model for Image Color, Surface Color from Image Color.

**UNIT - II:**
**Linear Filters:** Linear Filters and Convolution, Shift Invariant Linear Systems, Spatial Frequency and Fourier Transforms, Sampling and Aliasing, Filters as Templates, Edge **Detection:** Noise, Estimating Derivatives, Detecting Edges Texture 0: Representing Texture, Analysis (and Synthesis) Using Oriented Pyramids, Application: Synthesis by Sampling Local Models, Shape from Texture.

**UNIT - III:**
**The Geometry of Multiple Views**: Two Views Stereopsis: Reconstruction, Human Stereopsis, Binocular Fusion, Using More Cameras Segmentation by Clustering: What Is Segmentation? Human Vision: Grouping and Getstalt, Applications: Shot Boundary Detection and Background Subtraction, Image Segmentation by Clustering Pixels, Segmentation by Graph-Theoretic Clustering,

**UNIT - IV:**
**Segmentation by Fitting a Model:** The Hough Transform, Fitting Lines, Fitting Curves, Fitting as a Probabilistic Inference Problem, Robustness Segmentation and Fitting Using Probabilistic Methods: Missing Data Problems, Fitting, and Segmentation, The EM Algorithm in Practice, **Tracking With Linear Dynamic Models:** Tracking as an Abstract Inference Problem, Linear Dynamic Models,Kalman Filtering, Data Association, Applications and Examples
**UNIT - V:**

**Geometric Camera Models:** Elements of Analytical Euclidean Geometry, Camera Parameters and the Perspective Projection, Affine Cameras and Affine Projection Equations Geometric Camera Calibration: Least-Squares Parameter Estimation, A Linear Approach to Camera Calibration, Taking Radial Distortion into Account, Analytical Photogrammetry, Case study: Mobile Robot Localization Model- Based Vision: Initial Assumptions, Obtaining Hypotheses by Pose Consistency, Obtaining Hypotheses by pose Clustering, Obtaining Hypotheses Using Invariants, Verification, Case study: Registration in Medical Imaging Systems, Curved Surfaces and Alignment.

**Text Books:**
1. David A. Forsyth and Jean Ponce: Computer Vision – A Modern Approach, PHI Learning (Indian Edition), 2009.

**Reference Books:**
1. E. R. Davies: Computer and Machine Vision – Theory, Algorithms and Practicalities, Elsevier (Academic Press), 4th edition, 2013.
2. R. C. Gonzalez and R. E. Woods "Digital Image Processing" Addison Wesley 2008. 3. Richard Szeliski "Computer Vision: Algorithms and Applications" Springer-Verlag London Limited 2011.

| III Year II Semester | NO SQL DATABASES (PROFESSIONAL ELECTIVE -III) | L | T | P | C |
|---|---|---|---|---|---|
| | | 3 | 0 | 0 | 3 |

**Pre-requisites:** Basic Knowledge about DBMS

**Course Outcomes:** At the end of the Course the student will be able to
**CO1:** Explain and compare different types of NoSQL Databases
**CO2:** Compare and contrast RDBMS with different NoSQL databases.
**CO3:** Demonstrate the detailed architecture and performance tune of Document-oriented NoSQL databases.
**CO4:** Explain the performance tune of Key-Value Pair NoSQL databases.
**CO5:** Apply Nosql development tool so n different types of NoSQL Databases.

**UNIT - I**
Overview and History of NoSQL Databases. Definition of the Four Types of NoSQL Database, The Value of Relational Databases, Getting at Persistent Data, Concurrency, Integration, Impedance Mismatch, Application and Integration Databases, Attack of the Clusters, The Emergence of NoSQL, Key Points.

**UNIT - II**
Comparison of relational databases to new NoSQL stores, MongoDB, Cassandra, HBASE, Neo4j use and deployment, Application, RDBMS approach, Challenges NoSQL approach, Key-Value and Document Data Models, Column-Family Stores, Aggregate-Oriented Databases. Replication and sharding, MapReduce on databases. Distribution Models, Single Server, Sharding, Master-Slave Replication, Peer-to-Peer Replication, Combining Sharding and Replication.

**UNIT - III**
NoSQL Key/Value databases using MongoDB, Document Databases, Document oriented Database Features, Consistency, Transactions, Availability, Query Features, Scaling, Suitable Use Cases, Event Logging, Content Management Systems, Blogging Platforms, WebAnalyticsorReal-TimeAnalytics, E-Commerce Applications, Complex Transactions Spanning Different Operations, Queries against Varying Aggregate Structure.

**UNIT - IV**
Column-oriented NoSQL databases using Apache HBASE, Column-oriented NoSQL databases using Apache Cassandra, Architecture of HBASE, Column-Family Data Store Features, Consistency, Transactions, Availability, Query Features, Scaling, Suitable Use Cases, Event Logging, Content Management Systems, Blogging Platforms, Counters, Expiring Usage.

**UNIT - V**
NoSQL Key/Value databases using Riak, Key-Value Databases, Key-Value Store, Key-Value Store Features, Consistency, Transactions, Query Features, Structure of Data, Scaling, Suitable Use Cases, Storing Session Information, User Profiles, Preferences, Shopping Cart Data, Relationships among Data, Multioperation Transactions, Query by Data, Operations by Sets. Graph NoSQL databases using Neo4, NoSQL database development tools and

programming languages, Graph Databases, Graph Database. Features, Consistency, Transactions, Availability, Query Features, Scaling, Suitable Use Cases.

**Text Books:**
1. Sadalage, P. & Fowler, *NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence*, Wiley Publications,1st Edition,2019.

**Web References :**
1. https://www.ibm.com/cloud/learn/nosql-databases
2. https://www.coursera.org/lecture/nosql-databases/introduction-to-nosql-VdRNp
3. https://www.geeksforgeeks.org/introduction-to-nosql/
4. https://www.javatpoint.com/nosql-databa

| III Year II Semester | **DEEP LEARNING LAB** | L | T | P | C |
|---|---|---|---|---|---|
| | | 0 | 0 | 3 | 1.5 |

**Course Outcomes**: On completion of this course, the student will be able to
- Implement deep neural networks to solve real world problems
- Choose appropriate pre-trained model to solve real time problem
- Interpret the results of two different deep learning models

**Software Packages required:**
- Keras
- Tensorflow
- PyTorch

**List of Experiments:**

1. Implement multi-layer perceptron algorithm for MNIST Handwritten Digit Classification.

2. Design a neural network for classifying movie reviews (Binary Classification) using IMDB dataset.

3. Design a neural Network for classifying news wires (Multi class classification) using Reuters dataset.

4. Design a neural network for predicting house prices using Boston Housing Price dataset.

5. Build a Convolution Neural Network for MNIST Handwritten Digit Classification.

6. Build a Convolution Neural Network for simple image(dogs and Cats) Classification

7. Use a pre-trained convolution neural network (VGG16) for image classification.

8. Implement one hoten coding of words or characters.

9. Implement word embeddings for IMDB dataset.

10. Implement a Recurrent Neural Network for IMDB movie review classification problem.

**Text Books:**

1. Reza Zadehand Bharath Ram sundar, "Tensorflow for Deep Learning", O'Reilly publishers, 2018

**References:**

1. https://github.com/fchollet/deep-learning-with-python-notebooks

| III Year II Semester | **DATA VISUALIZATION LAB** | **L** | **T** | **P** | **C** |
|---|---|---|---|---|---|
| | | **0** | **0** | **3** | **1.5** |

**Course Objectives:**

- To visualize the different datasets using histograms, line charts.

- To understand the use of bar charts and box plots.

- To understand Scatter plots, mosaic plots

- To understand different Map visualizations

- To learn advanced graphs such as correlogram, heatmap and 3D graphs.

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

- Visualize the different datasets using histograms, line charts.

- Makeuse of bar charts and box plots on different datasets

- Apply Scatter plots, mosaic plots in R for different datasets

- Apply different Map visualizations in R

- Create advanced graphs such as correlogram, heatmap and 3D graphs.

**List of Experiments:**

1. a) Load VADeaths(Death Rates in Virginia)dataset in R and visualize the data using different histograms.

   b) Load air quality dataset in R and visualize La Guardia Airport's dialy maximum temperature using histogram.

2. Load AirPassengers dataset in R and visualize the data using line chart that shows increase in air passengers over given time period.

3. a) Load iris dataset in R, visualize the data using different Bar Charts and also demonstrate the use of stacked plots.

   b)Load air quality dataset in R and visualize ozone concentration in air.

4. a) Load iris dataset in R, visualize the data using different Box plots including group by option and also use color palette to represent species.

   b) Load air quality dataset in R and visualize air quality parameters using box plots.

5. Visualize iris dataset using simple scatter, multivariate scatter plot and also visualize scatter plot matrix to visualize multiple variables across each other.

6. Load diamonds dataset in R and visualize the structure in datasets with large data points using hexagon binning and also add color palette then use the

7.  Load HairEyeColor dataset in R and plot categorical data using mosaic plot.

8.  Load mtcars dataset in R and visualize data using heat map.

9.  Install leaflet library in R and perform different map visualizations.

10. Visualize iris dataset using 3d graphs such as scatter3d, cloud, xyplot.

11. Make use of correlogram to visualize data in correlation matrices for iris dataset.

12. Install maps library in R and draw different map visualizations.

**Web References:**

1.  https://www.analyticsvidhya.com/blog/2015/07/guide-data-visualization-r/
2.  https://www.geeksforgeeks.org/data-visualization-in-r/

.

| III Year II Semester | SOFT SKILLS | L | T | P | C |
|---|---|---|---|---|---|
| | (SKILL ENHANCEMENT COURSE) | 0 | 1 | 2 | 2 |

**Course Objectives:**
- To equip the students with the skills to effectively communicate in English
- To train the students in interview skills, group discussions and presentation skills
- To motivate the students to develop confidence
- To enhance the students' interpersonal skills
- To improve the students' writing skills

**UNIT – I**
**Analytical Thinking & Listening Skills:** Self-Introduction, Shaping Young Minds - A Talk by Azim Premji (Listening Activity), Self – Analysis, Developing Positive Attitude, Perception.
**Communication Skills:** Verbal Communication; Non-Verbal Communication (Body Language)

**UNIT – II**
**Self-Management Skills:** Anger Management, Stress Management, Time Management, Six Thinking Hats, Team Building, Leadership Qualities
**Etiquette:** Social Etiquette, Business Etiquette, Telephone Etiquette, Dining Etiquette

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

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

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

**Text books:**
1. Barun K. Mitra, Personality Development and Soft Skills, Oxford University Press, 2011.
2. S.P. Dhanavel, English and Soft Skills, Orient Blackswan, 2010.

**Reference books:**
1. R.S.Aggarwal, A Modern Approach to Verbal & Non-Verbal Reasoning, S.Chand& Company Ltd., 2018.
2. Raman, Meenakshi& Sharma, Sangeeta, Technical Communication Principles and Practice, Oxford University Press, 2011.

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

| III Year II Semester | TECHNICAL PAPER WRITING & IPR | L | T | P | C |
|---|---|---|---|---|---|
| | | 2 | 0 | 0 | - |

**Course Objective:** The course will explain the basic related to writing the technical reports and understanding the concepts related to formatting and structuring the report. This will help students to comprehend the concept of proofreading, proposals and practice

**UNIT- I:**
**Introduction:** An introduction to writing technical reports, technical sentences formation, using transitions to join sentences, Using tenses for technical writing.
**Planning and Structuring:** Planning the report, identifying reader(s), Voice, Formatting and structuring the report, Sections of a technical report, Minutes of meeting writing.

**UNIT-II:**
**Drafting report and design issues:** The use of drafts, Illustrations and graphics.
**Final edits:** Grammar, spelling, readability and writing in plain English: Writing in plain English, Jargon and final layout issues, Spelling, punctuation and Grammar, Padding, Paragraphs, Ambiguity.

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

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

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

**Text Books:**
1. Kompal Bansal & Parshit Bansal, "Fundamentals of IPR for Beginner's", 1st Ed., BS Publications, 2016.
2. William S. Pfeiffer and Kaye A. Adkins, "Technical Communication: A Practical Approach", Pearson.
3. Ramappa,T., "Intellectual Property Rights Under WTO", 2nd Ed., S Chand, 2015.

**Reference Books:**
1. Adrian Wallwork , English for Writing Research Papers, Springer New York Dordrecht Heidelberg London, 2011.

2. Day R, How to Write and Publish a Scientific Paper, Cambridge University Press(2006)

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