

## COMPUTER NETWORKS

L T P C  
3 0 0 4

### COURSE OBJECTIVES:

- ✓ To study about the fundamentals of data communications and Networking.
- ✓ To learn various methods in switching techniques.
- ✓ To apply the concepts of data link layer concepts error and flow control.
- ✓ To understand IP addressing concepts and routing.
- ✓ To know about the applications of transport and application layer protocols.
- ✓ To understand the future of Computer Networks.

### UNIT - 1 NETWORKING PRINCIPLES & LAYERED ARCHITECTURE 6

**Applications:** analyzing social networks, optimizing transportation routes, understanding biological pathways, designing communication networks.

Data Communications and Networking: A Communications Model - Data Communications - Evolution of network, Requirements , Applications, Network Topology (Line configuration, Data Flow), Protocols and Standards, Network Models (OSI, TCP/IP).

### UNIT - 2 CIRCUIT SWITCHING & PACKET SWITCHING 8

**Applications:** Telephone Networks(Circuit Switching)-Foundation for internet, mobile data networks, online gaming(packet switching).

Switched Communications Networks -Circuit Switching -Packet Switching -Comparison of Circuit Switching and Packet Switching -Implementing Network Software, Networking Parameters (Transmission Impairment, Data Rate and Performance)

### UNIT - 3 DATA LINK LAYER 9

**Applications:** reliable data transmission in all real world applications . Streamlined video conferencing(Flow control)

Error Detection and Correction -Hamming Code , CRC, Checksum- Flow control mechanism- Sliding Window Protocol - GoBack - N - Selective Repeat - Multiple access Aloha - Slotted Aloha - CSMA, CSMA/CD -Multiple Access Networks (IEEE 802.3), Token Ring(IEEE 802.5) and Wireless Networks (IEEE 802.11, 802.15)

## 9

IPV4 Address Space -Notations -Classful Addressing -Classless Addressing- Network Address Translation - IPv6 Address Structure - IPv4 and IPv6 header format- Routing-Link State and Distance Vector Routing Protocols- Implementation-Performance- Analysis- Packet Tracer.

## 8

TCP and UDP-Congestion Control-Effects of Congestion-Traffic Management-TCP Congestion Control-Congestion Avoidance Mechanisms-Queuing Mechanisms-QoS Parameters- Application layer-Domain Name System-Case Study : FTP-HTTP-SMTP-SNMP.

## 5

Network as service ( NaaS) -Next Gen WiFi - Software defined WAN(SD WAN)-IPv6 Roll out  
Proliferation of IoT Gadgets

**TOTAL: 30 PERIODS**

At the end of the course the students would be able to

1. explain the basic layers and its functions in computer networks.
2. understand the basics of how data flows from one node to another.
3. analyze routing algorithms.
4. describe protocols for various functions in the network.
5. analyze the working of various application layer protocols.
6. explain the future trends in computer Networks.

1. James F. Kurose, Keith W. Ross, “Computer Networking, A Top-Down Approach

Featuring the Internet”, 8<sup>th</sup> Edition, Pearson Education, 2021.

2. Behrouz A. Forouzan, :”Data Communications and Networking with TCP/IP Protocol Suite”, 6<sup>th</sup> Edition TMH, 2022

#### **REFERENCES:**

1. Larry L. Peterson, Bruce S. Davie, Computer Networks: A Systems Approach, 5<sup>th</sup> , Morgan Kaufmann Publishers Inc., 2012.
2. William Stallings, Data and Computer Communications, 10<sup>th</sup> Edition, Pearson Education, 2013.
3. Nader F. Mir, Computer and Communication Networks, 2<sup>nd</sup> Edition, Prentice Hall, 2014.
4. R. Kelly Rainer , Casey G. Cegielski , Brad Prince, “Introduction to Information Systems”, 5<sup>th</sup> Edition, Wiley Publication, 2014
5. it-ebooks.org

**COURSE OBJECTIVES:**

- ✓ To analyze the asymptotic performance of algorithms.
- ✓ To critically analyze the performance of divide-and-conquer algorithms.
- ✓ To analyze the algorithmic design strategies viz dynamic programming, branch & bound.
- ✓ To understand LPP and backtracking design strategies.
- ✓ To understand algorithmic optimization.
- ✓ To know about the real time applications.

**UNIT - 1 INTRODUCTION TO ALGORITHM ANALYSIS****8**

**Applications:** *Classification of objects, Linear regression, logical problem solving.*

Characteristics of Algorithm - Analysis of Algorithm: Asymptotic analysis of Complexity Bounds – Best, Average and Worst-Case behavior - Performance Measurements of Algorithm - Time and Space Trade-Offs - Analysis of Recursive Algorithms through Recurrence Relations: Substitution Method, Recursion Tree Method and Masters' Theorem - Iterative Algorithms: Measures of Progress and Loop Invariants - Steps to develop Iterative Algorithms

**UNIT - 2 ALGORITHMIC DESIGN STRATEGIES - I****8**

**Applications:** *Creating efficient and effective procedure for sorting, searching,*

Brute-Force technique: Brute Force– String Matching-Exhaustive Search –Traveling Salesman Problem - Knapsack Problem - Assignment problem - Divide and Conquer - Binary Search - Merge Sort - Quick Sort - Multiplication of Large Integers and Strassen's Matrix Multiplication - Closest Pair and Convex - Hull Problems.

**UNIT - 3 ALGORITHMIC DESIGN STRATEGIES - II****8**

**Applications:** *Network operations, Game theory, AI&ML, crew scheduling, and production planning.*

Dynamic programming–Principle of optimality –All Pairs Shortest paths - Warshall's Algorithm - Coin changing problem - Single-Source Shortest Path - Optimal Binary search trees - Memory Functions - Branch and Bound.

**UNIT - 4 ITERATIVE IMPROVEMENT****8**

**Applications:** *In manufacturing industries-production scheduling, inventory and transport companies like Ola, Uber & Rapido.*

Introduction to Linear Programming Problem - Solving LPP using Graphical method - Simplex

method, Backtracking: N Queen's Problem - Sum of Subsets - Graph Coloring and Hamiltonian Cycles.

**UNIT - 5      TRACTABLE AND INTRACTABLE PROBLEMS** 8

**Applications:** *Finding the least-cost cyclic route, Travelling salesman problem, finding prime factors of large number.*

Algorithm Complexities: Non deterministic Algorithms - The classes NP-Hard and NP-Complete - NP-Hard Graph Problems - NP-Hard Scheduling Problems - NP-Hard Code Generation Problems.

**UNIT - 6      REAL TIME APPLICATIONS** 5

Game development – Cryptography – Data mining – Machine Learning – Natural Language Processing – Big data - Navigation Apps – Google Maps.

**TOTAL: 45 PERIODS**

**OUTCOMES:**

At the end of the course the students would be able to

1. Analyze the asymptotic performance of algorithms by providing Optimal Solution.
2. Synthesize divide-and-conquer algorithms, derive and solve recurrences describing the performance of divide-and-conquer algorithms.
3. Analyze Dynamic Programming Problems.
4. Apply iterative improvement technique to solve optimization problems.
5. Understand the different types of algorithms.
6. Understand the real time applications of algorithms.

**TEXT BOOKS:**

1. Alfred Aho, John E.Hopcroft, "Design and Analysis of Computer Algorithms", Pearson Education, 2001.
2. Jon Kleinberg, Eva Tardos, "Algorithm Design", 1<sup>st</sup> Edition ,Pearson Education, 2013.
3. Jeff Edmonds, "How to think about Algorithms", 2<sup>nd</sup> Edition, York University, Cambridge University Press.2024.
4. G.Brassard and P.Bratley "Fundamentals of Algorithms", Pearson, 2015.
5. S.Sridhar," Design and Analysis of Algorithms", Oxford, 2023.

## REFERENCES:

1. Elitz Horowitz and Sartaj Sahani, S.Rajasekaran, "Fundamentals of Computer Algorithms", Universities Press, 2019.
2. M.Weiss, "Data Structures and Algorithm Analysis in C", 2<sup>nd</sup> Edition, Pearson Education, 2018.
3. G.A.V, PAI, "Data Structures and Algorithms", McGraw Hill, 2017.

## COURSE OBJECTIVES:

- ✓ To understand the basics of Operating Systems & Process.
- ✓ To describe about Deadlock and memory management schemes.
- ✓ To analyze File Management & I/O Systems.
- ✓ To understand various functionalities of Linux & Unix
- ✓ To analyze performance of Windows and MAC Operating Systems
- ✓ To compare the functionality of Android & iOS .

## UNIT - 1 INTRODUCTION TO OPERATING SYSTEMS & PROCESSES 8

**Applications:** Smartphones, Gaming Consoles , ATMs and Banking Systems , Air Traffic Control Systems , Web Browsers , Paint.

Operating System Overview - Objectives and Functions - Evolution of Operating System - Operating System Structure - System Calls - System programs. **Process:** Concepts - Operations on Processes- Inter-Process Communication. Scheduling Criteria-CPU Scheduling algorithms. **Thread:** Types- Concept of multithreads- Threading issues- Process Synchronization.

## UNIT - 2 DEADLOCK AND MEMORY MANAGEMENT 8

**Applications:** Resource Allocation, Transaction Processing , Game Synchronization, Efficient Resource Allocation , Memory Allocation for Apps.

**Deadlocks:** Definition- Methods -Conditions - Deadlock Prevention- Avoidance: Bankers Algorithm, Detection and Recovery. Basic Concept - Paging - Structure of Page table- Virtual Memory - Demand paging, Page Replacement algorithms. Allocation of frames, Thrashing.

## UNIT - 3 FILE MANAGEMENT & I/O SYSTEMS 8

**Applications:** Database Management Systems (DBMS) , Cloud Storage & Distributed File Systems, Multimedia Applications (Video Streaming & Editing) , Gaming & Virtual Reality (VR) , Embedded Systems & IoT Devices

**File :** Concept - File types-File operation-Access methods- Directory structure, File System Implementation- - Allocation method, Free-space management. Disk structure- Disk scheduling and Storage Device Management. **I/O Systems:** I/O Hardware-Application I/O Interface.

## UNIT - 4 LINUX & UNIX OPERATING SYSTEMS 8

**Application:** Ubuntu, CentOS, RedHat, AIX, Solaris , Fedora.

**History and Evolution** of Unix and Linux- Overview of Unix and Linux distributions - Similarities and differences - Basic Commands - Text Editor- File System Structure - Networking in Linux/Unix - System Administration and User Management - Troubleshooting and Debugging.

## **UNIT - 5 WINDOWS Vs MAC**

**8**

**Applications:** Banking & ATMs , Gaming & Virtual Reality , VMware & Hyper-V , Cybersecurity & Ethical Hacking , AI & Machine Learning.

Design principles - Features - Interface navigation - File management on MAC Vs Windows - Memory management - File systems - Network connectivity - Security features - System backup and recovery Troubleshooting common issues.

## **UNIT - 6 ANDROID & iOS**

**5**

**Application :** Smartphones, Tablets, Smart TVs, Wearables, IoT devices , Apple HomeKit , Xbox Cloud Gaming

**Mobile OS :** iOS Vs Android - Architecture, Media Layer, Services Layer, Core Operating Systems Layer, File System - User Interface (UI) Design- App Navigation and Controllers - Advanced UI with SwiftUI- Animations and Graphics.

## **COURSE OUTCOMES:**

At the end of the course the students will be able to

CO1 :Understand the basic functionality of Operating Systems & Process .

CO2 :Describe various Deadlock concepts and memory management techniques.

CO3 :Analyze the efficiency of storage management and Optimize Disk Scheduling using Algorithms.

CO4 :Understand various functionalities of LINUX & UNIX

CO5 : Analyze the performance of Windows and MAC Operating Systems.

CO6 :Compare Mobile Operating Systems Architectures & Explore its layers

**45 PERIODS**

## **PRACTICAL EXERCISES**

**30 PERIODS**

1. Basic UNIX Commands and Shell Programming
2. Implementation of Creation and Termination : a) Process b) Thread
3. Implement the following CPU scheduling algorithms:  
a) FCFS b) SJF c) Round Robin d) Priority
4. Program to simulate the concept of Dining-Philosophers problem.
5. Implement algorithm for Deadlock avoidance
6. Implement memory allocation methods: a) Best Fit b) Worst Fit c) Next Fit

7. Implement the following page replacement algorithms: a) FIFO b) LRU c) OPTIMAL
8. Implement the following File Allocation Strategies using C programs  
a) Sequential b) Indexed c) Linked
9. Implementation of Disk Scheduling algorithms
10. Comparison of file management in Windows and MAC OS
11. Installing and Managing applications in ANDROID and iOS

**TOTAL: 75 PERIODS**

### **TEXT BOOKS**

1. Abraham Silberschatz, Peter Baer Galvin and Greg Gagne, "Operating System Concepts" 10<sup>th</sup> Edition, John Wiley and Sons Inc., 2021.
2. "Operating Systems" 3<sup>rd</sup> Edition, by Achyut S. Godbole and Atul Kahate in 2017.
3. "Linux System Programming" by Robert Love, 2nd edition, May 2013.

### **REFERENCES**

- 1 Ramaz Elmasri, A. Gil Carrick, David Levine, "Operating Systems – A Spiral Approach", Tata McGraw Hill Edition, 2010.
- 2 "Modern Operating Systems" by Andrew S. Tanenbaum and Herbert Bos is the 5<sup>th</sup> edition, published in October 2022..
- 3 "Operating Systems: Internals and Design Principles" by William Stallings, 9<sup>th</sup> edition, published in 2018

**COURSE OBJECTIVES:**

- ✓ To introduce the essential software engineering concepts involved.
- ✓ To impart skills in the design and implementation of efficient software systems across disciplines.
- ✓ To familiarize engineering practices and standards used in developing software products and components.
- ✓ To understand the testing strategies.
- ✓ To explain the software evolution.
- ✓ To introduce the future trends in Software engineering for real time applications.

**UNIT - 1 OVERVIEW OF SOFTWARE ENGINEERING 6**

**Applications:** data science, health care, gaming industry.

Nature of Software, Software Engineering, Software process, project, product, Process Models  
Classical Evolutionary models, Overview of System Engineering. Software project scheduling and Tracking.

**UNIT - 2 INTRODUCTION TO SOFTWARE PROJECT MANAGEMENT 8**

**Applications:** developing a medical device, building a online shopping platform.

Planning scope, milestones deliverables, Risk Management, Metrics Measurements.  
Requirements Engineering process Requirement Elicitation, System Modelling - Requirements Specification and Requirement Validation

**UNIT - 3 SOFTWARE DESIGN 9**

**Applications:** E-commerce, Game development, mobile apps, operating systems, software libraries(modularity)

Design concepts and principles - Abstraction - Refinement - Modularity Cohesion coupling,  
Architectural design, Detailed Design Transaction Transformation, Refactoring of designs,  
Object-oriented Design User-Interface Design.

**UNIT - 4 VERIFICATION & VALIDATION 9**

**Applications:** Home Networks, Security enhancement(NAT)-Telephone Networks, Military applications(Routing protocol)

Strategic Approach to Software Testing, Testing Fundamentals Test Plan, Test Design, Test Execution, Reviews, Inspection Auditing.

## **UNIT - 5                      SOFTWARE EVOLUTION**

8

**Applications:** *Making the changes in OS, converts spread sheets in web apps, support to new hardware.*

Software Maintenance, Types of Maintenance, Software Configuration Management, Overview of RE-engineering Reverse Engineering. The Genesis of Agile, Introduction and background, Agile Manifesto and Principles, Overview of Scrum, Extreme Programming, Feature Driven development, Lean Software Development, Agile project management, Design and development practices in Agile projects.

## **UNIT - 6                      FUTURE TRENDS**

5

Software requirement specification- DevOps integration- Agile requirements gathering-Low-code/no-code development-Automated code reviews-AI& Automation- Secure software development Life cycle.

**TOTAL: 45 PERIODS**

### **OUTCOMES:**

At the end of the course the students would be able to

1. Apply the principles of the engineering processes in software development.
2. Demonstrate software project management activities such as planning, scheduling and Estimation.
3. Model the requirements for the software projects.
4. Design and Test the requirements of the software projects.
5. Implement the software development processes activities from requirements to validation and verification.
6. Apply future advancements for real world problems.

### **TEXT BOOKS:**

1. Roger Pressman, "Software Engineering: A Practitioner's Approach", 7th Edition, McGraw hill, 2010.
2. Ian Sommerville, "Software Engineering", 9th Edition, Addison-Wesley, 2016

### **REFERENCES:**

1. Pankaj Jalote, "A Concise Introduction to Software Engineering", Springer, 2008

2. William E. Lewis , “Software Testing and Continuous Quality Improvement”, 3<sup>rd</sup> Edition, Auerbach Publications, 2008.

## COURSE OBJECTIVES:

- ✓ To learn how to design object-oriented programs with Python classes.
- ✓ To understand threading concept and multithreading on Python.
- ✓ To learn about lambda function and regular expression.
- ✓ To develop GUI based applications and Graphics using python.
- ✓ To use Numpy, Pandas and Matplotlib packages.
- ✓ To know about real time applications of python.

## UNIT - 1      OBJECT ORIENTED PROGRAMMING IN PYTHON      8

**Applications:** OOPs concepts in Client Server Systems, OO Databases, Simulation and modeling.

**OOPs in python:** Features of Object Oriented Programming system (OOPs) - classes and objects - Creating a class - the self variable, types of variables, namespaces – inheritance in python, types of inheritance – polymorphism – overloading – overriding – data hiding.

## UNIT - 2      THREADS AND ADVANCED FILE OPERATIONS      8

**Applications:** Multiprocessing, better resource utilization, rendering graphics.

Threads in python: Difference between process and thread - types of threads - benefits of threads - creating threads – multithreading – starting a thread - thread synchronization. File operations – modes of opening a file – seek( ) and tell( ) - working with text file , binary file and CSV file – preparing a pdf file after data processing.

## UNIT - 3      LAMBDA FUNCTION AND REGULAR EXPRESSION      8

**Applications:** Quick data manipulation, less number of lines in coding, flexibility, pattern matching, word (or) character searching.

Lambda function – syntax - advantages over normal function – difference between lambda and def - Regular expression - sequence characters in regular expressions - quantifiers in regular expressions - special characters in regular expressions - using regular expression on files - retrieving information from a file.

## UNIT - 4      GRAPHICAL USER INTERFACE & GRAPHICS IN PYTHON      8

**Applications:** Self ticket booking system, Video games, Designing Web pages.

Graphical user interface: Creating a GUI in python - widget classes - working with Fonts and Colors, working with Frames, Layout manager, Event handling – Graphics in python - Turtle Graphics – turtle attributes and methods - Creating drawings and animations – simple shapes – working with colors and pen size – Using loops and functions – creating designs.

## UNIT - 5 PACKAGES IN PYTHON

8

**Applications:** *Data Analysis and Visualization, real time numerical calculations, image processing, AIML model predictions.*

Introduction to Numpy -Creation of vectors and matrices - Matrix manipulation - Pandas - Pandas data structures – Series and Data Frame - Data wrangling using pandas – Matplotlib - Scatter plot - Line plot - Bar chart.

## UNIT - 6 REAL TIME APPLICATIONS

5

Python in Web Development, Data Science , Artificial Intelligence and Machine Learning, Deep Learning, 3D game development, Web Scrap applications, Search Engine optimization. Familiar companies uses python – Netflix, Facebook, Spotify, Google, AWS uses Django and Flask.

**TOTAL: 45 PERIODS**

### OUTCOMES:

At the end of the course the students would be able to

1. able to understand OOP concepts in Python including inheritance and polymorphism.
2. able to work with files and perform operations on it using Python
3. able to implement regular expression and concept of threads for developing efficient program
4. Designing GUI based applications using Python
5. Knowledge of using Numpy, Pandas and Matplotlib packages.
6. Know the real time applications and familiar companies using python

### TEXT BOOKS:

1. Paul Gries , Jennifer Campbell, Jason Montojo, Practical Programming: An Introduction to Computer Science Using Python 3, Pragmatic Bookshelf, 3rd Edition, 2018.
2. Programming through Python, M. T Savaliya, R. K. Maurya, G M Magar, Revised Edition, Sybgen Learning India, 2020.

### REFERENCES:

1. Advanced Python Programming, Dr. Gabriele Lanaro, Quan Nguyen, Sakis Kasampalis, Packet Publishing, 2019.
2. Programming in Python 3, Mark Summerfield, Pearson Education, 2nd Ed, 2018.
3. Python: The Complete Reference, Martin C. Brown, McGraw Hill, 2018.
4. Beginning Python: From Novice to Professional, Magnus Lie Hetland, Apress, 2017.
5. Programming in Python 3, Mark Summerfield, Pearson Education, 2nd Ed, 2018.

**COURSE OBJECTIVES:**

- ✓ To implement Object Oriented concepts in programming
- ✓ To make use of Pandas and Numpy Libraries
- ✓ To implement graphics and animations using python

**LIST OF EXPERIMENTS:**

30

1. Write a Python class named Person with attributes name, age, weight (kgs), height (ft) and display the details of a person.
2. Write a python program to demonstrate operator overloading
3. Write a Program to demonstrate concept of threading and multitasking in Python.
4. Write a program to Python program to implement various file operations. (text file, binary file and CSV file )  
Write a program to Python program to demonstrate use of regular expression for the following :  
Check the validity of a password given by the user. The password should satisfy the following criteria:
  - i) Contain at least 1 letter between a and z
  - ii) Contain at least 1 number between 0 and 9
  - iii) Contain at least 1 letter between A and Z
  - iv) Contain at least 1 character from \$, #, @
  - v) Minimum length of password: 6
  - vi) Maximum length of password: 12
6. Write a python program to create a module and import the module in another python program.
7. Write a Python program to demonstrate working of OS Module and Calendar Module
8.
  - a) Write a python program to find min, max, sum, cumulative sum of array
  - b) Write a python program to implement Pandas Series with labels.
 Write a GUI Program in Python to design application that demonstrates
  - a. Different fonts and colors
  - b. Different Layout Managers
  - c. Event Handling
9.
  - a. Different fonts and colors
  - b. Different Layout Managers
  - c. Event Handling
10. Write a python program to create different shapes / objects using turtle package.

**TOTAL: 30 PERIODS**

**OUTCOMES:**

At the end of the course the students is expected to be able to

1. Implementing Object Oriented concepts in programming
2. Make use of Pandas and Numpy Libraries
3. Creating graphical objects and animation using python



(XXXXX)  
Subject  
Code

**LIFE SKILL IV – PUBLIC SPEAKING PRACTICES**

L T P C  
2 0 0 1

**COURSE OBJECTIVES:**

- ✓ To develop proficiency in public speaking, including its history, significance, and benefits. **(C 18)**
- ✓ To explore the history and evolution of public speaking. **(C 19)**
- ✓ To engage with the audience, especially in high-pressure situations like press meets.
- ✓ To apply structured debate formats to evaluate arguments critically and develop reasoning skills.
- ✓ To synthesize and articulate ideas effectively by engaging in complex and abstract discussions
- ✓ To demonstrate public speaking skills in real-world contexts by presenting on topics related to education, history, entertainment, and current affairs.

**UNIT - I BASICS OF PUBLIC SPEAKING**

2

History of public speaking – benefits

**Practice** :Effective practices for public speaking- Public speaking topics

**UNIT - II LISTENING TO FAMOUS SPEECHES**

2

TED talks – presentations, Josh talks **(C 20)**

**Practice:** Key elements of TED talk presentations - prioritize simplicity in visuals and strong stage presence

**UNIT - III RESPONDING AND TACKLING THE QUESTIONS**

2

Facing the audience (Press meet)

**Practice:** Engaging with the audience in a press meet – Handling Challenging questions **(C21)**

**UNIT - IV GROUP DEBATE**

2

Dos and Don'ts of group debate – Practicing various types of debate format **(C 22)**

**Practice** : Structuring a debate session - Focusing on moral reasoning **(C 23)**

## **UNIT - V      GROUP DISCUSSION LEVEL 2**

2

Engaging in discussions on complex and contemporary topics (C 24)

**Practice:** Argument structure & logical flow

## **UNIT - VI      PUBLIC SPEAKING PRACTICES**

2

Topics in Education, History, Entertainment, Current affairs

**Practice:** Content, delivery, engagement, and overall effectiveness

## **TOTAL: 12 PERIODS**

### **OUTCOMES:**

At the end of the course the students would be able to

1. Initiate and manage professional conversations with correct greetings and courtesies.
2. Demonstrate proficiency in asking clear and appropriate questions in professional and technical conversations.
3. Adjust their tone and intonation based on the context of the conversation.
4. Demonstrate confidence in handling one-on-one interviews and personal conversations through effective communication strategies.
5. Students will be able to formulate and ask open-ended brainstorming questions.
6. Understand the importance and purpose of group discussions in professional and academic settings.

### **REFERENCES:**

1. “Advanced Communication Skills” by Mathew Richardson, Charlie Creative Lab, 2020
2. Andy Gillett, Using English for Academic purposes for students in higher Education. <https://www.uefap.org/reading/>
3. Meenakshi Raman & Sangeeta Sharma (2017). Technical Communication: Principles and Practice (3rd Edition). Oxford University Press, India.
4. Rajendra Pal & J.S. Korlahalli (2011). Essentials of Business Communication. Sultan Chand & Sons, New Delhi.

## CO – PO MAPPING

| CO                            | PO |   |   |   |   |   |   |   |   |    |    |    | PSO |   |   |
|-------------------------------|----|---|---|---|---|---|---|---|---|----|----|----|-----|---|---|
|                               | 1  | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 1   | 2 | 3 |
| 1                             | -  | - | - | - | - | - | - | - | 3 | 3  | -  | 3  | -   | - | - |
| 2                             | -  | - | - | - | - | - | - | - | 3 | 3  | -  | 3  | -   | - | - |
| 3                             | -  | - | - | - | - | - | - | - | 3 | 3  | -  | 3  | -   | - | - |
| 4                             | -  | - | - | - | - | - | - | - | 3 | 3  | -  | 3  | -   | - | - |
| 5                             | -  | - | - | - | - | - | - | - | 3 | 3  | -  | 3  | -   | - | - |
| LOW (1); MEDIUM (2); HIGH (3) |    |   |   |   |   |   |   |   |   |    |    |    |     |   |   |