DIPLOMA WING

RAJIV GANDHI PROUDYOGIKI VISHWAVIDYALAYA, BHOPAL

SCHEME OF STUDIES & EXAMINATIONS (IMPLEMENTED FROM SESSION: JULY 2023)

SCHEME	NAME OF BRANCH	BRANCH CODE	SEMESTER
OCBC JULY 2022/2023	COMPUTER SCIENCE AND ENGINEERING	C04	THIRD (III)

S.N.	PAPER CODE	SUBJECT CODE	SUBJECT NAME	THEORY COMPONENT								PRACTICAL COMPONENT					TOTAL CREDITS	TOTAL MARKS
				HRS PER WEEK	CREDITS	TERM WORK				THEORY PAPER		HRS PER WEEK	CREDITS	LAB WORK	PRACTICAL EXAM/VIVA			
						QUIZ/ASSIGNMENT	MID TERM TEST*		TOTAL	MARKS	DURATION				MARKS	DURATION		
							I	II										
1	7481	301	COMPUTER PROGRAMMING	3	3	10	10	10	30	70	03 Hrs.	4	2	20	30	03 Hrs.	5	150
2	7482	302	SCRIPTING LANGUAGES	3	3	10	10	10	30	70	03 Hrs.	4	2	20	30	03 Hrs.	5	150
3	7483	303	DATA STRUCTURES	3	3	10	10	10	30	70	03 Hrs.	4	2	20	30	03 Hrs.	5	150
4	7484	304	COMPUTER SYSTEM ORGANISATION	4	4	10	10	10	30	70	03 Hrs.	0	0	0	0	0	4	100
5	7485	305	ALGORITHMS	4	4	10	10	10	30	70	03 Hrs.	0	0	0	0	0	4	100
6			**SUMMER INTERNSHIP- I	0	0	0	0	0	0	0	0	0	2	20	30	03 Hrs.	2	50
7			PROFESSIONAL DEVELOPMENT	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0
8			***RECOVERY CLASSES/LIBERARY etc.	0	0	0	0	0	0	0	0	5	0	0	0	0	0	0
TOTAL				17	17				150	350		19	8	80	120		25	700

NOTE - (1) * Two Best, out of Three Mid Term Tests (Progressive Tests) Marks should be entered here.

(2) ** 3-4 Weeks Summer Internship after II Semester.

(3) ***To recover courses if session delays due to summer internship.

GRAND TOTAL OF CREDITS	GRAND TOTAL OF MARKS
25	700

DIPLOMA WING

RAJIV GANDHI PROUDYOGIKI VISHWAVIDYALAYA, BHOPAL

DIPLOMA IN COMPUTER SCIENCE AND ENGINEERING (C04)

SEMESTER III

COURSE TITLE	:	COMPUTER PROGRAMMING
PAPER CODE	:	7481
SUBJECT CODE	:	301
THEORY CREDITS	:	03
PRACTICAL CREDITS	:	02

Course Learning Objectives:

To enable student, develop structured solutions to problems and implementing them using computers. This involves two parts:

I) Formulating a solution for a given problem as a well-defined sequence of actions, and

II) Expressing solution in a machine-readable form or a programming language. For the second part, we will learn the common units of programming languages. The first part can only be learned through the repeated practice of solving problems.

Course Outcomes :

After completing the course, students will be able to:

1. Apply computational thinking to solve problems logically using appropriate variables, data representations.
2. Demonstrate operators and read and write operations.
3. Illustrate if-else and loops to control flow of the program.
4. Practice use of arrays, strings and pointers.
5. Employ functions and recursion for problem solving.

Course Content:

The language of choice will be C. The focus will be on problem solving and problem where these ideas can be applied. The main focus of the class will to take examples of problems where these ideas can be employed.

Unit	Topics and Sub-topics	Hours	Marks
Unit 1: Introduction to Computational Thinking & Variables and Data Representation	• Concept of problem-solving and computational thinking, Steps involved in Problem Solving. • Components of computational thinking. Converting problems into Flowchart and Algorithms. • Variables, constants, Data types and Memory representation of variables	21	14
Unit 2: Operators and Input-Output	• Types of operators-Arithmetic, Relational, Logical and Bitwise Operators. Operator precedence and associativity. • Input and output functions, formatted output. • Basic file I/O - file modes, open, read, write and close.	20	14

Unit 3: **Conditional and Looping Constructs**	• Conditional Statement - if, if-else, nested if, Conditional Operator. • Looping Structure - for, while, do-while, Nested structure and related use cases. • Switch-Case, Continue & Break statement.	22	14
Unit 4: **Arrays and Pointer**	• Introduction to Arrays. One dimensional and multidimensional arrays. Memory organization of an array. • Strings-Create, use and modification of a string, String functions. • Pointers – Concept of pointer- declaration, initialization, and their use. • Dynamic memory allocation.	22	14
Unit 5: **Functions and Recursion**	• Functions - Function declaration, Scope and Lifetime of Variables in Functions, types of function. • Call by value vs Call by reference. • Recursion -Concept and Uses, Characteristics of Recursive Functions, Types of Recursions and Examples.	20	14

Suggested List of Practical:

1. Familiarization with the programming environment (Editor, Compiler, etc.)
2. Programs using I/O statements and various operators
3. Programs using expression evaluation and precedence
4. Programs using decision making statements and branching statements
5. Programs using loop statements
6. Programs to demonstrate applications of n dimensional arrays
7. Programs to use string manipulation functions
8. Programs demonstrating function using different parameter passing mechanism
9. Programs to implement recursion
10. Programs to solve the problems using pointers
11. Programs to demonstrate dynamic memory allocation
12. Programs to demonstrate file operations

Suggested Learning Resources:

1. Let Us C, Yashvant Kanetkar
2. Problem Solving and Programming in C, R.S. Salaria, Khanna Publishing House
3. C Programming Absolute Beginner's Guide, Dean Miller and Greg Perry
4. The C Programming Language, Kernighan and Ritchie, Prentice Hall of India
5. Programming in ANSI C, E. Balagurusamy, Tata McGraw-Hill
6. C Programming & Data Structures, B. A. Fouruzan and R. F. Gilberg, CENGAGE Learning.
7. Outline of Programming with C, Byron Gottfried, Schaum, McGraw-Hill

******

DIPLOMA WING
RAJIV GANDHI PROUDYOGIKI VISHWAVIDYALAYA, BHOPAL
DIPLOMA IN COMPUTER SCIENCE AND ENGINEERING (C04)

SEMESTER III

COURSE TITLE	:	SCRIPTING LANGUGAGES
PAPER CODE	:	7482
SUBJECT CODE	:	302
THEORY CREDITS	:	03
PRACTICAL CREDITS	:	02

Course Learning Objectives:

To learn how to work with a scripting language.

Course Outcomes:

By the end of the course, students should be able to:

1: Understand the Fundamentals of scripting programming

2: Identify and explain advanced data types.

3: Implement Control Structures and Looping mechanisms

4: Organize Python Programs Using Functions and Modules

5: Handle File Operations and Perform Text Processing

Course Content:

Unit	Topics and Sub-topics	Hours	Marks
Unit 1: **Introduction, Variables and Data Types**	• History, Version and features of python. • Python installation and environment setup. • Working with python- Writing and running Code. • Basic syntax- input, output, comments and indentation. • Types of error. • Understanding variables, basic data types – numeric, string, Boolean etc. • String operations, Basic operators.	21	14
Unit 2: **Advanced Data types**	• Introduction to advanced data types. • Lists – creating, accessing, modifying slicing and iterating operations on list. • Tuples – creating tuples, accessing elements, understanding immutability, slicing, unpacking, and iterating over tuples. • Dictionaries – creation, accessing values, adding and modifying key-value pairs, deleting items, and iteration. • Sets – creation, adding and removing elements. • Arrays – introduction to arrays, importing array module, creating arrays, accessing and modifying elements, and operations.	22	14
Unit 3: **Control Structures**	• Understanding coding blocks. • Conditional blocks using if, else and elif. • For loops and iterations with iterating over sequences and range. • While loop, loop manipulation in Python using continue, break, else, and pass. • Programming using conditional and loops block	20	14
Unit 4: **Functions,**	• Organizing Python codes using functions- Introduction to functions, defining and calling functions, function parameter- Positional Parameters, Default Parameters, Keyword	22	14

Modules and Packages	Arguments, return values, variable scope and recursion. • Organizing Python projects into module - Introduction, Create user defined Module, Namespace and Scoping • Importing own module as well as external modules. • Understanding Packages -Create user defined Package, importing User defined and built-in packages, installing packages using PIP.		
Unit 5: **File I/O, Text Processing, Regular Expressions**	• File handling modes. • Understanding file related operations -read, write, open, append, rename delete and close. • Basic understanding of exception handling. • Powerful pattern matching and searching with python re module. • Power of pattern searching using regex.	20	14

Suggested List of Practical:

1. Practice Basic Coding Syntax
   - Practical 1.1: Write a script to display "Hello, World!"
   - Practical 1.2: Practice using print (), comments, and basic expressions
2. Write and Execute Scripts Based on Data Types
   - Practical 2.1: Accept user input and perform simple arithmetic operations
   - Practical 2.2: Perform operations on strings (slicing, formatting, functions)
3. Write and Execute Scripts Based on Advanced Data Types
   - Practical 3.1: Implement list operations (append, insert, remove, slicing, length)
   - Practical 3.2: Implement tuple operations (create, access, slicing and iteration)
   - Practical 3.3: Implement dictionary operations (create, access, update, remove and iteration)
4. Write and Execute Python Scripts with Conditionals and Loops
   - Practical 4.1: Use if-else to check whether a number is even or odd
   - Practical 4.2: Print multiplication table using for loop
5. Write and Execute Scripts Based on Functions and Modules
   - Practical 5.1: Create a function to calculate the factorial of a number.
   - Practical 5.2: Create a user-defined module and import it into another script
6. File Processing Scripts
   - Practical 6.1: Read contents from a text file and display line by line
   - Practical 6.2: Write user input data to a file and append it
7. Write and Execute Regular Expressions
   - Practical 7.1: Validate university roll number and determine branch code and college code.
   - Practical 7.2: Validate email addresses using regex

Suggested Learning Resources:

S.No.	Title of Book	Author(s)	Publication
1	Taming Python by Programming	Jeeva Jose	Khanna Publishing House
2	Starting Out with Python	Tony Gaddis	Pearson
3	Core Python Programming,	Wesley J. Chun	Prentice Hall
4	Python Programming: Using Problem Solving Approach	Reema Thareja	Oxford University
5	Introduction to Computation and Programming Using Python.	John V. Guttag	MIT Press.
6	Beginning Python using Python 2.6 and Python 3	James Payne	Wrox publishing
7	Practical Programming: An Introduction to Computer Science using Python 3	Paul Gries	The Pragmatic Bookshelf

****

DIPLOMA WING
RAJIV GANDHI PROUDYOGIKI VISHWAVIDYALAYA, BHOPAL
DIPLOMA IN COMPUTER SCIENCE AND ENGINEERING (C04)

SEMESTER III

COURSE TITLE	:	DATA STRUCTURES
PAPER CODE	:	7483
SUBJECT CODE	:	303
THEORY CREDITS	:	03
PRACTICAL CREDITS	:	02

Course Learning Objectives:

To provide strong foundation for implementing programming language to formulate, analyze and develop solutions related to various data structures problems

Course Outcomes:

By the end of the course, students should be able to:

1. Apply basics of Data structures
2. Apply linear data structure in linear problem solving
3. Explain linked list and perform various operations on it
4. Explain tree data structure and it's representation
5. Explain graph data structure and it's representation

Course Content:

Unit	Topics and Sub-topics	Hours	Marks
Unit 1: Basics of Data structure	• **Classification of Data Structure:** Linear, Non-Linear, Primitive, Non-Primitive, etc. • **Pointers:** Introduction, Declaring and initializing pointers, Accessing variables using pointers, Pointer arithmetic, Array of Pointers, Row-major & Col-major implementation of 2-D array • **Structure:** Definition, Declaration, Initializing Structure, • **Operations on Data Structures :** Traversing, Searching, Insertion, Deletion, Sorting, Merging, Updating.	21	14
Unit 2: Searching and Sorting Techniques	• **Searching Techniques:** Linear search and Binary search. • **Sorting Techniques:** Insertion sort, Selection sort, Bubble sort, quick sort, and heap sort	14	14
Unit 3: Linear data Structures	• **Stacks:** Introduction to Stacks, Array Representation of Stacks, Applications of Stacks. Infix-to-Postfix Transformation, Postfix Expressions. • **Queues:** Introduction to Queues, Array Representation of Queues, Types of Queues-DeQueue, Circular Queue, Applications of Queues	21	14
Unit 4: Linked List	• **Linked List :** Singly Linked List, Representation in Memory, Operations on a Singly Linked List (traversal, insertion, deletion, and searching), Circular Linked Lists, Doubly Linked Lists.	21	14
Unit 5: Non Linear data structure	• **Trees**: Basic Terminologies (root, nodes, edges, parent, child, sibling, leaf, subtree, degree, level, and height), Binary Trees, Representations of a Binary Tree using Arrays, Operations on a Binary Tree - Insertion, Deletion, Traversals, Types of Binary Trees : full binary trees, complete	28	14

	binary trees, perfect binary trees, balanced binary trees • **Graphs**: Vertices, edges, degree, walks, paths, cycles , Representation of Graphs- Set, Linked, Matrix, Graph Traversals – BFS and DFS		

Suggested List of Practical:

S. No.	Topic
1	Write a program using recursive and non-recursive functions to perform search opera- tion in a given list of integers using linear search technique
2	Search operation in a given list of integers using binary search technique
3	Write a program to implement insertion sorting for a given random data
4	Write a program to implement bubble sorting for a given random data
5	Write a program to implement quick sorting for a given random data
6	Write a program to implement selection sorting for a given random data
7	Write a program to implement heap sorting for a given random data
8	Write a program to implement single linked list
9	Write a program to implement double linked list
10	Write a program to implement circular linked list
11	Write a program to Implement Stack operations using array and linked list
12	Write a program to Implement Queue operations using array and linked list.
13	Write a program to implement Breadth First Search (BFS)
14	Write a program to implement Depth First Search (DFS)
15	Write a program to implement a binary tree of integers
16	Write a program to find the minimum depth of a binary tree

Remark : Use 'C' as programming language for the purpose.

Suggested Learning Resources:

1. Data Structures, Schaum Lipschutz, Tata Mcgraw-hill
2. Data Structure Through C, Yashavant Kanetkar, BPB Publications
3. Data Structures, R.S. Salaria, Khanna Book Publishing
4. Data Structures Using C, Reema Thareja, Oxford University Press India.
5. Classic Data Structures, Samanta Debasis, Prentice Hall of India.
6. Fundamentals of Data Structure in C, Horowitz, Ellis, Sahni, Sartaj, Anderson-Freed, Susan, University Press, India.
7. Data Structures: A Pseudo code approach with C, Richard F. Gilberg, Behrouz A. Forouzan, CENGAGE Learning, India.
8. Data Structures and Algorithms: Concepts, Techniques and Applications, G. A. V. Pai, McGraw- Hill Education, India.

****

DIPLOMA WING
RAJIV GANDHI PROUDYOGIKI VISHWAVIDYALAYA, BHOPAL
DIPLOMA IN COMPUTER SCIENCE AND ENGINEERING (C04)

SEMESTER III

COURSE TITLE	:	COMPUTER SYSTEM ORGANISATION
PAPER CODE	:	7484
SUBJECT CODE	:	304
THEORY CREDITS	:	04
PRACTICAL CREDITS	:	00

Course Learning Objectives:

To have a thorough understanding of the basic structure and operation of a digital computer, its architectures and computational designs.

Course Outcomes:

By the end of the course, students should be able to:

1. Understand basic structure and operation of Digital Computer
2. Describe Instruction sets and Instruction Format used in Digital Computer.
3. Discuss Control Unit Organization
4. Interpret Memory and I/O organization
5. Illustrate functioning of 8085 and assembly language programming

Course Content:

Unit	Topics and Sub-topics	Hours	Marks

Unit 1:
 	
Basic Computer Structure
	• Functional blocks: CPU, Memory, Input/Output, Control unit
	• Von Neumann vs Harvard architectures & the bottleneck
	• Bus structures- Data, Address and Control Bus, basic operational concepts.
	• Number System – Decimal, Binary, Octal, Hexadecimal and conversion.
	• Registers , types of Registers and their functions, Instruction Cycle
	• Register Transfer , Bus and Memory Transfer , RTL
	• Microoperations – Arithmetic, Logical and Shift microoperations.
	• ALU and Arithmetic Logical Shift Unit.
	• Data Representation:
	  ○ Fixed point representation of Integers, Decimal numbers,
	  ○ Floating point Representation.
	  ○ Negative number Representation – Using sign bit, 1's Complement and 2's Complement.
	  ○ Arithmetic addition, subtraction using 2's complement
		15	14

Unit 2: **Instruction Set Architecture**	• Instruction code, addressing modes, assembly vs machine code • Instruction cycle, timing and control • General register organization and stack architecture. • Instruction Set, Instruction Format – Three address, Two address, One address, Zero Address • Data Transfer Instructions, Data Manipulation Instructions, Program Control Instructions, Interrupts and interrupt handling • RISC and CISC • Pipeline architecture, Vector processing and Array processing	15	14

Unit 3: **Control Unit Organization**	• Role of control unit and control signals • Hardwired Control Unit: Concept, advantages, limitations • Microprogrammed Control Unit: Architecture, microinstructions • Microinstruction formats: Horizontal vs Vertical • Control memory and address sequencing • Comparison: Hardwired vs Microprogrammed with examples	10	14
Unit 4: **Memory and I/O Organization**	• Memory hierarchy, SRAM and DRAM • RAM, ROM, EPROM chips and their interfacing with CPU. • I/O configuration and interfacing. • Interrupt-driven I/O, handshaking, and asynchronous transfer • DMA (Direct Memory Access) and I/O processors	10	14
Unit 5: **Microprocessor 8085 Case study and Assembly Language Programming**	• 8085 Microprocessor - Introduction to 8085 Microprocessor, Architecture of 8085, Functional Blocks: ALU, Registers, Control Unit, Pin Diagram and Signal Description, Instruction Set Overview, Addressing Modes, Applications of 8085. • Assembly Programming using GNUSim8085: Format, Directives, Flags, - Sample Programs: Addition (8-bit, 16-bit), Data Transfer, Comparisons, Loops	10	14

Suggested Learning Resources:

S. No.	Title of Book	Author(s)	Publication
1	Computer System Architecture	M.Moris Mano	Pearson/PHI
2	Microprocessor Interfaces	Douglas Hall	Tata Mc Graw Hills
3.	Computer Organization and Design – ARM Edition	Patterson & Hennessy	Elsevier

****

DIPLOMA WING
RAJIV GANDHI PROUDYOGIKI VISHWAVIDYALAYA, BHOPAL
DIPLOMA IN COMPUTER SCIENCE AND ENGINEERING (C04)

SEMESTER III

COURSE TITLE	:	ALGORITHMS
PAPER CODE	:	7485
SUBJECT CODE	:	305
THEORY CREDITS	:	04
PRACTICAL CREDITS	:	00

Course Learning Objectives:

The objective of this course is to prepare the student with the algorithmic foundations of computing. A sound grasp of algorithms is essential for any computer science engineer. Almost all programming involves algorithms at some level.

Course Outcomes:

By the end of the course, students should be able to:
1. Explain fundamental concepts of algorithms.
2. Describe various sorting techniques.
3. Describe various searching technique.
4. Illustrate concepts of graphs, minimum spanning tree and shortest path.
5. Explain concepts of data compression techniques.

Course Content:

Unit	Topics and Sub-topics	Hours	Marks
Unit 1: Fundamental of Algorithms	**Introduction to Algorithms**: • Definition, Characteristics and Importance • Steps in algorithm design **Programming Models**: • Pseudocode representation • Flowcharts and structured approach • Iterative Vs. Recursive processes **Algorithm Analysis**: • Time and Space complexity: Concept • Asymptotic notations: Big-O, Omega, Theta • Best-case, Worst-case, and Average-case analysis • Analysis of simple algorithms like addition and multiplication of array	14	14
Unit 2: Sorting	**Introduction to Sorting**: • Importance of sorting in data organization • Applications of sorting **Elementary Sorting Methods**: • Bubble Sort: concept, process and complexity • Selection Sort: concept, process and complexity • Insertion Sort: concept, process and complexity • Merge Sort and Quick sort: concept of divide and conquer and complexity by recurrence relation **Comparison of Sorting Methods**: • Relative performance (best, worst, average case) • Situations where each method is useful	12	14

Unit 3: Searching	**Searching Techniques**: • Linear search: concept, process, and complexity • Binary search: concept, process and complexity by recurrence relation **Binary Search Trees (BST)**: • Structure and properties of BST • Insertion, deletion and search operations **Balanced Search Trees**: • Need for balancing • Concept of height balancing **Hash Tables**: • Definition and purpose of hashing • Hash functions: Concept • collision handling techniques: concept, overview of chaining, open addressing	10	14
Unit 4: Graphs	**Introduction to Graphs**: • Directed and Undirected graphs • Graph representation: adjacency list and matrix • Paths, Cycles, Connected/Disconnected graphs • Spanning Trees **Graph Algorithms**: • Topological Sorting: concept, example • Finding Minimum Spanning Trees: Prim's and Kruskal's algorithms • Finding Shortest Path: Dijkstra's algorithm	14	14
Unit 5: Strings	**String Sorting**: • concept, process • Applications in text processing **Substring Search**: • concept, process **Regular Expressions**: • Meaning and importance • Basic operators and patterns • Applications: validation of email, phone numbers **Elementary Data Compression**: • Need for data compression • Huffman coding: concept and working • Applications in file storage and transmission	10	14

Suggested Learning Resources:

S. No.	Title of Book	Author(s)	Publication
1.	Analysis & Design of Algorithm	Horowitz & Sahani	Universities Press
2.	Algorithms	Dasgupta	TMH
3.	Design and Analysis of Computer Algorithms	Alfred & Ullman	The (Addison-Wesley Series in Computer Science and Information Processing)
4.	Algorithm Design	Michael T Goodrich, Robarto Tamassia.	Wiley India
5.	Design and analysis of algorithm	S.Sridhar	Oxford University Press
6.	Introduction to Algorithms	Cormen Thomas,Leiserson CE, Rivest RL	PHI

****

DIPLOMA WING
RAJIV GANDHI PROUDYOGIKI VISHWAVIDYALAYA, BHOPAL
DIPLOMA IN COMPUTER SCIENCE AND ENGINEERING (C04)

SEMESTER III

COURSE TITLE	:	SUMMER INTERNSHIP - I
PAPER CODE	:	--
SUBJECT CODE	:	--
THEORY CREDITS	:	00
PRACTICAL CREDITS	:	02

SUMMER INTERNSHIP –

3-4 weeks summer internship after IInd Semester.

It should be undertaken in an industry/Govt. or Pvt. Certified Agencies which are in social sector/Govt. Skill Centers/Schemes.

Evaluation is based on work done, quality of report, performance in viva-voce, presentation etc.

****

DIPLOMA WING
RAJIV GANDHI PROUDYOGIKI VISHWAVIDYALAYA, BHOPAL
DIPLOMA IN COMPUTER SCIENCE AND ENGINEERING (C04)

SEMESTER III

COURSE TITLE	:	PROFESSIONAL DEVELOPMENT
PAPER CODE	:	--
COURSE CODE	:	--
THEORY CREDITS	:	00
PRACTICAL CREDITS	:	00

Course Objectives:

Following are the objectives of this course:-

(1) To learn the principles of Professional and Social ethics.
(2) To know the concept of Lifelong learning and Self-directed learning.
(3) To present self for employment.
(4) To introduce the need of industrial visits.
(5) To understand CV, Resume, Bio-data and Interview and their significance.
(6) To develop the skills of Group Discussion.

Course Content:

Unit – I Professional and Social Ethics

Professional ethics, its need and importance, general code of ethics for engineers, ethical issues for engineers.

Need and importance of social skills, social skills for better group performance, important social skills such as social perceptiveness, coordination, negotiation, persuasion etc.

Unit – II Lifelong learning and Self-directed Learning

Lifelong learning, its examples, self-directed learning, its examples, important steps in lifelong learning.

Need for planning self-directed learning, planning self-directed learning plan, examples.

Unit – III Career Planning

Importance of career planning, major career opportunities in concerned branch of engineering , study of the important career opportunities regarding qualification, knowledge, skills, experience required for them, role of personal factors like personal life style, interest areas, desires, personal preferences in career planning.

Identification and detailing of important career opportunities in relation to branch of diploma, identification and detailing of important self-personal factors and self-personal preferences, development of self-career plan.

Unit – IV Industrial Visits

Necessity of exposure to environment and practices, lectures by industry experts.

Importance of Students' industrial visits, learning through observing real life industrial systems, planning and organizing the industrial visits.

Unit – V CV, Resume, Bio-data and Interview

Need of presenting self for employment, salient features and formats of bio-data, CV, resume, comparison of the three for their merits, limitations and specific uses, study of cases and examples of bio-data, CV, resume and covering letter by all students for self of for the given cases.

Importance of employment related interviews, purpose of interview, dress code, body language and posture of interviewee, do's and don'ts for interviews, interview checklist, practice of facing employment related interviews for all students.

Unit – VI Group Discussion

Need and importance of group discussion in professional work, ideal group discussion and skills needed to effectively participate in group discussion, practice of group discussion skills.

Course Outcomes:

After completing this course, the student will be able to:-

(1) Demonstrate his/her understanding of Professional and Social ethics.
(2) Plan self-learning and self-directed learning for completing the task.
(3) Suggest an action plan for his career planning.
(4) Demonstrate his/her learning from visits to industry.
(5) Prepare CV, Resume and Bio-data along with a covering letter for a job.
(6) Effectively face an interview.
(7) Participate in Group discussion.

****