

**DIPLOMA CURRICULUM OF
INFORMATION TECHNOLOGY
(SECOND YEAR)
(3rd Semester)**

(To be implemented from 2025-26)

Prepared by;



**National Institute of Technical Teachers' Training & Research Kolkata
Block – FC, Sector – III, Salt Lake City, Kolkata – 700106**

Vetted by:

Domain experts from Polytechnics of Odisha



**State Council for Technical Education & Vocational Training
Near Raj Bhawan, Unit-VIII, Bhubaneswar, Odisha**

Table of Contents

1	Curriculum Structure for Second year (Semester III)	3
2	Content details of Semester III	4 - 24

Teaching and Evaluation Scheme for Third Semester Information Technology

				Study Scheme				Evaluation Scheme					
Sl. No.	Category of Course	Course Code	Course Title	Pre-requisite	Periods/Week			Theory		Practical		Total Marks	Credit
					L	T	P	End Sem Exam	Progressive Assignment	End Sem Exam	Progressive Assignment		
1	Programme Core	ITPC201/TH1	Programming in C		3	0	0	70	30	0	0	100	3
2		ITPC203/TH2	Programming with Python		3	0	0	70	30	0	0	100	3
3		ITPC205/TH3	Data structures and algorithms		3	0	0	70	30	0	0	100	3
4		ITPC207/TH4	Digital Electronics and Computer Organization		3	0	0	70	30	0	0	100	3
5		ITPC209/TH5	Computer Graphics and Multimedia		3	0	0	70	30	0	0	100	3
6		ITPC211/PR1	C Programming Lab		0	0	4	0	0	15	35	50	2
7		ITPC213/PR2	Programming with Python Lab		0	0	4	0	0	15	35	50	2
8		ITPC215/PR3	Data Structures and Algorithms Lab		0	0	4	0	0	15	35	50	2
9		ITPC217/PR4	Multimedia Lab		0	0	4	0	0	15	35	50	2
10	Summer Internship	SI201	Summer Internship-I(4weeks) after II SEM		0	0	0	0	0	15	35	50	2
					15	0	16	350	150	75	175	750	25

SEMESTER III
Detailed Curriculum

TH:1- PROGRAMMING IN C

L	T	P	Total Marks=100	Course Code: ITPC201/TH1
3	0	0		Theory Assessment
Total Contact Hrs:45				End Term Exam:70
Theory:45				Progressive Assessment:30
Tutorial:0				Category of Course: PC
Practical:0				
Pre-requisite: NIL				
Credit:3				

RATIONALE: This course has been designed by keeping in mind to have a basic knowledge about the Programming Concepts including various control structures like branching and looping statements, modular programming using user defined functions and handling pointers. The high-level language that has been used for instruction is C.C Programming language, which is a versatile language and knowledge, and skill acquired may be used in system programming.

LEARNING OUTCOMES:

After competing this course, student will be able to:

- Explain the basic concept about the programming
- Demonstrate programming using expression and statements
- Illustrate branching and looping statement using C language
- Declare and use array in the programs
- Exhibit use of modular programming using user defined functions
- Demonstrate addressing memory and multiple data type using pointers and structures.

DETAILED COURSE CONTENT:

UNIT NO.	TOPIC/SUB-TOPIC	ALLOTTED TIME (HRS.)
I	Introductory Concept • Types of Programming Languages • Introduction to C	2
II	C Fundamentals • The 'C' Character Set • Identifiers and Keywords • Data Types • Constants • Variables and Arrays • Declarations • Expressions • Statements • Symbolic Constants	3

III	Operators and Expressions • Arithmetic Operators • Unary operators • Relational and Logical Operators • Assignment Operators • The Conditional Operator • Library Functions	4
IV	Data Input and Output • Preliminaries • Single Character Input-The getchar Function • Single Character Output-The putchar Function • Entering Input data-The scanf function • Writing Output Data-The printf function • gets and puts Function	4
V	Preparing and running a complete C Program • Planning a C Program • Writing a C Program • Entering the Program into the Computer • Compiling and Executing the Program • Error Diagnostics • Debugging Techniques	4
VI	Control Statements • Preliminaries • Branching: The if statement, if-else statement, elseif ladder • Looping: while, do-while & for statements • Nested Control Structures • The switch statement • The break statement • The continue statement • The Comma operator • The goto statement	8
VII	User Defined Functions • Concepts of a Function • Accessing a Function • Function Prototypes • Passing parameters in Functions	5
VIII	Arrays • Defining an Array • Processing an Array • Multidimensional Arrays • Arrays and Strings & its Library functions	5

IX	Pointers • Fundamentals of Pointers • Pointer Declaration • Operation on pointers • Passing pointer to a function • Pointers and one-dimensional array • Pointers and two -dimensional array	5
X	Structures and Unions • Defining a Structure • Processing a Structure • Structure and Pointers • Arrays of Structure • Unions	5
	Total	45

REFERENCE BOOKS:

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

TH:2- PROGRAMMING WITH PYTHON

L	T	P	Total Marks=100	Course Code: ITPC203/TH2
3	0	0		Theory Assessment
Total Contact Hrs:45				End Term Exam:70
Theory:45				Progressive Assessment:30
Tutorial:0				Category of Course: PC
Practical:0				

RATIONALE:

Python is a versatile and beginner-friendly scripting language, renowned for its simplicity and readability. It enables rapid development of applications in fields like web development, data analysis, automation, and artificial intelligence. Learning Python equips students with a powerful toolset to solve complex problems and adapt to evolving technology demands.

LEARNING OUTCOMES:

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

- Define Python's core syntax, data types, and key concepts of object-oriented programming.
- Explain how control structures and data structures function in Python.
- Implement Python programs using file handling, modules, and libraries like NumPy, Pandas, and Matplotlib for data analysis and visualization.
- Analyze Python scripts to identify and resolve logical or syntactic errors and optimize code using advanced techniques like recursion and lambda functions.
- Develop a real-world mini-project by integrating Python concepts such as OOP, libraries, and automation tools for practical problem-solving.

DETAILED COURSE CONTENTS:

Unit No.	Topic/Sub-Topic	Allotted Time (Hours)
I	Introduction to Python: Overview of Python: Features and Applications Setting Up the Python Environment (Python Installation, IDEs), Python Syntax: Variables, Data Types, and Operators Writing, Executing, and Debugging Python Scripts	8
II	Control Structures and Functions: Conditional Statements: if, else, elif Loops: for, while, and Nested Loops Functions: Defining, Calling, and Scope of Variables Introduction to Lambda Functions and Recursion	8
III	Data Structures in Python: Lists, Tuples, Sets, and Dictionaries: Operations and Applications, List Comprehensions and Dictionary Comprehensions Working with Strings: Methods and Manipulation, Introduction to Python's Collections Module	8

IV	File Handling and Modules: File Operations, Reading, Writing, and Appending Files Working with CSV and JSON Files, Python Modules: Built-In Modules (e.g., math, os, datetime) Creating and Using Custom Modules	7
V	Object-Oriented Programming (OOP) in Python: Understanding Classes and Objects Concepts of Encapsulation, Inheritance, and Polymorphism, Working with Magic Methods and Operator Overloading, Exception Handling in Python	7
VI	Advanced Python and Applications: Introduction to Libraries: NumPy, Pandas, Matplotlib Basics of Web Scraping: Using requests and BeautifulSoup, Scripting for Automation: Working with os and shutil Modules, Mini-Project: Developing a Python Script for a Real-World Problem	7

REFERENCES:

1.	John M. Zelle, Python Programming: An Introduction to Computer Science, 2nd Edition, Franklin, Beedle & Associates Inc., Portland, 2010.
2.	Al Sweigart, Automate the Boring Stuff with Python, 2nd Edition, No Starch Press, San Francisco, 2019.
3.	Reema Thareja, Python Programming Using Problem Solving Approach, Oxford University Press, New Delhi, 2017.
4.	Sheetal Taneja and Naveen Kumar, Python Programming: A Modular Approach with Graphics, Database, Mobile, and Web Applications, Pearson India, New Delhi, 2018.
5.	R. Nageswara Rao, Core Python Programming, Dreamtech Press, New Delhi, 2018.
6.	A.K. Sharma, Python for Beginners: With Hands-On Examples, BPB Publications, New Delhi, 2020.

TH:3- DATA STRUCTURES AND ALGORITHMS

L	T	P	Total Marks=100	Course Code: ITPC205/TH3
3	0	0		Theory Assessment
Total Contact Hrs:45				EndTermExam:70
Theory:45				ProgressiveAssessment:30
Tutorial:0				
Practical:0				
Pre-requisite: NIL				
Credit:3				Category of Course: PC

RATIONALE:

Data Structures and Algorithms (DSA) form the backbone of computer science and software engineering. They enable efficient data storage, retrieval, and manipulation, while algorithms provide systematic methods to solve complex computational problems. Mastering DSA equips students with problem-solving skills essential for developing optimized, scalable, and robust software systems.

LEARNING OUTCOMES:

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

- Define the basic terminologies and classifications of data structures.
- Explain the representation and operations of linear data structures like stacks, queues, and linked lists.
- Implement algorithms for sorting and searching using appropriate data structures.
- Analyze the time and space complexity of data structure operations and algorithm paradigms like dynamic programming and backtracking.
- Design solutions using advanced data structures for real-world applications, such as shortest path problems or flow-based algorithms.

DETAILED COURSE CONTENTS:

Unit No.	Topic/Sub-Topic	Allotted Time (Hours)
I	Introduction to Data Structures: Basic Terminology, Classification of Data Structure, Operations on Data Structure, Asymptotic and worst-case analysis of algorithms.	6
II	Linear Data Structures: Stacks-Introduction to Stacks, Array Representation of Stacks, Operations on a Stack, Applications of Stacks-Infix-to-Postfix Transformation, evaluating Postfix Expressions. Queues: Introduction to Queues, Array Representation of Queues, Operations on a Queue, Types of Queues-DeQueue, Circular Queue, Applications of Queues-Round Robin Algorithm.	7
III	Linked Lists: Singly Linked List, Representation in Memory, Operations on a Single Linked List, Circular Linked Lists, Doubly Linked Lists, Linked List Representation and Operations of Stack, Linked List Representation and Operations of Queue.	6

IV	Non-Linear Data Structures: Trees-Basic Terminologies, Definition and Concepts of Binary Trees, Representations of a Binary Tree using Arrays and Linked Lists, Operations on a Binary Tree-Insertion, Deletion, Traversals, Types of Binary Trees. GRAPHS: Graph Terminologies, Representation of Graphs- Set, Linked, Matrix, Graph Traversals	7
V	Algorithm Paradigms: Greedy, Divide and Conquer, Branch and Bound, Dynamic Programming and Backtracking.	5
VI	Sorting: The sorting problem. Bubble sort, Selection sort, Insertion sort, Merge sort, Quicksort, Searching: Symbol Tables, Binary Search Trees, Balanced Search Trees. Hash Tables.	8
VII	Graphs: Definition of a directed and undirected graph. Paths, Cycles, spanning trees, Directed Acyclic Graphs. Topological Sorting. Minimum Spanning Tree algorithms, Shortest Path algorithms: Dijkstra's algorithm. Flow-based algorithms.	6

REFERENCES:

1.	Narasimha Karumanchi, Data Structures And Algorithms Made Easy: Data Structures And Algorithmic Puzzles, 2nd Edition, CareerMonk Publications, India, 2011.
2.	Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, Introduction to Algorithms, 3rd Edition, MIT Press, Cambridge, 2009.
3.	Reema Thareja, Data Structures Using C, 2nd Edition, Oxford University Press India, New Delhi, 2014.
4.	Ellis Horowitz, Sartaj Sahni, and Susan Anderson-Freed, Fundamentals of Data Structure in C, 2nd Edition, University Press, India, 2008.
5.	Gajendra Sharma, Design & Analysis of Algorithms, 1st Edition, Khanna Publishing House, New Delhi, 2016.
6.	Robert Sedgewick and Kevin Wayne, Algorithms, 4th Edition, Pearson Education, United States, 2011.

TH:4- DIGITAL ELECTRONICS AND COMPUTER ORGANISATION

L	T	P	Total Marks=100	Course Code: ITPC207/TH4
3	0	0		Theory Assessment
TotalContactHrs:45				EndTermExam:70
Theory:45				ProgressiveAssessment:30
Tutorial:0				Category of Course: PC
Practical:0				
Pre-requisite: NIL				
Credit:3				

RATIONALE:

Digital Electronics and Computer Organization form the foundation of modern computing, enabling the design and functioning of digital systems. They cover essential concepts like logic circuits, memory, and system architecture, bridging hardware and software integration. Learning these topics equips students with the skills to understand and develop efficient, reliable computing systems.

LEARNING OUTCOMES:

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

- Define key concepts of digital electronics, including number systems, Boolean algebra, and logic gates.
- Explain the principles behind combinational and sequential circuits, such as multiplexers, flip-flops, and counters, and their applications.
- Implement simplified logic circuits using Karnaugh Maps and Boolean algebra to solve real-world digital design problems.
- Analyze the instruction cycle, memory organization, and processor architecture to evaluate system performance and identify bottlenecks.
- Design to simulate a basic CPU operation or create a functional digital circuit using the concepts of digital electronics and computer organization.

DETAILED COURSE CONTENTS:

Unit No.	Topic/Sub-Topic	Allotted Time (Hours)
I	Introduction to Digital Electronics: Difference Between Analog and Digital Signals Number Systems: Binary, Octal, Decimal, and Hexadecimal Conversion Between Number Systems, Binary Arithmetic Boolean Algebra: Basic Operations, Laws, and Simplification	8

II	Logic Gates and Circuits: Logic Gates: AND, OR, NOT, NAND, NOR, XOR, XNOR, Design and Simplification of Logic Circuits Using Boolean Algebra, Karnaugh Maps (K-Maps) for Simplification Practical Applications of Logic Gates in Real-World Circuits	7
III	Combinational and Sequential Circuits: Combinational Circuits: Multiplexers, Demultiplexers, Encoders, and Decoders, Sequential Circuits: Flip-Flops (SR, JK, D, T) and Their Applications Counters: Synchronous and Asynchronous Counters Registers and Shift Registers: Types and Uses	8
IV	Fundamentals of Computer Organization: Basic Structure of a Computer: CPU, Memory, Input/Output Devices Instruction Cycle: Fetch, Decode, Execute Memory Organization: Types of Memory (RAM, ROM, Cache, Virtual Memory), Introduction to Buses: Address Bus, Data Bus, and Control Bus	8
V	Processor Architecture and Control: Unit Introduction to Microprocessors and Microcontrollers, Basics of Arithmetic Logic Unit (ALU) and Control Unit, Instruction Set Architecture (ISA): RISC vs CISC Pipelining and Performance Optimization in Processors	7
VI	Input/Output Systems and Advanced Topics: I/O Devices and Interfaces: Keyboard, Mouse, Printers, and Storage Devices Interrupts and DMA (Direct Memory Access), Overview of Modern Trends: Multicore Processors, GPUs, and Embedded Systems, Mini- Project: Design a Simple Digital Circuit or Simulate a Basic CPU Operation	7

REFERENCES:

1.	M. Morris Mano, Digital Design, 5th Edition, Pearson Education, India, 2013.
2.	David A. Patterson and John L. Hennessy, Computer Organization and Design: The Hardware/Software Interface, 5th Edition, Morgan Kaufmann, San Francisco, 2013.
3.	V. Rajaraman and T. Radhakrishnan, Digital Logic and Computer Organization, 1 st Edition, PHI Learning, New Delhi, 2006.
4.	R. P. Jain, Modern Digital Electronics, 4th Edition, Tata McGraw-Hill Education, New Delhi, 2009.
5.	Dhananjay M. Dhamdhare, Computer Organization and Assembly Language Programming, 1st Edition, Tata McGraw-Hill Education, New Delhi, 1987.

TH:5- COMPUTER GRAPHICS & MULTIMEDIA

L	T	P	Total Marks=100	Course Code: ITPC209/TH5
3	0	0		
Total Contact Hrs: 45				Theory Assessment
Theory:45				End TermExam:70
Tutorial:0				Progressive Assessment:30
Practical:0				Category of Course: PC
Pre-requisite: NIL				
Credit:3				

RATIONALE: This course has been designed to provide the student the knowledge and necessary skill related to computer graphics. It deals with various components like coordinate system, different algorithm for creation of object shapes. It also provide an knowledge and skill set related to various transformation such as translation, rotation and scaling. It also provides knowledge about the various hardware components required for computer graphics. In laboratory the students will be developing their skill set using various multimedia tools.

LEARNING OUTCOMES: After completing this course, student will be able to:

- Define concepts of Computer Graphics and Multimedia
- Identify various hardware requirement for operations pertaining to graphics
- Implement rasterization and transformation to any graphical entity
- Explain solid modelling with curves and surfaces
- Identify algorithms for various graphical representation
- Understand rendering and methods of animation

DETAILED COURSE CONTENT:

UNIT NO.	TOPIC/SUB-TOPIC	ALLOTTED TIME (HRS.)
1	Introduction to Computer Graphics: Introduction, Image Processing and Picture analysis, Conceptual frame work for interactive graphics, Classification	5
2	Hardware: Various display devices, Video controller, Random-scan display processor, Image scanners, Interaction hardware	4
3	Raster Graphics Techniques: Interaction handling, Raster graphics features, Line drawing algorithms, Circle drawing algorithms, Scan conversion, Polygon filling, Pattern filling, Halftoning, Clipping techniques	8
4	Geometric Transformation and Viewing: 2D and 3D transformation, Representation and composition, 3D viewing	3

5	User Interfacing: Interaction handling models, Window management Input/Output handling, Tool kits	5
6	Curves & Surfaces and Solid Modelling: Polygon meshes, Parametric cubic curves, Quadric surfaces, Bezier and B-spline curves, Representing solids: sweep representation, boundary representation, Spatial partitioning	7
7	Visibility: Hidden line and Hidden surfaces, Floating horizon algorithm Robert's algorithm, Z-buffer List priority algorithms	5
8	Rendering: Illumination models, Shadows Shading Transparency	4
9.	Animation: Conversion & Computer Aided animation, Rules & Technology	4
TOTAL		45

REFERENCEBOOKS:

- Computer Graphics by Donald Heran and M. Pauline Baker, Pearson Education Ltd.,
- Fundamentals of Computer Graphics & Multimedia by D.P. Mukherjee, Printice Hall India Pvt. Ltd.
- Multimedia–An Introduction by John Villamil-Casanova and Louis Molina, Pearson Education Ltd.,
- Multimedia– Sound & Video by Jose Lozano, Pearson Education Ltd.,
- Multimedia Graphics by John Villamil & Leony Fernandez, Pearson Education Ltd.,
- Principles of Multimedia by Ranjan Parekh, Mc Graw Hill.

PR:1- C PROGRAMMING LAB

L	T	P	Total Marks=50	Course Code: ITPC211/PR1
0	0	4		Theory Assessment
Total Contact Hrs:60				End Term Exam:15
Theory:0				Progressive Assessment:35
Tutorial:0				Category of Course: PC
Practical:60				
Pre-requisite: NIL				
Credit:2				

RATIONALE: This course has been designed keeping in mind to develop the skill related to programming aspect. In this course the students will verify the basic concepts that are delivered in theory classes of Programming in C.

LEARNING OUTCOMES: After completing this course, student will be able to:

- Demonstrate the fundamental concepts related to coding,
- Demonstrate programming using expression and statements,
- Exhibit branching and looping statement using C language,
- Write programs using array,
- Exhibit modular programming using user defined functions.

DETAILED COURSE CONTENT: *Topic for Practice*

1	Familiarization with 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 (if, else-if nested else if ladder)
5	Programs using loop statements (While, Do-while & For Loop Statement)
6	Programs to demonstrate applications of n dimensional arrays
7	Programs to demonstrate use of string manipulation functions
8	Programs to demonstrate parameter passing mechanism
9	Programs to demonstrate recursion
10	Programs to demonstrate use of pointers
11	Programs to demonstrate command line arguments
12	Programs to demonstrate dynamic memo

REFERENCE BOOK:

1.Programming in ANSIC, E. Balagurusamy, TataMcGraw-Hill

LIST OF EQUIPMENT:

Hardware: Standalone PC(for detail, please refer Annex-I)

Software: C Compiler

PR:2- PROGRAMMING WITH PYTHON LAB

L	T	P	Total Marks=50	Course Code: ITPC213/PR2
0	0	4		Theory Assessment
Total ContactHrs:60				End Term Exam:15
Theory:0				Progressive Assessment:35
Tutorial:0				Category of Course: PC
Practical:60				
Pre-requisite: NIL				
Credit:2				

RATIONALE: Python is a versatile and beginner-friendly scripting language, renowned for its simplicity and readability. It enables rapid development of applications in fields like web development, data analysis, automation, and artificial intelligence. Learning Python equips students with a powerful toolset to solve complex problems and adapt to evolving technology demands.

LEARNING OUTCOMES:

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

- Describe Python's features, applications, and fundamental concepts such as control structures, functions, and recursion.
- Develop Python programs using built-in data structures, string manipulations, file handling, and modules for practical problem-solving.
- Implement object-oriented programming concepts, including classes, objects, inheritance, and exception handling, to structure programs effectively.
- Evaluate advanced Python libraries such as NumPy, Pandas, and Matplotlib to process, analyze, and visualize data for specific applications.
- Design a mini-project to integrate Python programming concepts, demonstrating real-world applications and problem-solving capabilities.

DETAILED COURSE CONTENTS:

Unit No.	Topic/Sub-Topic	Allotted Time (Hours)
I	Introduction to Python: • Install Python and set up an IDE (e.g., PyCharm, VS Code, Jupyter, Spyder) • Write simple Python scripts to demonstrate variable declarations • Data types, and operators • Debug Python scripts to identify and fix errors.	10
II	Control Structures and Functions: • Implement conditional statements (if, else, elif) in real-life scenarios	10

	• Write programs using loops (for, while, and nested loops) to solve repetitive tasks • Define custom functions, including examples of recursion, use lambda functions for inline operations.	
III	Data Structures in Python: • Perform CRUD operations on lists, tuples, sets, and dictionaries, use list comprehensions to filter and transform data • Perform string manipulations using built-in methods • Introduce Python's collections module with practical examples.	10
IV	File Handling and Modules: • Write programs to read, write, and append text files • Work with CSV files using Python's csv module • Read and write JSON files to store structured data • Explore built-in modules like os, math, and datetime, create and import custom modules.	10
V	Object-Oriented Programming (OOP) in Python: • Define classes and create objects with attributes and methods • Implement encapsulation, inheritance, and polymorphism • Work with magic methods (e.g., <code>__init__</code>, <code>__str__</code>) and operator overloading, • Write programs to handle exceptions using try, except, and finally.	10
VI	Advanced Python and Applications: • Use NumPy for numerical operations and Pandas for data analysis • Mini-Project: Develop a Python script to solve a real-world problem (e.g., a data analysis script, a file organizer, or a basic web scraper).	10

REFERENCES:

1.	Yashavant Kanetkar, Let Us Python, 2nd Edition, BPB Publications, India, 2020.
2.	Reema Thareja, Python Programming: Using Problem Solving Approach, 1st Edition, Oxford University Press, India, 2017.
3.	R. Nageswara Rao, Core Python Programming, 1st Edition, Dreamtech Press, India, 2018.
4.	Satish Jain & Shashi Singh, Python Programming for Beginners, 1st Edition, BPB Publications, India, 2019.
5.	A. K. Sinha, Python Programming: A Modular Approach with Graphics, Database, Mobile and Web Applications, 1st Edition, BPB Publications, India, 2021.
6.	Yashavant Kanetkar, Let Us Python, 2nd Edition, BPB Publications, India, 2020.

PR:3- DATA STRUCTURES AND ALGORITHMS LAB

L	T	P	Total Marks=50	Course Code: ITPC215/PR3
0	0	4		
Total Contact Hrs: 60				Theory Assessment
Theory:0				EndTermExam:15
Tutorial:0				ProgressiveAssessment:35
Practical:60				
Pre-requisite: NIL				
Credit:2				Category of Course: PC

RATIONALE:

Data Structures and Algorithms (DSA) form the backbone of computer science and software engineering. They enable efficient data storage, retrieval, and manipulation, while algorithms provide systematic methods to solve complex computational problems. Mastering DSA equips students with problem-solving skills essential for developing optimized, scalable, and robust software systems.

LEARNING OUTCOMES:

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

- Explain basic terminologies and operations on data structures.
- Perform asymptotic and worst-case analysis of algorithms.
- Implement linear data structures.
- Apply trees and graphs to solve problems.
- Implement sorting and searching algorithms.

DETAILED COURSE CONTENTS:

Unit No.	Topic/Sub-Topic	Allotted Time (Hours)
I	Introduction to Data Structures: • Write a program to analyze and compare the time complexity of basic operations (e.g., searching, insertion) on arrays and linked lists.	2
II	Linear Data Structures: • Implement stack operations (push, pop, peek) using arrays and linked lists • Develop programs for applications of stacks (e.g., infix-to-postfix conversion and postfix evaluation) • Implement queue operations (enqueue, dequeue) using arrays and linked lists • Write programs for types of queues: circular queues and dequeue	12

III	Linked Lists: - Implement singly linked list operations (insertion, deletion, traversal). - Write programs to create and manipulate circular and doubly linked lists - Implement stack and queue operations using linked lists.	10
IV	Non-Linear Data Structures: - Implement binary tree operations (insertion, deletion, traversal) - Develop programs for types of binary trees (binary search tree, AVL tree) - Implement graph representations (adjacency list, adjacency matrix) - Basic graph traversals (BFS, DFS).	16
V	Sorting: Implement sorting algorithms: bubble sort, selection sort, insertion sort, merge sort, quicksort.	10
VI	Searching: - Write programs for searching using binary search trees (BST) and hash tables - Implement symbol table operations using balanced search trees.	10

REFERENCES:

1.	Seymour Lipschutz , Schaum's Outlines - Data Structures With C, Tata-MacGraw-Hill
2.	Sahni Sartaj, Data Structures, Algorithms and Applications in C++, 2nd Edition, Universities Press, India, 2005.
3.	Yashavant Kanetkar, Data Structures Through C, 2nd Edition, BPB Publications, India, 2008.
4.	Reema Thareja, Data Structures Using C, 2nd Edition, Oxford University Press, India, 2014.
5.	Debasis Samanta, Classic Data Structures, 2nd Edition, PHI Learning Pvt. Ltd., India, 2009.
6.	A. M. Tenenbaum, Data Structures Using C and C++, 2nd Edition, Pearson Education, India, 2003.
7.	Sahni Sartaj, Data Structures, Algorithms and Applications in C++, 2nd Edition, Universities Press, India, 2005.

PR:4- MULTIMEDIA LAB

L	T	P	Total Marks=50	Course Code: ITPC217/PR4
0	0	4		Theory Assessment
Total ContactHrs:60				End Term Exam:15
Theory:0				Progressive Assessment:35
Tutorial:0				Category of Course: PC
Practical:60				
Pre-requisite: NIL				
Credit:2				

RATIONALE: This course has been designed and developed to augment the skill set of the student in the field of multimedia application development using various tools and applications. This includes basic skill development in image and graphics, video editing, and basic multimedia application development in the field of education and other related domain.

LEARNING OUTCOMES:

After completing this course, student will be able to:

- Demonstrate skills related to image and graphics development
- Demonstrate skills related to animation and video editing
- Demonstrate skills related to multimedia application development in the field of education and other related domain

DETAILED COURSE CONTENT:

Unit No.	Topic/Sub-Topic	Allotted Time (Hours)
1	Adobe Photoshop: Scanning image, creating new images, changing foreground and background colors, Creating and using paths, Editing and retouching Duplicating images, Layers-linking with layers, Grouping an image, Rubber stamp and pattern stamp tool, Painting-paint brush tool, air-brush tool, pencil tool, eraser tool, gradient tools Photo shop filters	15
2	Sound Forge: Sound recording and editing through sound forgeXP: The main screen, the data window, opening an existing file-playing a sound file, playing a section of a file, copying data to a new file, saving a file, Simple editing, Advanced editing, editing sound formats, applying sound processing	15

	functions Recording sound using sound forge.	
3	Adobe Premiere: Creating desktop video with Adobe Premiere, Creating on Adobe Premiere movie, Starting a new project importing clips, assembling the clipping construction window, previewing the movie, changing duration of a cell, creating a transition, adding other clips and transitions, Applying filters to a clip, Changing the time unit in the construction window, Using preview command to preview the transition and filter effects, Adding sound to movie, Connecting and capturing source video through broad way cord, Editing and compressing the video	15
4	Author ware: Introduction-system requirements, installing, general features, Knowledge objects-introduction to knowledge objects, choosing a knowledge object, adding a knowledge object file, author ware knowledge objects, Authoring basics - icon based authoring what each icon does the tool bar, working with icons on the flow line, authoring-step by step, distribution requirements, packaging an AW piece, packaging and AW piece for the web, Creating interactions - components of an interaction, How an interaction works, tracing the flow through an interaction, setting up an interaction step by step, Directing the flow - Decision structure, frameworks, navigation structures -step by step, Transitions, Positioning and motion - using transition for special effects, positioning objects using the motion icon, making objects move step by step	15

REFERENCEBOOKS:

1. Reference Manual of Adobe Master Collection CS5.5
2. Manuals for Sound Forge, Adobe Premiere, Adobe Photoshop, Author ware Attain, Director, Tool book Instructor
3. Multimedia–Sound C Video–by Jose Lozano–Prentice Hall
4. Multimedia Graphics–by John Villamil C Leony Fernandez, Elias–Prentice Hall

LIST OF EQUIPMENT:

1. Hardware : Multimedia PC (for detail, please refer Annex – I)
2. Software : Abobe Master Collection CS5.5 or Latest Edition

SUMMER INTERNSHIP-I

L	T	P	Total Marks=50	Course Code: SI201
0	0	0		Theory Assessment
Total ContactHrs:0				End Term Exam:15
Theory:0				Progressive Assessment:35
Tutorial:0				Category of Course: SI
Practical:0				
Pre-requisite: NIL				
Credit:2				

RATIONALE: Summer Internship-I is to offer a structured and practical learning experience that prepares individuals for their future careers, helps them make in form career choices, and equips them with the skills and knowledge necessary to succeed in their chosen field. This course provides opportunities to students for hands-on industry experience.

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

- Apply theoretical knowledge gained in their academic course work to real-world situations.
- Develop and refine specific skills relevant to the field.
- Gains hands-on experience in a professional network by interacting with mentors And industry professionals.
- Learn to manage their time effectively.
- Clarify career goals.

DETAILED COURSE CONTENTS SUGGESTED ACTIVITIES:

I Orientation: ☐ Introduction to the organization's mission, values, and culture. ☐ Familiarization with work place policies, procedures, and safety guide lines. ☐ Orientation to the team and organizational structure.
II Project-Based Learning: ☐ Description of the main project or tasks the intern will be working on during internship. ☐ Detailed project goals and objectives. ☐ Training and guidance on project-specific tools, technologies, or methodologies. ☐
III Technical and Skill Development:

☐ Training sessions or workshops to enhance technical skills relevant to the internship role (e.g., programming languages, software tools, laboratory techniques). ☐ Soft skills development, including communication, teamwork, problem solving, and time management
IV Mentorship and Supervision: ☐ Regular meetings with a designated mentor or supervisor for guidance, feedback, and support. ☐ Mentorship objectives and expectations.
V Professional Development: ☐ Sessions on professional etiquette, networking, and ☐ building a personal brand Resume writing and interview preparation workshops.
VI Industry and Field-Specific Knowledge: ☐ Lectures, seminars, or presentations on industry trends, best practices, and emerging technologies. ☐ Guest speakers from the field to share insights and experiences.
VII Reporting and Documentation: ☐ Training on how to document project progress, results, and findings. ☐ Practice in creating reports, presentations, or other deliverables.
VIII Ethics and Professionalism: ☐ Discussions on ethical considerations within the field. ☐ Scenarios and case studies related to ethical decision-making
IX Feedback and Evaluation: ☐ Regular performance evaluations and feedback sessions. ☐ Self- assessment and goal-setting exercises.
X Networking and Industry Exposure: ☐ Opportunities to attend industry conferences, webinars, or networking events. ☐ Encouragement to connect with professionals in the field.

NOTE As per AICTE guidelines, in Summer Internship-I, students are required to be involved in Inter/ Intra Institutional Activities viz;

Training with higher Institutions;

- Soft skill training organized by Training and Placement Cell of the respective institutions;
- Contribution at incubation/innovation/entrepreneurship cell of the institute; participation in conferences/ workshops/ competitions etc.;
- Learning at Departmental Lab/Tinkering Lab/Institutional workshop; Working for consultancy/ research project within the institutes and Participation in all the activities of Institute's Innovation Council for eg: IPR workshop/Leadership Talks/ Idea/ Design/ Innovation/ Business Completion/ Technical Expos etc.