GOVERNMENT POLYTECHNIC, NAGPUR.
(An Autonomous Institute of Govt. of Maharashtra)

COURSE CURRICULUM

PROGRAMME : **DIPLOMA IN CM/IT**

LEVEL NAME : **PROFFESSIONAL COURSES**

COURSE CODE : **CM403E** [$]

COURSE TITLE : **DATA STRUCTURES**

PREREQUISITE : **CM401E**

TEACHING SCHEME : **TH: 03; TU: 00; PR: 04(CLOCK HRs.)**

TOTAL CREDITS : **05** (1 TH/TU CREDIT = 1 CLOCK HR., 1 PR CREDIT =2 CLOCK HR.)

TH. TEE EXAM : **03 Hrs**

PR. TEE EXAM : **02 Hrs (External)**

PT. EXAM : **01 Hr**

❖ **RATIONALE:**

In the present era it is very essential to develop programs and organize data in such a way that it solves a complex problem efficiently. Data structure is such a tool, which aims in developing data organizing and programming skills.

❖ **COURSE OUTCOMES:**

After completing this course students will be able to–

1. Apply the features and concepts of data structures.
2. Select proper data structures as per the need of applications.
3. Design algorithms for various operations performed on different data structures.
4. Design the programs using different data structures.
5. Debug and execute the programs.
6. Develop efficient software using various data structures

1

❖ **COURSE DETAILS:**

A. THEORY :

Units	Specific Learning Outcomes (Cognitive Domain)	Topics and subtopics	Hrs.
1. Introduction to data	1. Define the related terms. 2. State the needs of data structure	1.1 Definition of data structure. Basic Terminology, Elementary Data Organization	02

❖ **COURSE DETAILS:**

A. **THEORY :**

Units	Specific Learning Outcomes (Cognitive Domain)	Topics and subtopics	Hrs.
1. Introduction to data structure	1. Define the related terms. 2. State the needs of data structure. 3. List different types of data structure. 4. Compare different data structure.	1.1 Definition of data structure. Basic Terminology, Elementary Data Organization Data structure operations 1.2 Need of data structure. 1.3 Types of data structure: Arrays, Stack, Queue, Linked List, Trees & Graphs	02
2. Stack & recursion	1. Define searching & sorting. 2. List types of searching & sorting techniques. 3. Define Stack. 4. State the operations on stack. 5. Describe the representation of stack in C. 6. Construct different types of polish expression. 7. Enlist the applications of stack. 8. Describe recursion in C. 9. Design & develop programs using recursion. 10. List the use of stack in recursion.	2.1 Searching – Linear Search & Binary Search and Sorting– Selection, Bubble, Insertion, Merge, Radix 2.2 Stack–Introduction Definition, basic terminologies, Examples 2.3 Operation on Stack :- Push and Pop operation, Algorithm for push and pop operation 2.4 Representation of Stack in C Array representation and Linked representation of stack 2.5 Polish Notation:- Infix, Postfix, Prefix Notation, Evaluation of Postfix and prefix expression, Conversion of infix to postfix Expression, Conversion of prefix to postfix expression 2.6 Application of stack 2.7 Recursion in C Definition, Programs based on Recursion 2.8 Implementation of Recursion using Stack 2.9 Implementation of Tower Of Hanoii problem using Stack	12
3. Queue	1. Describe the representation of Queue. 2. State the operations of queue. 3. Compare the circular and double ended queue. 4. Develop the programs based on queue .	3.1 Representation of Queue Array representation and linked representation of queue 3.2 Operation on Queue:- Special terminology for inserting and deleting operation ,Algorithm for inserting and deleting	10

2

		operation 3.3 Implementation of Queue in C 3.4 Circular Queue:- Insert and delete operation on circular queue, Advantages of Circular queue 3.5 Double Ended Queue	
4. Linked list	1. Describe the concept of linked list. 2. Describe the use of different functions for dynamic memory allocation. 3. Illustrate the implementation of singly linked list. 4. State the various operations performed on linked list. 5. Design the program for searching, Adding and deleting a node from linked list, counting number of nodes.	4.1 Introduction – Dynamic memory Allocation 4.2 Singly Linked List – Implementation 4.3 Searching operation 4.4 Creating Sorted Linked List 4.5 Adding & deleting a node from different position 4.6 Counting number of nodes in a Linked List 4.7 Reversing a Linked List	08
5. Tree	1. Describe the concepts of tree 2. Define the binary tree 3. State the binary tree Representation 4. Describe the binary tree traversal 5. Design Algorithms for different Tree Traversal 6. Define binary search tree 7. State the use of operations like searching ,adding, deleting node from BST 8. State the use of Heap sort	5.1 Introduction 5.2 Binary Tree 5.3 Binary tree representation 5.4 Binary tree traversal 5.5 Algorithms for In-order Traversal , Pre- order Traversal, Post-order Traversal 5.6 Binary tree with header node 5.7 Binary Search Tree 5.8 Searching a node in a BST 5.9 Adding a new node in BST 5.10 Deleting a node from BST 5.11 Heap Sort	10
6. Graph & their application	1. Describe the concepts of Graph 2. Define different terminology used in graph 3. State the use of Adjacency Matrix ,Pathmatrix Warshall's Algorithm 4. Describe different traversal method in graphs 5. Define the automatic list Management	6.1 Introduction 6.2 Different terminology used in Graph 6.3 Sequential representation of Graph 6.4 Adjacency Matrix , path Matrix 6.5 Warshall's Algorithm 6.6 Linked representation of Graph 6.7 Traversal used for Graph 6.8 Automatic List Management 6.9 Dynamic memory Management	6

	6. State and describe the dynamic memory management 7. Design Algorithms for different graph Traversal		
		Total Hrs	**48**

B. LIST OF PRACTICALS/LABORATORY EXPERIENCES/ASSIGNMENTS:

Practicals	Specific Learning Outcomes (Psychomotor Domain)	Units	Hrs.
1	Develop and execute a program for Linear search	Stack and Recursion	2
2	Develop and execute a program for Binary search		2
3	Develop and execute a program for Selection sort		2
4	Develop and execute a program for Bubble sort		2
5	Develop and execute a program for Insertion sort		2
6	Develop and execute a program for Merge sort		2
7	Develop and execute a program for Radix sort		2
8	Develop and execute a program to implement a stack		4
9	Develop and execute a program for Implementation of Recursion using Stack		4
10	Develop and execute a program to convert a expression into post fix expression.	Queue	4
11	Develop and execute a program for implementation of Queue		2
12	Develop and execute a program for Implementation of Tower Of Hanoii problem using Queue		4
13	Develop and execute a program to evaluate postfix expressions		4
14	Develop and execute a program to create sorted link list.	Linked list	4
15	Develop and execute a program for inserting and deleting a node from a sorted link list.		2
16	Develop and execute a program for counting number of nodes and reverting a link list		2
17	Develop and execute a program for searching a node in binary tree.	Trees	4
18	Develop and execute a program for inserting and deleting a node from Binary tree		4
19	Develop and execute a program for traversal of graph	Graphs & their application	4
20	Develop and execute a program using the dynamic memory allocation functions		4
		Skill Assessment	**4**
		Total Hrs	**64**

4

❖ **SPECIFICATION TABLE FOR THEORY PAPER:**

Unit No.	Units	Levels from Cognition Process Dimension			Total Marks
		R	**U**	**A**	
01	Introduction to Data structures	02(02)	04(00)	00(00)	06(02)
02	Stack and Recursion	00(02)	08(04)	06(00)	14(06)
03	Queue	02(00)	08(08)	06(00)	16(08)
04	Linked list	02(00)	04(04)	06(06)	12(10)
05	Trees	02(00)	04(08)	06(00)	12(08)
06	Graph & their application	02(00)	08(00)	00(06)	10 (06)
	Total	10(04)	36(24)	24 (12)	70 (40)

R – Remember   U – Understand   A – Analyze / Apply

❖ **QUESTION PAPER PROFILE FOR THEORY PAPER**

Q. No	Bit 1			Bit 2			Bit 3			Bit 4			Bit 5			Bit 6			option
	T	L	M	T	L	M	T	L	M	T	L	M	T	L	M	T	L	M	
01	1	R	2	3	R	2	4	R	2	5	R	2	6	R	2	1	R	2	5/7
	2	R	2																
02	1	U	4	2	U	4	2	U	4	3	U	4	2	U	4				3/5
03	3	U	4	3	U	4	4	U	4	3	U	4	4	U	4				3/5
04	5	U	4	6	U	4	6	U	4	5	U	4	5	U	4				3/5
05	2	A	6	3	A	6	4	A	6										2/3
06	4	A	6	5	A	6	6	A	6										2/3

T= Unit/Topic Number   L= Level of Question   M= Marks

R-Remember   U-Understand   A-Analyze/ Apply

❖ **ASSESSMENT AND EVALUATION SCHEME:**

			To Whom	Frequency	Max Marks	Min Marks	Evidence Collected	Course Outcomes
Direct Assessment Theory	**CA (Continuous Assessment)**	Progressive Test (PT)	Students	Two PT (average of two tests will be computed)	20	--	Test Answer Sheets	1, 2, 3
		Assignments		Continuous	10	--	Assignment Book / Sheet	1, 2, 3
	TEE (Term End Examination)	End Exam	Students	End Of the Course	70	28	Theory Answer Sheets	1, 2, 3
				Total	100	40		
Direct Assessment Practical	**CA (Continuous Assessment)**	Skill Assessment	Students	Continuous	20	--	Rubrics & Assessment Sheets	4,5,6
		Journal Writing		Continuous	05	--	Journal	4,5,6
				TOTAL	25	10		
	TEE (Term End Examination)	End Exam	Students	End Of the Course	50	20	Rubrics & Practical Answer Sheets	4,5,6
Indirect Assessment		Student Feedback on course	Students	After First Progressive Test	Student Feedback Form			1, 2, 3, 4,5,6
		End Of Course		End Of The Course	Questionnaires			

6

❖ **SCHEME OF PRACTICAL EVALUATION:**

S.N.	Description	Max. Marks
1	Writing the steps for algorithm and designing the program	20
2	Performance	20
3	Viva voce	10
	TOTAL	**50**

❖ **MAPPING COURSE OUTCOMES WITH PROGRAM OUTCOMES:**

1. Computer Engineering:-

Course Outcomes (COs)	Program Outcomes (POs)										PSO's	
	1	2	3	4	5	6	7	8	9	10	1	2
1	-	3	-	-	-	-	-	-	-	-	3	-
2	-	3	-	-	-	-	-	-	-	-	3	-
3	-	3	-	-	-	-	-	-	-	-	3	2
4	-	3	3	3	-	-	-	3	3	3	3	-
5	-	3	3	3	-	-	-	3	3	3	3	-
6	-	3	3	3	-	-	-	3	3	3	3	2

1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High)

2. Information Technology:-

Course Outcomes (COs)	Program Outcomes (POs)										PSO's	
	1	2	3	4	5	6	7	8	9	10	1	2
1	-	3	-	-	-	-	-	-	-	-	-	3
2	-	3	-	-	-	-	-	-	-	-	-	3
3	-	3	-	-	-	-	-	-	-	-	-	3
4	-	3	3	3	-	-	-	3	3	3	-	3
5	-	3	3	3	-	-	-	3	3	3	-	3
6	-	3	3	3	-	-	-	3	3	3	-	3

1: Slight (Low) 2: Moderate (Medium) 3: Substantial (High)