**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

* **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. Definition of data structure.   Basic Terminology, Elementary Data Organization Data structure operations   1. Need of data structure. 2. 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. | 1. Searching – Linear Search & Binary Search and Sorting– Selection, Bubble, Insertion, Merge, Radix 2. Stack–Introduction Definition, basic terminologies, Examples 3. Operation on Stack :-   Push and Pop operation,  Algorithm for push and pop  operation   1. Representation of Stack in C   Array representation and  Linked representation of stack   1. Polish Notation:-   Infix, Postfix, Prefix Notation,  Evaluation of Postfix and  prefix expression,  Conversion of infix to postfix  Expression, Conversion of  prefix to postfix expression   1. Application of stack 2. Recursion in C   Definition, Programs based on  Recursion   1. Implementation of Recursion using Stack 2. 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 . | 1. Representation of Queue   Array representation and linked representation of queue   1. Operation on Queue:-   Special terminology for  inserting and deleting  operation ,Algorithm for  inserting and deleting operation   1. Implementation of Queue in C 2. Circular Queue:-   Insert and delete operation on circular queue, Advantages of Circular queue  3.5 Double Ended Queue | 10 |
| 4. Linked list | 1. Describe the concept of linked list. 2. Describe the use of different functions for   dynamic memory allocation.   1. Illustrate the implementation of singly linked list. 2. State the various operations   performed on linked list.   1. Design the program for searching ,Adding and deleting a node from linked list, counting number of nodes. | 1. Introduction – Dynamic memory Allocation 2. Singly Linked List –   Implementation   1. Searching operation 2. Creating Sorted Linked List 3. Adding & deleting a node from different position 4. Counting number of nodes in a Linked List 5. 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 | 1. Introduction 2. Binary Tree 3. Binary tree representation 4. Binary tree traversal 5. Algorithms for In-order Traversal , Pre- order Traversal, Post-order Traversal 6. Binary tree with header node 7. Binary Search Tree 8. Searching a node in a BST 9. Adding a new node in BST 10. Deleting a node from BST 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. State and describe the dynamic  memory management  7. Design Algorithms for different  graph Traversal | 1. Introduction 2. Different terminology used in Graph 3. Sequential representation of Graph 4. Adjacency Matrix , path Matrix 5. Warshall’s Algorithm 6. Linked representation of Graph 7. Traversal used for Graph 8. Automatic List Management 9. Dynamic memory Management | 6 |
| **Total Hrs** | | | **48** |

**B. LIST OF PRACTICALS/LABORATORY EXPERIENCES/ASSIGNMENTS:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Practicals** | **Specific Learning Outcomes (Psychomotor Domain)** | **Units** | **Hrs.** |
| **1.** | Explore ? Study and overview of different data structures | Introduction to Data structures | **2** |
| **2** | Develop and execute a program for Linear search | Stack and Recursion | **2** |
| **3** | Develop and execute a program for Binary search | **2** |
| **4** | Develop and execute a program for Selection sort | **2** |
| **5** | Develop and execute a program for Bubble sort | **2** |
| **6** | Develop and execute a program for Insertion sort | **2** |
| **7** | Develop and execute a program for Merge sort | **2** |
| **8** | Develop and execute a program for Radix sort | **2** |
| **9** | Develop and execute a program to implement a stack | **4** |
| **10** | Develop and execute a program for Implementation of Recursion using Stack | **4** |
| **11** | Develop and execute a program to convert a expression into post fix expression. | Queue | **4** |
| **12** | Develop and execute a program for implementation of Queue | **2** |
| **13** | Develop and execute a program for Implementation of Tower Of Hanoii problem using Queue | **4** |
| **14** | Develop and execute a program to evaluate postfix expressions | **4** |
| **15** | Develop and execute a program to create sorted link list. | Linked list | **2** |
| **16** | Develop and execute a program for inserting and deleting a node from a sorted link list. | **2** |
| **17** | Develop and execute a program for counting number of nodes and reverting a link list | **2** |
| **18** | Develop and execute a program for searching a node in binary tree. | Trees | **4** |
| **19** | Develop and execute a program for inserting and deleting a node from Binary tree | **4** |
| **20** | Develop and execute a program for traversal of graph | Graphs & their application | **4** |
| **21** | Develop and execute a program using the dynamic memory allocation functions | **4** |
| **Skill Test** | | | **4** |
| **Total Hrs** | | | **64** |

* **SPECIFICATION TABLE FOR THEORY PAPER:**

R – Remember U – Understand A – Analyze / Apply

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **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)** |

* **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 | U | 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:**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **What** | | **To Whom** | **Frequency** | **Max Marks** | **Min Marks** | **Evidence Collected** | **Course Outcomes** |
| **Direct Assessment Theory** | **CA**  (Continuous Assessment) | PT | Students | Two PT (average of two tests will be computed) | 20 | -- | Test Answer Sheets | 1, 2, 3 |
| Class Room Assignments | Assignments | 10 | -- | Assignment Book | 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) | ST | Students | One skill test at end of term | 20 | -- | Practical Answer Sheets | 4,5,6 |
| Journal Writing | Assignments | 05 | -- | Journal | 4,5,6 |
|  | TOTAL | **25** | **10** |  |  |
| **TEE**  (Term End Examination) | End Exam | Students | End Of the Course | **50** | **20** | Practical Answer Sheets | 4,5,6 |
| **Indirect Assessment** | Student Feedback on course | | Students | After First PT | Student Feedback Form | | | 1, 2, 3, 4,5,6 |
| End Of Course | | End Of The Course | Questionnaires | | |

* **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:**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Outcomes**  **(COs)** | **Program Outcomes (POs)** | | | | | | | | | |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** |
| **1** | 3 | 3 | 2 | 3 | 2 | - | 2 | 2 | 2 | 3 |
| **2** | 3 | 3 | 2 | 3 | 2 |  | 2 | 2 | 2 | 3 |
| **3** | 3 | 3 | 2 | 3 | 2 | - | 2 | 2 | 2 | 3 |
| **4** | 3 | 3 | 2 | 3 | 2 | - | 2 | 3 | 2 | 3 |
| **5** | 3 | 3 | 2 | 3 | 2 | - | 2 | 3 | 2 | 3 |
| **6** | 3 | 3 | 2 | 3 | 2 | - | 2 | 3 | 2 | 3 |

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

* **REFERENCE & TEXT BOOKS:**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.N.** | **Title** | **Author, Publisher, Edition and Year Of publication** | **ISBN Number** |
| 1. | Data Structures in ‘C’ | Tanenbaum, Langsman, Augenstein Pearson Education, 2003 | 10: 8131702294  13: 978-8131702291 |
| 2. | [Introduction to Data Structure : With](http://www.abebooks.com/servlet/BookDetailsPL?bi=19552407011&searchurl=isbn%3D9780074624715%26sortby%3D17) applications | [Jean Paul Tremblay](http://www.amazon.in/Jean-Paul-Tremblay/e/B001HCX64M/ref=dp_byline_cont_book_1), [Paul Gordon Sorenson](http://www.amazon.in/s/ref=dp_byline_sr_book_2?ie=UTF8&field-author=Paul+Gordon+Sorenson&search-alias=stripbooks), International Student Edition ,1983 | 10:0070651574  13:9780070651579 |
| 3. | Data Structures with C | Seymour Lipschutz, Schaum's Outline  Series, 2010 | 10: 0070701989  13: 978-0070701984 |

* **E-REFERENCES:**

[**http://nptel.ac.in/courses/106102064/,**](http://nptel.ac.in/courses/106102064/,%20) **accessed on 02nd September 2016**

[**http://www.old.w3professors.com/Pages/Courses/Data-Structure/Algorithms/Data-Structure-Algorithm.html**](http://www.old.w3professors.com/Pages/Courses/Data-Structure/Algorithms/Data-Structure-Algorithm.html) **accessed on 02nd September 2016**

[**https://www.youtube.com/watch?v=92S4zgXN17o**](https://www.youtube.com/watch?v=92S4zgXN17o)**, accessed on 02ndSeptember 2016**

**LIST OF MAJOR EQUIPMENTS/INSTRUMENTS WITH SPECIFICATION**

1. Computer (Dual CORE and above) with Internet connection.
2. Network printer.
3. Office Suite
4. C compiler

* **LIST OF EXPERTS & TEACHERS WHO CONTRIBUTED FOR THIS CURRICULUM:**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.N.** | **Name** | **Designation** | **Institute / Industry** |
| 1 | Mr.S.P.Lambhade | HOD ,Computer Engineering | Government Polytechnic, Nagpur. |
| 2 | Dr.A.R.Mahajan | HOD ,Information Technology | Government Polytechnic, Nagpur. |
| 2 | Ms. D.M.Shirkey | Lecturer in Computer Engineering | Government Polytechnic, Nagpur. |
| 3. | Dr.A.R.Mahajan | HOD,Information Technology | Government Polytechnic, Nagpur. |
| 4 | Prof.V.A.Raje | Lecturer in Computer Engineering | Government Polytechnic, Nagpur. |
| 5 | Shri. Atul Upadhyay | CEO | Vista Computers , Ram Nagar, Nagpur |
| 6. | Shri. N. V. Chaudhari | Asst. Professor (CSE) | DBACEO, Wanadongri, Nagpur |
| 7 | Shri. Manoj Jethawa | HOD Computer Science | Shri Datta Meghe Polytechnic, Nagpur |

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
 (Member Secretary PBOS) (Chairman PBOS)**