

GOVERNMENT POLYTECHNIC, NAGPUR
(An Autonomous Institute of Government of Maharashtra)
(CURRICULUM DEVELOPMENT CELL)

CURRICULUM

PROGRAMME : DIPLOMA IN INFORMATION TECHNOLOGY

COURSE CODE : IT6407

NAME OF COURSE : DATA STRUCTURE USING “C”

NAME OF LEVEL : BASIC TECHNOLOGY COURSES

TOTAL CREDITS : 06

THEORY : 04

PRACTICAL : 02

EXAMINATION SCHEME :

MARKS	THEORY			PRACTICAL			TOTAL
	TERM EXAM	PROGRESSIVE TEST	TOTAL	PRACT - EXAM	TERM WORK	ORAL EXAM	
MAX	75	25	100	25	25	-	150
MIN	30	----	40	10	10	-	60

PREREQUISITE COURSE / COURSES TO BE COMPLETED : IT6202

TIME ALLOTTED FOR TERM EXAM : 3 HOURS

TIME ALLOTTED FOR PROGRESSIVE TEST : 1 HOUR

• 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.

• SKILLS :-

1. Understand the concept of Data structure.
2. Understand the features various Data Structure.
3. Write programs on various Data Structure.
4. Develop a project on various Data Structure.

• OBJECTIVES :-

After completing this course, the students will be able to

1. Write programs in ‘C ‘ using different types of data structures.
2. Understand concepts of arrays, pointers, link list, stacks, queues, trees, and graphs.
3. Use proper data structures for particular problem.
4. Develop efficient software using various data structures.

• **PROVISION OF LEARNING EXPERIENCES :**

A) THEORY :-

1. INTRODUCTION TO DATA STRUCTURE (H 02, M- 03)

What is Data Structure, Need of Data Structure, types of Data Structures:-
Array, Link List, Stacks, Queues, Trees and graphs

2. ARRAY OPERATIONS(SORTING): (H 03, M- 06)

Insertion, deletion, of element, Selection sort, Bubble Sort, Insertion Sort, Merge sort and Radix sort..

3..ARRAY OPERATIONS(SEARCHING): (H 05, M- 05)

Searching: Linear search, Binary search .Hashing.

4.LINKED LISTS: (H 07, M- 08)

Dynamic memory Allocation, singly link list, Representation of link list
creating a sorted link list, searching.

5. LINKED LIST OPERATIONS: (H 06, M- 08)

creating, traversing, inserting, deleting in sorted as well as
unsorted link list. Counting number of nodes, reverting a link list.

6. STACKS : (H 05, M- 06)

Stacks: Concept, representing stacks in 'C'. Polish Notations (Prefix, postfix), Infix,
Applications of stacks. .

7. RECURSION: (H 06, M- 07)

Recursive definitions and processes, Recursion in 'C', writing recursive programs
factorial, fibonacci, ackermann function, Tower of Hanoi, Implementation of recursive
procedures by means of stack.

8. QUEUES: (H 08, M- 08)

The queue and its sequential representation, concept of dequeues, priority queues,
Implementation of PUSH & POP operations, Tower of Hanoi, Evaluate postfix
expressions, Infix to postfix conversions.

9. BINARY TREES: (H 07, M- 06)

Introduction, Binary trees, Binary tree representation, Traversing binary tree, Traversal
algorithms using stacks, Header nodes.

10. THREADED TREES: (H 07, M- 09)

Binary search tree (BST), searching and inserting in BST, deleting from BST, Heap,
Heapsort, pathlengths: Huffman. Creating a binary tree, inorder, preorder and
postorder traversal of binary tree, deleting a node from binary tree

11. GRAPHS AND THEIR APPLICATIONS: (H 08, M- 09)

Introduction, Graph theory terminology, Sequential representation of graphs, Adjacency
matrix, Path matrix, Warshalls algorithm, shortest path, linked representation of graph,
Operations on graphs, traversing a graph and spanning forest. General trees, automatic lists
management, dynamic memory management.

B) LABORATORY /FIELD EXPERIENCES

• **List of Laboratory/Field experiences**

1. Create a programs using Selection sort and Bubble Sort.
2. Create a programs using Insertion Sort.
3. Create a programs using Linear search and Binary search .
4. Write a program to create sorted link list.
5. Create a programs for inserting and deleting a node from a sorted link list.
6. Create a programs for counting number of nodes and reverting a link list.
7. Create a programs to implement a stack.
8. Create a program to convert a expression into post fix expression.
9. Create a program for implementation of PUSH & POP operations.
10. Create a program to evaluate postfix expressions.
11. Create a programs for searching a node in binary tree.
12. Create a program for inserting and deleting a node from Binary tree .

C) INDUSTRIAL / FIELD TRAINING

NIL

D) INDUSTRIAL VISITS

NIL

E) STUDENT'S PROJECTS

• **INSTRUCTIONAL STRATEGIES TO BE ADOPTED**

- **Lecture method**
- **Improved Lecture Method**
- **Question - Answer Technique.**

• **LEARNING RESOURCES TO BE USED.**

- **PRINT ----- Text Books / Reference Books**
- **NON PRINT ----- Transparencies / Slides/ etc.**

REFERENCES :

1. Data Structures in 'C' -Tanenbaum, Langsman, Augenstein(PHI Publications)
2. Data Structures - Tremblie and Sorrenson(TMh Publications)
3. Data Structures - Lipschultz (Schaum Outline Series)