

# AMITY UNIVERSITY

JHARKHAND



## LAB MANUAL

**Course Title:** Data Structure

**Course Level:** UG

**Course Code:** CSIT124

**Program:** B.Tech(CSE)

**Semester:** III

**Faculty:**

PROF. PALLAB BANERJEE  
Assistant Professor  
Amity School of Engg. & Tech.

**Approved By:**

PROF. (DR.) AJIT KUMAR PANDEY  
Director  
Amity University Jharkhand

### **General instructions to students**

1. Students should be regular and come prepared for the lab practice.
2. In case a student misses a class, it is his/her responsibility to complete that missed experiment(s).
3. Students should bring the observation book, lab journal and lab manual.
4. Prescribed textbook and class notes can be kept ready for reference if required.
5. They should implement the given experiment individually.
6. Once the experiment(s) get executed, they should show the program and results to the instructors and copy the same in their observation book.
7. Questions for lab tests and exam need not necessarily be limited to the questions in the manual but could involve some variations and / or combinations of the questions.
8. All the students must maintain silence inside the lab.
9. All the students must carry their id card before entering the lab, and college uniform is strictly mandatory, otherwise students will not be permitted to sit inside the lab.
10. No food or beverage items are allowed inside the lab.
11. Keep your bags outside the lab.
12. Do not use cell phone inside the lab. (If anybody found using cell phone inside the lab, his/her mobile phone will be seized by the responsible authority.)
13. Shut down the system and arrange your chair before leaving the lab.
14. While using lab sign in the register with your name, enrollment number and branch to mark your attendance.
15. Do not plug any device without permission.
16. Do not use internet without permission.

**Note: Above mentioned instructions can be modified based on the context of the lab.**

### **Credit Unit**

<b>L (Lecture)</b>	<b>T</b>	<b>P/S (Practical/Studio)</b>	<b>SW/FW</b>	<b>Total Credit Units</b>
3	-	2	-	4

### **Lab/ Practical/ Studio Assessment**

	<b>Continuous Assessment/Internal Assessment</b>					<b>End Term Examination</b>	
<b>Components (Drop down)</b>	<b>Continuous Assessment</b>	<b>Mid Term</b>	<b>Viva</b>	<b>Lab Record</b>	<b>Attendance</b>	<b>Practical</b>	<b>Viva</b>
<b>Weightage (%)</b>	10	10	5	10	5	40	20

### **Course Objectives:**

The aim of this course is to

- Impart in-depth knowledge of data structure and its implementation in computer programs.
- Make students understand the concepts of linear and non linear data structure.
- Illustrate asymptotic notations and their usage.

**Pre-requisites:** Basics of C Programming.

**Student Learning Outcomes:**

After successful completion of this course, the student will be able to

- Apply advance C programming techniques such as pointers, dynamic memory allocation, structures to developing solutions for problems.
- Describe and implement abstract data types such as linked list, stack, queue and tree by using 'C' for static and dynamic implementations.
- Analyze, and evaluate appropriate abstract data types and algorithms to solve problems.

**Pedagogy for Course Delivery:**

Subject will be taught based on class room lectures and practical. Emphasis will be given on practical explaining use case scenario for various algorithms. Focus will be on student's involvement while imparting the course contents.

**List of Experiments:**

**#WEEK 1:**

1. Write a program to:
  - a. Find a factorial of a number.
  - b. To check a given number is prime or not.
  - c. To print a Fibonacci series of a given number.
  - d. Create a function for each subpoints and take a user input to enter the number.
2. Write a program to:
  - a. Enter a three-digit number and check whether it is Armstrong number or not
  - b. Enter a four-digit number and reverse it to check whether it is palindrome or not
  - c. Use separate function to implement these two operations and take a user input.

**#WEEK 2:**

1. Write a program to create an integer array and implement the following operations:
  - a. find the largest and smallest number of an array.
  - b. Search an element into an array
  - c. Find the sum of an array
  - d. Reverse the element of an array
2. Write a program to
  - a. enter your name and reverse it.
  - b. Count the number of vowels and consonants in the given string.
  - c. Use separate function to implement these two operations and take a user input.

**#WEEK 3:**

1. Write a program to enter two 3x3 matrices and implement:
  - a. Addition of two matrices
  - b. Multiplication of two matrices
  - c. Transpose of a matrix.
  - d. Use separate function to implement these two operations and take a user input.
  - e. Implement call by reference while creating and calling functions.

2. Write a program to create a character array and enter your name and reverse it using pointer.

**#WEEK 4:**

1. Write a program to search an element using Linear Search.
2. Write a program to search an element using Binary Search.
3. Write a program to sort the given array using Bubble Sort.

**#WEEK 5:**

1. Write a program to sort the given array using Selection Sort.
2. Write a program to sort the given array using Insertion Sort.
3. Write a program to sort the given array using Quicksort.

**#WEEK 6:**

1. Write a program to sort the given array using Merge Sort.
2. Write a program to insert a new element in the given unsorted array at  $k^{\text{th}}$  position.
3. Write a program to delete an element from given sorted array.

**#WEEK 7:**

1. Write a program to merge to given sorted arrays.
2. Write a program to implement Stack using array, also show overflow and underflow in respective push and pop operations.
3. Write a program to implement Queue using array, which shows insertion and deletion operations.

**#WEEK 8:**

1. Write a program to implement Circular Queue using array, which shows insertion and deletion operations.
2. Write a program to implement Linear Linked List, showing all the operations, like creation, display, insertion, deletion and searching.
3. Write a program to implement Stack, using Linked List. Implement Push, Pop and display operations.

**#WEEK 9:**

1. Write a program to implement Queue, using Linked List. Implement Insertion, deletion and display operations.
2. Write a program to count the number of times an item is present in a linked list.
3. Write a program to increment the data part of every node present in a linked list by 10. Display the data both before incrementation and after.

**#WEEK 10:**

1. Write a program to implement Doubly Linked List, showing all the operations, like creation, display, insertion, deletion and searching.
2. Write a program to create a Binary Search Tree and display its contents using recursive preorder, post-order and in-order traversal.
3. Write a program to implement deletion of a node in binary search tree.

#### **#WEEK 11:**

1. Write a program to implement Binary tree and display the contents using non-recursive preorder, post-order and in-order traversal techniques.
2. Write a program to sort the given array using Heapsort.
3. Write a program of Graph Traversal-Depth first search and Breadth first search.

#### **#WEEK 12:**

1. Write a program to implement Prim's algorithm
2. Write a program to implement Kruskal algorithm.

\*\*\*\*\*

#### ***Textbooks:***

- Yashwant Kanetkar," Data Structure using C", BPB Publication, 5<sup>th</sup> Edition ,2011
- A.Tannenbaum,Y. Lanhgsam and A.j. Augenstein ," Data Structures Using C And C++ ",Prentice Hall of India,2<sup>nd</sup> Edition,2009.
- Jean-Paul Tremblay, P.G Sorenson, "An Introduction to Data Structures with applications", McGraw-Hill ,2nd Edition ,1984.

#### ***References:***

- Robert L Kruse, "Data Structure and Program Design in C", Prentice Hall (1991).
- Noel Kalicharan ,"Data Structure in C" ,Ist Edition Create space publisher, 2008.
- Mark Allen Weiss,"Data Structure and algorithm Analysis in C",2<sup>nd</sup> Edition AddisonWesley,1996.
- E. Balagurusamy, "Problem Solving through C language", TMH publication, Fourth Edition, 2008.
- R.S Salaria,"Data Structures & Algorithms using C",Khanna Publication,4<sup>th</sup> Edition,2009
- E.Horowitz and S.Sahni,"Fundamentals of Data Structures in C ",2<sup>nd</sup> Edition, Universities Press,2008.