

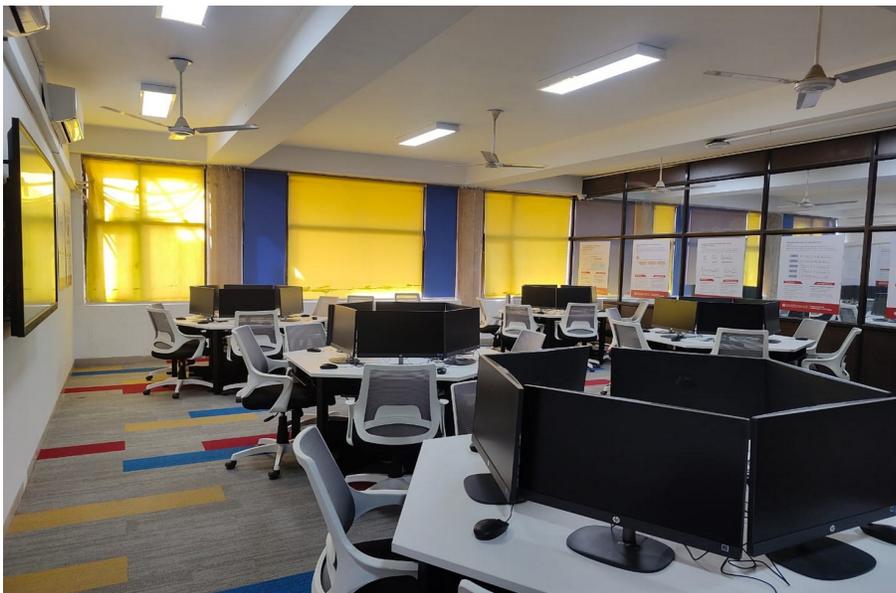
# **Data Structure and Algorithm Lab**

## **Department of Information Technology**

### **Lab Description:**

The Data Structures and Algorithms Lab offers practical exposure to the implementation and analysis of fundamental data structures and algorithms. Students design and execute programs involving arrays, linked lists, stacks, queues, trees, and graphs. The lab emphasizes algorithmic efficiency, time and space complexity, and problem-solving skills. It strengthens logical thinking and prepares students for competitive programming and real-world software development.

### **Lab Photos:**







**Available Software:**

S.No.	Software Name
1	Windows 11
2	Ms Office
3	Dev C++
4	Java(JDK 24)
5	MySQL 8.0
6	Cisco Packet Tracer
7	Python 3.12
8	Android Studio
9	Google Chrome

**Major Equipments Available in Lab:**

S.NO	Equipments Name	Items
1	Computer System	36
2	Keyboard	36
3	Mouse	36
4	LED Smart Panel	Samsung 75"

### **Labs Conducted in E 108A:**

Odd Sem: Data Structure Lab (BCS 351), Design and Analysis of Algorithms Lab (BCS 553)

Even Sem: OOPs with Java Lab (BCS 452)

### **List of Experiments – Data Structure Lab**

**Subject Code: BCS 351**

**Branch: IT**

**Subject Name: Data Structure Lab**

**Sem: III**

<b>Course Outcomes</b>	<b>Statement(On completion of this course, students will be able to)</b>
<b>CO1</b>	To evaluate the suitability of various data structures like arrays, linked lists, stacks, and queues and implement them for solving real world problems based on factors like time complexity, space usage, and implementation constraints.
<b>CO2</b>	To implement and evaluate various searching and sorting algorithms.
<b>CO3</b>	To analyze the performance and behavior of tree and graph algorithms like DFS, BFS, and MST under different problem constraints.

### **As per AKTU Syllabus**

<b>Exp. No.</b>	<b>Name of Experiment</b>	<b>CO Mapping</b>
1	Write a program to implement Bubble Sort, Selection Sort and Insertion Sort.	CO2
2	Write a program to implement Linear Search and Binary Search.	CO2
3	Write a program to implement singly linked list: Insert, Display, Delete, Traversal.	CO1
4	Write a program to implement doubly linked list: Insert, Display, Delete, Traversal.	CO1
5	Write a program to implement circular linked list: Insert, Display, Delete,	CO1

	Traversal.	
6	Write a program to implement polynomial addition.	CO1
7	Write a program to implement Stack using Array and Linked List.	CO1
8	Write a program to implement the conversion of infix notation to postfix notation.	CO1
9	Write a program to implement queue using Array and Linked List.	CO1
10	Write a program to implement circular queue using array.	CO1
11	Write a program to implement priority queue using linked list.	CO1
12	Write a program to implement Quick Sort and Merge Sort	CO2
13	Write a program to implement binary search tree using linked list.	CO3
14	Write a program to implement recursive traversal: Preorder, Postorder, and Inorder.	CO3
15	Write a program to implement heap sort.	CO2,CO3
16	Write a program to implement BFS and DFS using linked list.	CO3

**Additional Programs (Content Beyond Syllabus):**

<b>Exp. No.</b>	<b>Name of Experiment</b>	<b>CO Mapping</b>
17.	Write a program to implement Counting Sort.	CO2
18.	Write a program to implement B+ tree.	CO3

19.	Write a program to implement Red Black Tree.	CO3
20.	Write a program to implement any minimum spanning tree algorithm.	CO3

### **List of Experiments – Design and Analysis of Algorithm Lab**

**Subject Code: BCS 553**

**Branch: IT**

**Subject Name: Design and Analysis of Algorithm Lab**

**Sem: V**

<b>Course Outcomes</b>	<b>Statement(On completion of this course, students will be able to)</b>
<b>CO1</b>	Understand and implement algorithms to solve problems by iterative approach and by divide and conquer approach.
<b>CO2</b>	Understand and implement algorithm to solve problems by the Greedy algorithm approach.
<b>CO3</b>	Understand and analyze algorithm to solve problems by Dynamic programming, Backtracking and by branch and bound approach

<b>S. No.</b>	<b>Experiment Name</b>	<b>CO Mapping</b>
1	Program for Recursive Binary & Linear Search	CO1
2	Write a program to implement of Bubble & Insertion Sort.	CO1
3	Write a program to implement Heap Sort.	CO1
4	Write a program to implement Merge Sort	CO1
5	Write a program to implement Quick Sort.	CO1

6	Write a program to implement Selection Sort.	CO1
7	Write a program to implement Shell Sort.	CO1
8	Write a program to implement Counting Sort, Bucket Sort & Radix Sort.	CO1
9	Write a program to implement 0/1 and Fractional Knapsack Problem using Greedy solution.	CO2
10	Write a program to find the minimum cost spanning tree of a given connected undirected graph using Kruskal's algorithm.	CO2
11	Write a program to find the minimum cost spanning tree of a given connected undirected graph using Prim's Algorithm.	CO2
12	Write a program to find shortest paths to other vertices using Dijkstra's algorithm.	CO2
13	Write a program to find the Single Source Shortest Path problem using Bellman Ford's Algorithm.	CO2
14	Write a program to solve Matrix Chain Multiplication Problem using Dynamic Programming Approach.	CO3
15	Write a program to solve Longest Common Subsequence Problem using Dynamic Programming Approach.	CO3
16	Write a program to implement 0-1 Knapsack problem using Dynamic Programming Method	CO3
17	Write a program to perform Travelling salesman problem using Branch & Bound approach.	CO3
18	Write a program to implement the N-Queen problem using Backtracking.	CO3
19	Sort a Given n number of integer elements using quick sort method and compute its time complexity. run the program for n>5000 varied values of and record the time taken to sort. Plot a graph of the time taken versus non graph sheet. The elements can be read from a file or can be generated through random number generator. Demonstrate using java how the divide-and-conquer method works along with its time complexity analysis: worst case, average case and best case	CO1
20	Sort a Given n number of integer elements using merge sort method and compute its time complexity. run the program for n>5000 varied values of and record the time taken to sort. Plot a graph of the time taken versus non	

	graph sheet. The elements can be read from a file or can be generated through random number generator. Demonstrate using java how the divide-and-conquer method works along with its time complexity analysis: worst case, average case and best case.	CO1
21	Write a program to implement all pair Shortest problems using Flyod's Algorithm.	CO3
22	Design and implement to find a subset of a given set $S = \{S_1, S_2, \dots, S_n\}$ of n positive integers whose SUM is equal to a given positive integer d. For example, if $S = \{1, 2, 5, 6, 8\}$ and $d = 9$ , there are two solutions $\{1,2,6\}$ and $\{1,8\}$ . Display a suitable message, if the given problem instance doesn't have a solution.	CO3
	Write a program to design and implement to find all Hamiltonian Cycle in a connected undirected graph using Backtracking principles.	CO3
23	The absolute difference is the positive difference between two values a and b, is written $ a-b $ or $ b-a $ and they are equal. If $a=3$ and $b=2$ , $ 3-2 = 2-3 =1$ . Given an array of integers, find the minimum absolute difference between any two elements in the array	CO2
24	There are two -element arrays of integers, A and B. Permute them into some A' and B' such that the relation $A'[i]+B'[i] \geq k$ holds for all where $0 \leq i < n$ . There will be q queries consisting of A, B, and k. For each query, return YES if some permutation A', B' satisfying the relation exists. Otherwise, return NO.	CO3
25	Given an amount and the denominations of coins available, determine how many ways change can be made for amount. There is a limitless supply of each coin type.	CO3

### List of Experiments –OOPs with Java Lab

**Subject Code: BCS452**

**Branch: IT**

**Subject Name: Object Oriented Programming with Java**

**Sem: III**

<b>Course Outcomes</b>	<b>Statement(On completion of this course, students will be able to)</b>
<b>CO1</b>	To apply fundamental Java programming constructs, object creation techniques, and development tools to design and execute basic Java applications.
<b>CO2</b>	To implement and analyze object-oriented concepts, exception handling,

	multithreading, packages, file I/O, GUI components, and modern Java features in Java programs.
<b>CO3</b>	<b>To build, test, and evaluate industry-oriented Java applications using Spring Framework and Spring Boot for web, RESTful, and frontend applications.</b>

### As per AKTU Syllabus

<b>Exp. No.</b>	<b>Name of Experiment</b>	<b>CO Mapping</b>
1	Use Java compiler and eclipse platform to write and execute java program. a) Write a java program to check the number if it is prime or not. b) Write a java program to generate Fibonacci series.	<b>CO1</b>
2	Create simple java programs using command line arguments. a) Create a java program to initialize the objects of a class by using :- a) reference variable b) method c) constructor	<b>CO1</b>
3	Understand OOP concepts and basics of Java programming. a) Write a program to create a class student with data “name”, “city” and age along with method printData() to print the data. Create two objects s1, s2 to declare and access the values.	<b>CO1</b>
4	Create a java program to implement abstraction.	<b>CO2</b>
5	Create Java programs to implement inheritance.	<b>CO2</b>
6	Create a java program to implement method overloading and method overriding (polymorphism).	<b>CO2</b>
7	Create a java program to implement error-handling techniques by exception handling and multithreading. a) Write a program in java if number is less than 10 and greater than 50. It generate the exception out of range. Else it display the square of number. b) Write a program in java to enter the numbers through command line arguments, if first and second number not entered it will generate the exception. Also divide the first number with second number and generate the Arithmetic Exception and also handle it.	<b>CO2</b>
8	Create java program with the use of java packages.	<b>CO2</b>
9	Construct java program using Java I/O package a) Write a java program to read a file and display the content on screen.	<b>CO2</b>

10	Create a java AWT program to create button, textfield, radiobuttons and checkboxes.	CO2
11	Create a java program to implement stream API functions (Java's new features)	CO2
12	Create industry oriented application using Spring Framework.	CO3
13	Test RESTful web services using Spring Boot.	CO3
14	Test Frontend web application with Spring Boot.	CO3

**Additional Programs (Content Beyond Syllabus):**

<b>Exp. No.</b>	<b>Name of Experiment</b>	<b>CO Mapping</b>
17.	Create a Java program using <b>Collections Framework</b> to store student details in an ArrayList and sort the data using <b>Comparator</b> interface.	CO2
18.	Create a Java program to implement <b>Generics</b> using a generic class and a generic method.	CO2
19.	Create a Java program to implement <b>Functional Interfaces and Lambda Expressions</b> .	CO2
20	Create a Java program using <b>JDBC</b> to connect to a database and perform insert and retrieve operations.	CO3