



MANAKULA VINAYAGAR INSTITUTE OF TECHNOLOGY

An Autonomous Institution

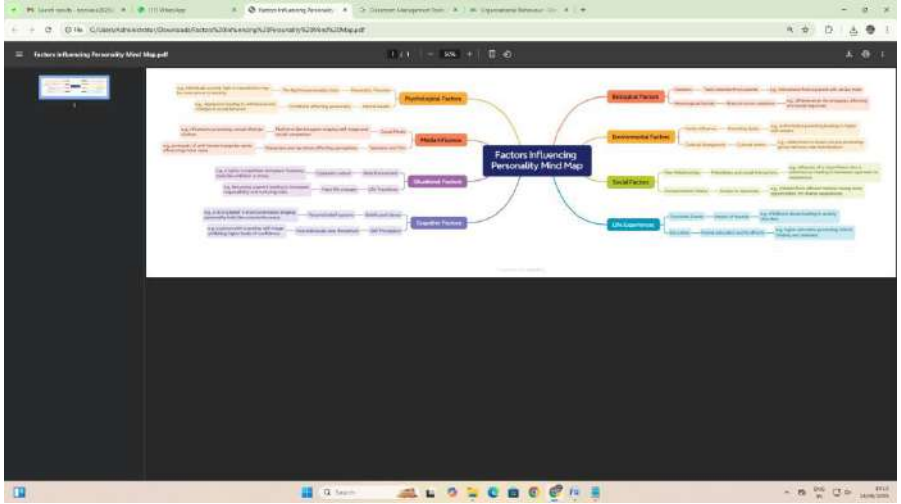
Affiliated to Pondicherry University, Approved by AICTE, New Delhi,
Accredited by NBA, New Delhi and NAAC with 'A' Grade
Kalitheerthalkuppam, Puducherry- 605 107.



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

INNOVATIVE ICT TOOLS EMPLOYED IN CLASSROOMS

1. VIRTUAL LAB: 1. https://ds2-iiith.vlabs.ac.in/exp/min-spanning-trees/prims-algorithm/mst_prims_demo.html
2. https://ds2-iiith.vlabs.ac.in/exp/min-spanning-trees/kruskals-algorithm/mst_kruskal_demo.html

Subject Code/Name	CSPL402/DESIGN AND ANALYSIS OF ALGORITHM
Year/Sem	II CSE-B/IV
Date	
Tool Employed	VIRTUAL LAB
Topic	Preparation of Balance sheet
Outcome	The Outcome of this Mindmap is to analyse individual and group dynamics within a corporate environment. While each enterprise is distinct and varies across sectors, the common aspiration among them is to foster an organizational culture that harmonizes productivity with employee contentment.
PHOTO	

INNOVATIVE ICT TOOLS EMPLOYED IN CLASSROOMS

DATA STRUCTURE AND ALGORITHMS

Virtual Labs
Bubble Sort

★★★★☆ Rate Me Report a Bug

Instructions

Question: Sort the given array using Bubble Sort.

24 39 30 49 11 76

Observation:
CORRECT ANSWER

Submit Next Stop Undo Reset

Virtual Labs
Bubble Sort

★★★★☆ Rate Me Report a Bug

Computer Science and Engineering > Data Structures - 1 > Experiments

Aim
Overview
Recap
Prestest
Bubble Sort
Optimized Bubble Sort
Code Assessment
Analysis
Posttest
Further Readings/References
Feedback

Choose difficulty: Beginner Advanced

1. Which of the following is an array?
 a. 1, 2, 3, 4, 5 Explanation
 b. [A, 3, 4, "Hello world"] Explanation
 c. [Test, Poku, Y, 100] Explanation
 d. [True, False, 6, 7, 8] Explanation

2. Which of the following is an array sorted in ascending order?
 a. 1, 4, 5, -100 Explanation
 b. -10, -15, 15, 100 Explanation
 c. -100, 0, 10, 20 Explanation
 d. 100, 50, 30, 0 Explanation

3. Which of the following is not a sorting algorithm?
 a. Bubble Explanation
 b. Selection Explanation
 c. Merge Explanation
 d. Binary Explanation

4. Consider the following array: A = [1, 2, 3, 4, 5]. Identify A[z], i.e. the element with index z from the following (assume 0-indexed array).
 a. 4 Explanation
 b. 5 Explanation
 c. 1 Explanation
 d. 6 Explanation

Submit Quiz
Score: 4 out of 4

INNOVATIVE ICT TOOLS EMPLOYED IN CLASSROOMS

CSPC403/DESIGN AND ANALYSIS OF ALGORITHM

The screenshot shows a virtual lab environment for the Bubble Sort algorithm. The interface includes a navigation menu on the left, a main content area with problem details, and a code editor with an output window.

Navigation Menu:

- Aim
- Overview
- Recap
- Pretest
- Bubble Sort
- Optimized Bubble Sort
- Code Assessment
- Analysis
- Posttest
- Further Readings/References
- Feedback

Main Content:

Bubble Sort

PROBLEM 1 | **PROBLEM 2**

Basic Bubble Sort

Objective

Implement basic Bubble Sort algorithm for the given input array

Input Format

Input consists of an array of unsorted array - `inp1`. It may be of arbitrary length.

Output Format

An array of numbers Eg. 1,2,4,5 sorted in ascending orders.

Code Editor:

```
1 const func = (inp1) => {
2   let n = inp1.length;
3   for (let i = 0; i < n - 1; i++) {
4     for (let j = 0; j < n - i - 1; j++) {
5       if (inp1[j] > inp1[j + 1]) {
6         let temp = inp1[j];
7         inp1[j] = inp1[j + 1];
8         inp1[j + 1] = temp;
9       }
10    }
11  }
12  return inp1;
13 }
14 const result = func([10, 0, 36, 40, 11, 99]);
```

Output:

Correct

Code Output: [11, 10, 40, 36, 46, 99]

Expected Output: [11, 10, 40, 36, 46, 99]

Submit

VIRTUAL LABORATORY ASSIGNMENT

Design and Analysis of Algorithms | Department of Computer Science and Engineering
Academic Year 2025-2026

Virtual Lab Working Screenshots

Experiment 1: Prim's Algorithm — Greedy Technique (MST)

Virtual Lab - Prim's Algorithm
Graph Visualization: Program Output

```
6 vertices
Edge number of vertices: 6
Edge number of edges: 9
0 1 1
0 2 1
0 3 1
0 4 1
0 5 1
1 2 1
2 3 1
3 4 1
4 5 1
Total MST Cost: 14
```

Experiment 2: Kruskal's Algorithm — Greedy Technique (MST)

Virtual Lab - Kruskal's Algorithm
Graph Visualization: Program Output

```
6 vertices
Edge number of vertices: 6
Edge number of edges: 9
0 1 1
0 2 1
0 3 1
0 4 1
0 5 1
1 2 1
2 3 1
3 4 1
4 5 1
Total MST Cost: 14
```

Experiment 3: Multistage Graph — Dynamic Programming (Shortest Path)

Virtual Lab - Multistage Graph - Dynamic Programming
Graph Visualization: Program Output

```
6 vertices
Edge number of vertices: 6
Edge number of edges: 9
0 1 1
0 2 1
0 3 1
0 4 1
0 5 1
1 2 1
2 3 1
3 4 1
4 5 1
Total MST Cost: 14
```

- Aim
- Overview
- Recap
- Pretest
- Quick Sort
- Code Assessment
- Analysis
- Posttest**
- Further Readings/References
- Feedback

Quick Sort Experiment

1 Quick-Sort is:

- a. A stable sorting algorithm
- b. Not a stable sorting algorithm
- c. Depends on the implementation

2. Given a random policy for choosing the pivot, when is the worst case time complexity of Quick Sort hit:

- a. When most of the chosen pivots are the smallest in the subarray
- b. When most of the chosen pivots are the biggest in the subarray
- c. When most of the chosen pivots are smaller than some and bigger than some
- d. a and b.

3 Function

```
quick_sort(array) -> sorted_array:
    if array is empty: return array
    pivot = choose_pivot_index()
    smaller_array = [], bigger_array = []
    for i = 1 to sizeof(array):
        if (-----A-----):
            append array[i] to smaller_array
        else if (-----B-----):
            append array[i] to bigger_array
    return quick_sort(smaller_array) + [array[pivot]] + quick_sort(bigger_array)
```

Which of the following pairs can be selected simultaneously for blanks (A) and (B) in the quick-sort algorithm, and would cover all edge cases.

- a. array[i] < array[pivot] and array[i] > array[pivot]
- b. array[i] < array[pivot] and array[i] > array[pivot]
- c. array[i] < array[pivot] and array[i] > array[pivot]