

Algorithm and Complexity Analysis

- I. For this activity, organize yourselves into groups of five members each.
- II. You are required to implement the following algorithms in Java:
 1. Searching Algorithms
 - a. Linear Search
 - b. Binary Search
 2. Sorting Algorithms
 - a. Quicksort
 - b. Merge Sort
 - c. Heap Sort
 - d. Bubble Sort
 - e. Insertion Sort
 - f. Selection Sort
 - g. Shell Sort
- III. Instructions
 1. Create a Java program named AlgorithmComplexityTest.java.
 2. Implement all the listed algorithms inside separate methods.
 3. Generate test input data:
 - a. Use arrays of random integers.
 - b. Test with at least five different input sizes, e.g.,
 - 1,000
 - 5,000
 - 10,000
 - 50,000
 - 100,000
 4. For searching algorithms:
 - a. Ensure the array is sorted before testing Binary Search.
 - b. Pick a random key from the array to search.
 5. For sorting algorithms:
 - a. Create a copy of the same random array for each sorting algorithm so results are consistent.
 - b. Sort and measure the time for each algorithm separately.
 6. Measure Execution Time:
 - a. Display results in milliseconds
 7. Display Results:
 - a. For each input size, display the execution time of all algorithms in a table-like format.
 8. Analysis:
 - a. Compare the execution times of the algorithms.
 - b. Identify which algorithms scale well and which become inefficient as input size increases.
 9. Presentation:
 - a. Each group member will be randomly selected to present and explain the algorithm that has been assigned or picked for them.