CIT300 Graded Practical Assignment 3 (Week 14):
Algorithm Analyzer – Measuring Performance & Complexity

Scope: Weeks 1–13 (Search, Sort, Hashing, Algorithmic Complexity)

Objective: Each student will develop a simple program to measure and display how long their assigned algorithm takes to run on different input sizes.
This practical assignment contributes 10% towards the final module grade.

Each team member should:

1. Implement **one algorithm** (as assigned below).
2. Run it on arrays of different sizes (e.g., **100, 500, 1000 elements**).
3. Measure the **execution time** using System.currentTimeMillis() or System.nanoTime().
4. Display the results in a small **table format**.

Example Output :

Algorithm: Bubble Sort

```
Input Size | Time (ms)
----------------------
100        | 0.15
500        | 0.80
1000       | 3.20
```

Team Role Distribution:

Member	Task
Member 1	Implement **Linear Search** — measure time for finding an element in arrays of sizes 100, 500, and 1000.
Member 2	Implement **Binary Search** — sort the array first, then measure search time for the same input sizes 100, 500 and 1000.
Member 3	Implement **Bubble Sort** — measure sorting time for input sizes 100, 500, and 1000.
Member 4	Implement **Quick Sort** — measure sorting time for input sizes 100, 500, and 1000.

Hints:

- Generate random integer arrays for testing.
- You can use the same random numbers for each run.

Deliverables

- GitHub repository link + one merged demo video
- Each member records their own part and all clips are merged before submission
- Collaboration must be visible through branches, commits, and pull requests

Deadline: Submit the completed assignment to the given link in the LMS on or before November 16th.