1. Understand Array Representation:

Explain how arrays are represented in memory and their advantages.

Ans: Arrays are represented in memory as a contiguous block of storage, allowing for constant-time access to elements via their indices. Their advantages include efficient memory usage, fast access times, and cache friendliness, making them suitable for many applications where performance and straightforward implementation are key.

4. Analysis:

Analyze the time complexity of each operation (add, search, traverse, delete).

Ans: Time Complexity of Each Operation are

1.Add: O(1)

2.Search : O(n)

3.Traverse : O(n)

4.Delete : O(n)

Discuss the limitations of arrays and when to use them.

Ans: Limitations:

1. Once created, the size cannot be changed, leading to potential waste or insufficiency.

2. Inefficient for inserting or deleting elements, as it requires shifting.

3. Requires contiguous memory, which can be problematic for large arrays.

4. No built-in methods for operations like resizing .

5. stores elements of the same type.

6. Fixed size may lead to inefficient memory use.

When to use them: When the number of elements is known and constant, When fast, constant-time access by index is needed, For managing large amounts of simple data with contiguous memory needs.