**Exercise 4: Employee Management System**

**1. Understand Array Representation:**

**Explanation of Arrays in Memory:**

* **Contiguous Memory Allocation:** Arrays are stored in contiguous memory locations, which means all elements are stored next to each other in memory. This allows for efficient access to any element using its index.
* **Fixed Size:** The size of an array is fixed at the time of creation, which means the number of elements it can hold is determined when it is created.
* **Direct Access:** Elements in an array can be accessed directly using their index. This provides constant time complexity (O(1)) for accessing elements.

**Advantages of Arrays:**

* **Fast Access:** Direct access to elements using their index allows for quick read and write operations.
* **Simple Data Structure:** Arrays are straightforward to implement and use.
* **Memory Efficiency:** For a known number of elements, arrays use memory efficiently.

**Analysis:**

**Time Complexity of Each Operation:**

* **Add Operation:** O(1) (if there is space in the array)
* **Search Operation:** O(n) (linear search through the array)
* **Traverse Operation:** O(n) (iterating through all elements)
* **Delete Operation:** O(n) (finding the element to delete, then shifting elements)

**Limitations of Arrays:**

* **Fixed Size:** Once an array is created, its size cannot be changed. This can lead to wasted space if the array is not fully used or the need to reallocate and copy to a larger array if it is exceeded.
* **Inefficient Deletions and Insertions:** Deleting or inserting elements (except at the end) requires shifting elements, which can be inefficient (O(n) time complexity).
* **Memory Allocation:** Arrays must be allocated with a fixed size at creation, which can be inefficient if the exact number of elements is not known.

**When to Use Arrays:**

* **Fixed Number of Elements:** When the number of elements is known and does not change frequently.
* **Fast Access:** When fast access to elements is required using their index.
* **Simple Use Cases:** For simple use cases where dynamic resizing or frequent insertions and deletions are not required.