**Employee Management System**

# **Explain how arrays are represented in memory and their advantages**

**Representation of Arrays in Memory:**

* In Arrays elements are stored in contiguous memory location.
* Each element of in an array has fixed memory size.
* Array uses ‘0’ based indexing.

**Advantages of Arrays:**

* Any i’th element in array can be found by this formula:  
  **[Address of i’th element in memory = M + I + sizeOf(element)]**. This helps in accessing element in **constant time complexity - O(n).**
* We can predict memory usage if we use array to store elements as the elements has fixed memory size.
* Due to their contiguous memory allocation, arrays have a good data locality of reference. When a part of the array is loaded into the CPU cache, subsequent accesses to nearby elements are faster, as they are likely already in the cache.

# **Analyse the time complexity of each operation (add, search, traverse, delete).**

* **Add:** In my EMS, I’m storing employees in sorted manner based on their employeeId.
  + **Best case:** O(1) if the employeeId is larger than the rest, so it is stored at the end.
  + **Avg.** **case:** O(n), if the employeeId is to be stored at the middle of array. Then, shifting operation need to be performed.
  + **Worst case:** O(n), if the employeeId is to be stored at first position of array. Then, all the elements need to be shifted by one.
* **Search:**
  + Searching with employeeId takes O(log(n) time in worst case as it uses **Binary Search.**
  + Searching with employeeName takes O(n) time in worst case as it uses **Linear Search.**
* **Traverse:**
  + Traversing operation takes O(n) time complexity, as we’ve to iterate over the whole array.
* **Delete:**
  + Delete is taking same time complexity as the scenario is same as adding.

# **Discuss the limitations of arrays and when to use them.**

* Arrays typically has a fixed length. So, we can’t increase or decrease its size dynamically unless we use dynamic arrays like ArrayList.
* Insertion & Deletion require shifting of element, so that spaces could be filled between element.