# **Lab 2 Report-**

Dhiren Mahajan

**Class 1:**

* Add item: we are just iterating over the list from the beginning, adding a new item wherever we find an empty space (null pointer), and then ensuring that the array remains sorted by shifting items until the new element is in its proper place.
* Remove item: We are deleting the element at a specific index and relocating all the items after it to fill up the empty space. This approach works well if the list has few entries. When the list's length grows, a lot of comparisons and moving of components must be performed.

**Class 2:**

* Add item: the rationale is the same as in Job 1, with the exception that we are iterating over the list from the beginning.
* Remove item: the same as job 1. This approach does the same thing as job 1. When compared to job 1, there are no notable advantages or disadvantages.

## **Class 3:**

* Add item: In this procedure, the proper location for the element to be added is found first, and then the element is inserted in the position by moving the elements after they have created a space for it. As a result, fewer comparisons must be done, but many items must be shifted.
* Remove item: The item at the specified index is removed. To fill the vacant area, the following pieces are not moved. Therefore, 0 movements are required in this function. The delete function is particularly fast since it requires fewer comparisons and movements. However, several movements are required for the add item function.

Task 4.2:

Text

Description automatically generatedText

Description automatically generatedText

Description automatically generated

Tasks 1 and 2 provide comparable results, as previously stated. When adding an element, Task 3 makes a lot of motions. As previously stated, Task 3's delete function moves 0 items.

Task 4.3:

Text

Description automatically generated

Text

Description automatically generatedText

Description automatically generated

Tasks 1 and 2 get comparable results. Their performance deteriorates as the number of components increases. However, for job 3, the add item method conducts fewer comparisons but more movements. The deleted item corresponds to our previous observations.

Task 4.4:

Text

Description automatically generated

Text

Description automatically generated

Text

Description automatically generated

When the list length is short, classes 1 and 2 perform exceptionally well. Class 3 is less effective for fewer components. Only when the total number of components to be added is 50 or greater does it outperform the other classes.