**OBJECTIVE:**

The objective of this homework assignment is to practice the implementation of linked list techniques. Your task is to implement the bubble sort algorithm using **single** and **doubly** linked lists.

Though this is not the best way to implement to program bubble sort, it is a good exercise for the creation, addition, and deletion of items from these types of data structures.

**IMPLEMENTATION**  
Your implementation must generate a random number n of size between 0 and 100. n in this case represents the number of elements to be generated and sorted.

Second, you must generate and link n **random** positive integers in a linked list where each item is less than 50,000.

Your implementation must involve two approaches: a single linked list and a doubly linked list data structure.

**OUTPUT**  
The output must include  
1) The unsorted list after the items are linked with indication that the list is out of the single or doubly linked lists  
2) The sorted lists  
3) For each case, the **number of comparisons** made, the **number of exchanges**, as well as the total **number of node traversals** made. This should be in a separate word or text document with a brief explanation.

**NOTES**

YOU HAVE TO WRITE YOUR OWN IMPLEMENTATION OF SINGLE AND DOUBLY LINKEDLIST. WRITE THE METHODS ONLY WHICH YOU NEED.

You might need a Node class

You might need a ListIterator

LinkedList:-

The **number of comparisons** made- for bubble sort the no of comparison made is the size of LinkedList exponential which is (size)\*(size-1), as two for loops are used for comparison

the **number of exchanges**,- it depends upon the LInkedList

1. If linkedList is already sorted then no need to exchange node.

means No. of exchange = 0

1. If linkedList sorted in descending order and we need to sorted it by ascending order then No. of exchange = (size-1)!

as well as the total **number of node traversals** made- to check whether LinkedList is sorted or not we need to travel the nodes once for minimum and maximum (size) \* (size-1) times.