Name: Ali Albayrak

Student ID: P304320

Date: 12/09/17

Portfolio Activity Task 1.3: Sorting Algorithm

Lecturer: Stewart Godwin

Table of Contents

[Introduction: 3](#_Toc493006809)

[Pseudo Code for Insertion Sort: 3](#_Toc493006810)

[The computational complexity of the Insertion Sort algorithm: 3](#_Toc493006811)

[Discussion: 4](#_Toc493006812)

[Advantages of Insertion Sort: 4](#_Toc493006813)

[Disadvantages of Insertion Sort: 4](#_Toc493006814)

[Conclusion: 4](#_Toc493006815)

[Test: 5](#_Toc493006816)

[Test table and expected outputs: 5](#_Toc493006817)

[Evidence of testing procedure: 5](#_Toc493006818)

[Load button test: 5](#_Toc493006819)

[Add First button test: 6](#_Toc493006820)

[Sort Button Test: 6](#_Toc493006821)

[References: 8](#_Toc493006822)

# Introduction:

Insertion Sort

Insertion sort is a sorting algorithm that works by taking elements from the set one by one and inserting them in their correct position into a new sorted set.

# Pseudo Code for Insertion Sort:

i ← 1

**while** i < length(A)

x ← A[i]

j ← i - 1

**while** j >= 0 **and** A[j] > x

A[j+1] ← A[j]

j ← j - 1

**end while**

A[j+1] ← x[[4]](https://en.wikipedia.org/wiki/Insertion_sort#cite_note-4)

i ← i + 1

**end while**

# The computational complexity of the Insertion Sort algorithm:

Worst Case: [O](https://en.wikipedia.org/wiki/Big_O_notation)(*n*2)

Best Case: O(n)

Average case for a random array: [O](https://en.wikipedia.org/wiki/Big_O_notation)(*n*2)

Almost sorted case: O(n)

# Discussion:

## Advantages of Insertion Sort:

* It has a simple implementation.
* Only requires a constant amount O(1) of additional memory space.
* It is efficient for smaller data sets.
* More efficient in practice than most other simple quadratic (i.e., [O](https://en.wikipedia.org/wiki/Big_O_notation)(*n*2)) algorithms such as selection sort or bubble sort

## Disadvantages of Insertion Sort:

* Insertion sort has (n\*n) time complexity on average and worst cases. Because of this, insertion sorting algorithm is not suitable for large data sets.

# Conclusion:

As a conclusion, when we consider that our data is not large, time complexity of insertion sort is not a big issue for us. Also, it is faster than bubble sort and selection sort.

Comparing with Selection and Bubble Sort:

While all of three sorts have same Big O Complexity values, they are some difference between their speed of run process. For example, while selection and bubble sort must scan the remaining parts of the array when placing an element, insertion sort only scans as many elements as necessary.

Moreover, because of the advantages of insertion sort such as simple implementation and little memory space, I decided to use it as a sorting algorithm.

# Test:

## Test table and expected outputs:

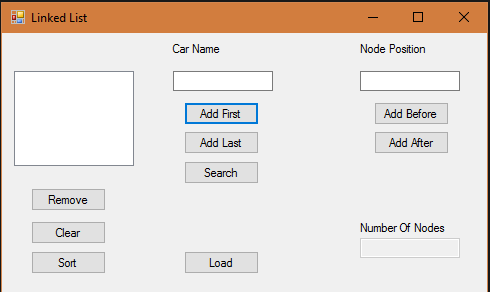
This program is not first version. I builded this program on previous assessment. So, I am testing only new features that added. Load and Sort buttons have been added in this version.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Button | CarName Textbox | NodePosition Textbox | Expected Output | Result |
| Load | - | - | Loads list of car names from cars.txt | Success |
| Add First | Tesla | - | Tesla should be placed in the beginning of the list | Success |
| Sort | - | - | All of the list should be sorted alphabetically | Success |

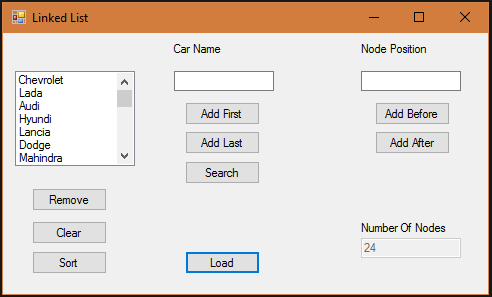
# Evidence of testing procedure:

## Load button test:

Before button clicked:

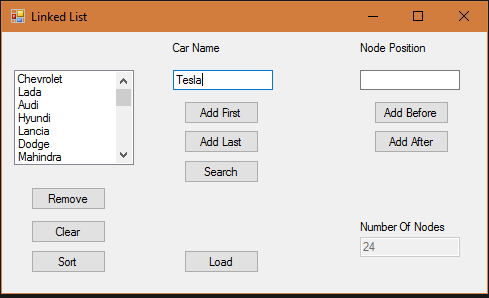


After button clicked:

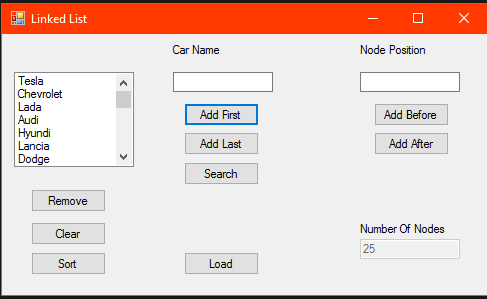


## Add First button test:

Before Button Clicked:



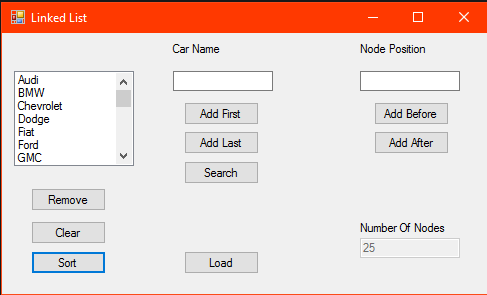
After button clicked:



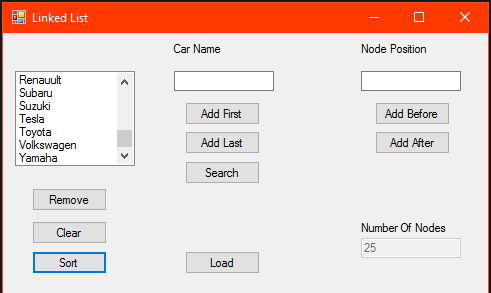
## Sort Button Test:

After sort button clicked:

First screenshot:



Second screenshot:



# References:

<https://en.wikipedia.org/wiki/Insertion_sort>

<http://www.geeksforgeeks.org/insertion-sort/>

<https://www.khanacademy.org/computing/computer-science/algorithms/insertion-sort/a/insertion-sort>

<https://www.tutorialspoint.com/data_structures_algorithms/insertion_sort_algorithm.htm>

<https://johnderinger.wordpress.com/2012/12/28/quadratic-and-linearithmic-comparison-based-sorting-algorithms/>

<https://www.quora.com/Why-is-insertion-sort-faster-than-bubble-sort-while-having-the-same-big-O-notation>

<http://cheetahonfire.blogspot.com.au/2009/05/selection-sort-vs-insertion-sort.html>