Ryan Ellis
COSC 220-003
Project #2: Linked Lists
Section 3.3 – Timing Sorts
11/5/2024

Recorded Output Findings

Reviewing the given data for these sorting methods, it is apparent that the **Linked List**, **Doubly Linked List** and **Bubble Sort** methods listed are significantly slower in sorting comparatively to **Insertion** and **Selection Sort**. Each of the slower methods all hold values that are close in range to each other, and comparatively the other methods hold values that are also within the same range.

Within the prospective grouping of the similarly timed sorting methods, each group does tend to share the same amplitude of graph shape given their data yields similar results. Both groups have a parabolic figure with differentiating change in slopes indicating the slower methods are costly for the machine, however the rate of change seems to be consistent between the two graphs.

Conclusion

The underlying principle that separates the **Linked List**, **Doubly Linked List**, and **Bubble Sort** from the **Insertion** and **Selection Sort** lies within the algorithm differences between the two methods of sorting. The **Linked Lists** structures and **Bubble Sort** use a linear method of sorting, whereas the **Insertion** and **Selection Sort** methods use a more intuitive and efficient method to sort.

List Sizes	Linked (s)	Double (s)	Bubble (s)	Insertion (s)	Selection (s)
10000	0.209023	0.190745	0.514556	0.07399	0.133511
25000	1.89739	1.87655	3.2342	0.462815	0.831096
50000	9.32933	9.19114	12.9565	1.82135	3.31705
75000	23.0542	23.3958	29.0653	4.08228	7.50057
100000	48.9321	45.5336	51.8135	7.27822	13.261

Table 1: Sorting times

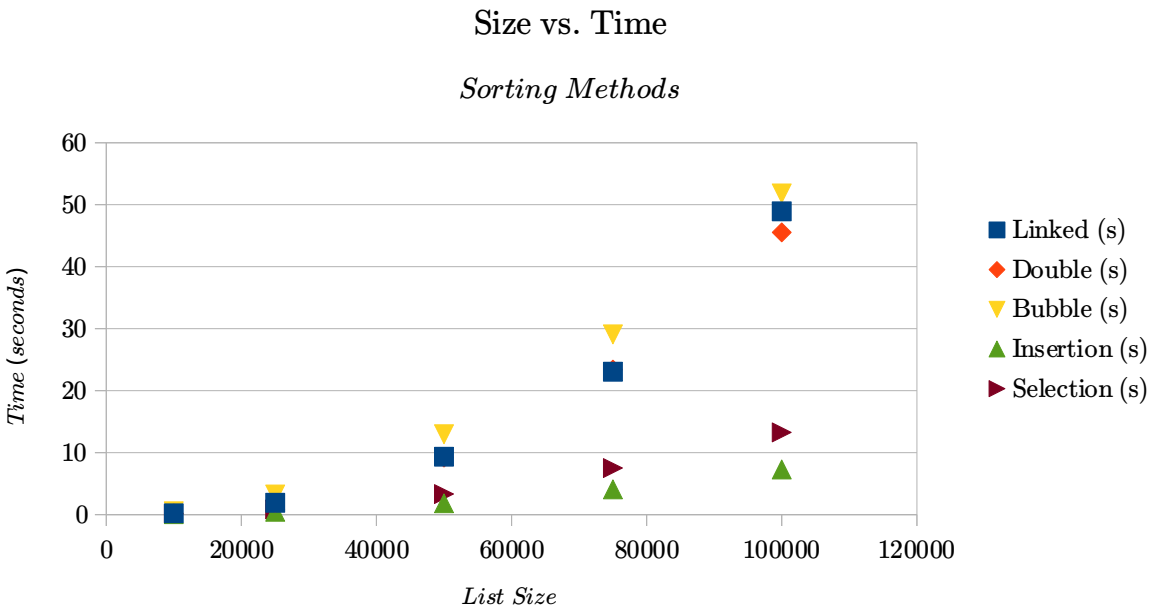

Figure 1: Sorting Times