Priyanka Kakria
MSWE 26443979

Exercise 1 Set Implementations (Report)

All experimental results are average of 100 runs, to obtain reliable statistics.

1. LinkedList (LL)

- Time-complexity is O(n) - To insert the new word, it will go through each element and add new value to end of the LL until it finds the existing word. If word exists, then searching will be stopped.
- **Insertion**- Run-time trend– each time when we want to add something new to our linked list set, we need to iterate our list from beginning to the end to check whether this new word already present, which takes linear runtime. Trend from Figure 1(a) shows the linear runtime.
- **Insertion**- Standard deviation is plotted in Figure 1(b) and its scale is larger than the scale for mean. The hypothesis for large std deviation is that system processes may be interrupting some runs causing additional run time.
- **Searching**- Searching's average time complexity is O(n) per word.

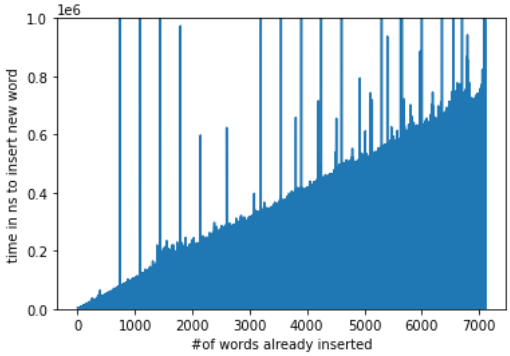

Figure 1(a) Mean

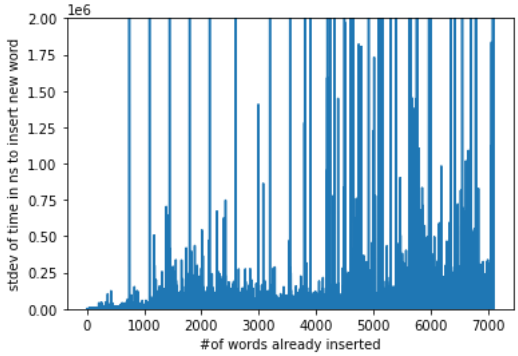

Figure 1(b) Std. deviation

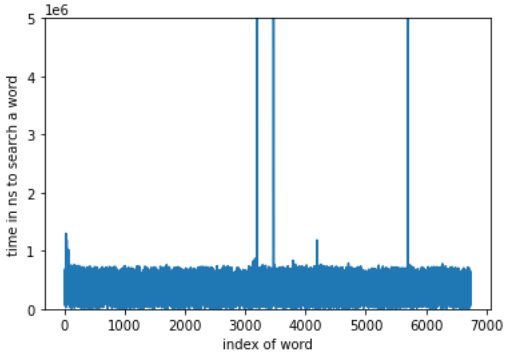

Figure 1(c) Word search time

2. Binary search tree (BST)

- Average time complexity for addition of each is 0(logn)
- **Insertion-** In BST, the trend from the Figure 2(a) is logarithmic, as x increases 7 times from 1000 words to 7000 words, y increases only about 1.8 times from ~5000ns to ~9000ns.
- **Insertion-** Standard deviation is plotted in Figure 2(b) and its scale is larger than the scale for mean. Apart from possibly system processes causing additional wait times, different length words get added at different levels of the tree and hence take different time.
- **Searching-** Searching's average time complexity is O(logn) per word. Minimum search time is taken by the word 'The', the root of the tree.

Priyanka Kakria
MSWE 26443979

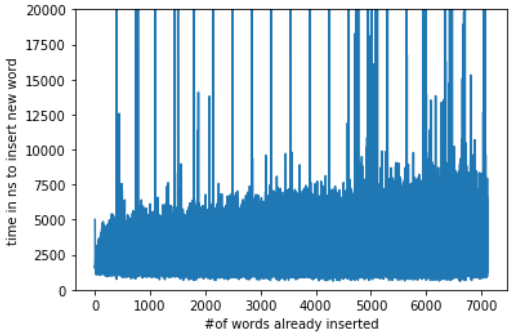

Figure 2(a). Mean

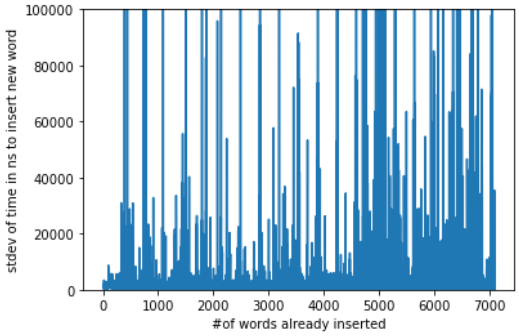

Figure 2(b). Std Dev

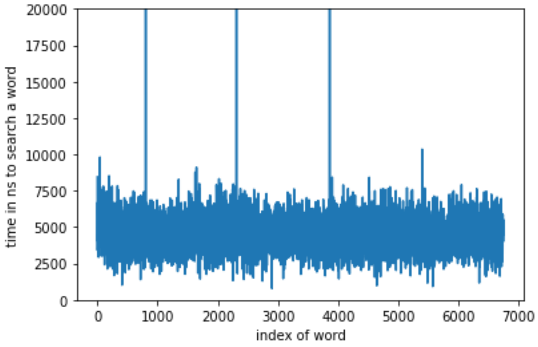

Figure 2(c) Word search time

3. HashMap

- Average time complexity of adding a word is 0(1).
- **Insertion-** Mean run-time trend shown in figure 3(a) is more or less constant per word with some jumps due to possible collisions and linear probing.
- **Insertion-** Standard deviation is shown in figure 3(b) and is again affected by system processes, and collisions.
- **Searching-** Average time complexity of searching a word is O(1).

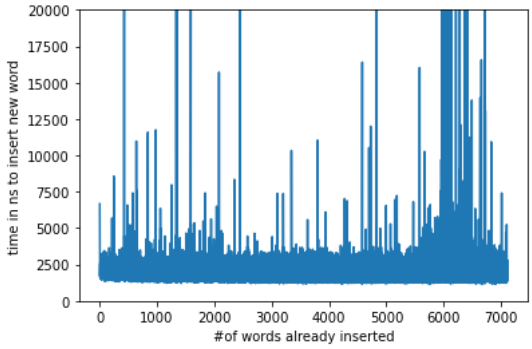

Figure 3(a) Mean

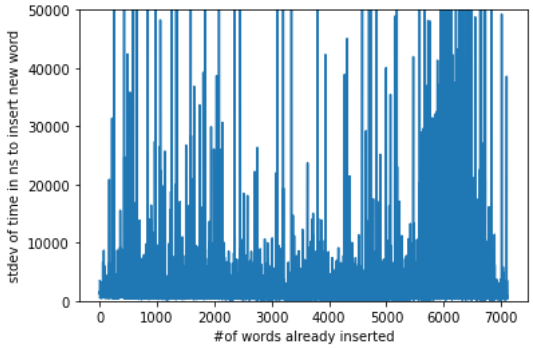

Figure 3(b) Std. dev

Priyanka Kakria
MSWE 26443979

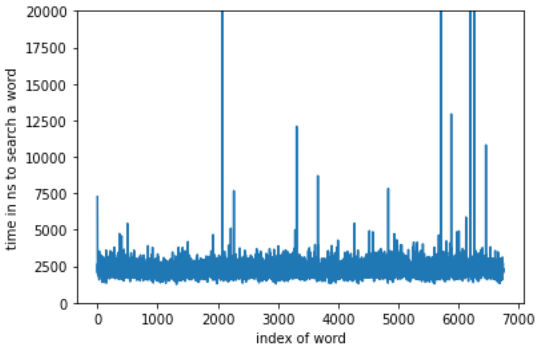

Figure 3(c): Word search time

Conclusion:

Based on average time to add all words into the set, Linked List takes the longest, followed by Tree, followed by the Hash Map.
Average searching time of Hash Map is better than Tree, both of which are significantly better than a linked list. Best case times for searching are comparable. In worst case times, Tree did the best, followed by Hash Map, followed by Linked List.

Time taken for Searching a word (after 100 repeats)

	Best time (Best word)	Avg time	Worst time
Linked list	720ns (‘The’)	5108.494us	32.011291220s
Binary tree	780ns (‘The’)	4572.14ns	35.410us
Hash table	1260ns (‘I’)	2343.26ns	138.020us