**Introduction:**

In this report, I am comparing the complexity of two distinct data structures: the Linked List and the Patricia Trie. Each structure is designed to handle data in a fundamentally different way.

**Input data structure:**

The inputs for the test cases are provided in the form of CSV files. The "Official Name Suburb" column will be used as the key for insertion and search operations in the linked list and Patricia Trie.

**Experimental Results:**

**dataset_1.csv and test1.in**

There is only one entry in the CSV file. In the in file, another key needs to be searched for additionally. I found that when searching for existing entries, since there is only one node, both data structures show the same result. When searching for non-existent entries, the linked list shows "NOTFOUND", while the Patricia trie shows the data of the only node in the tree. This is because the trie will calculate the edit distance and return the result with the closest distance.

**dataset_15.csv and test15.in**

In terms of the number of bit comparisons, the linked list is significantly more than the Patricia trie. Since the linked list needs to traverse all nodes and compare one by one, the number of node accesses and the number of string comparisons are both equal to the number of entries.

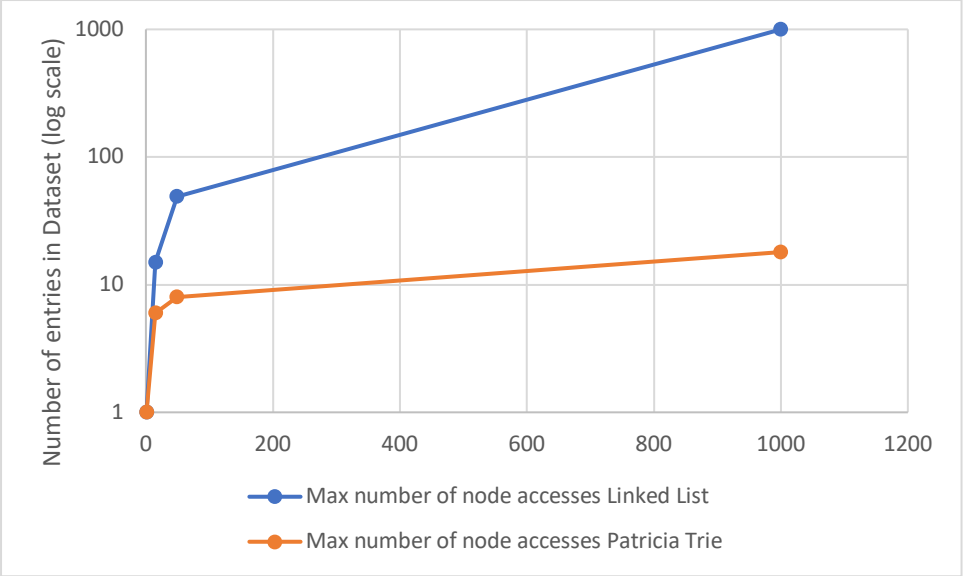
When searching in the Patricia trie, there is also a large difference in the number of bit comparisons between different keys, which indicates that the search efficiency of the Patricia tree is closely related to the shared prefixes of keys.

**dataset_1000.csv and test1000.in**

The difference in the number of comparisons for search operations of the two data structures is huge. When the number of entries is much greater, it can be clearly found that the number of comparisons used by the Patricia trie is much smaller than that of the linked list.

**Discussion and conclusion:**

| Number of entries in Dataset | Max number of node accesses Linked List | Max number of node accesses Patricia Trie |
|---|---|---|
| 1 | 1 | 1 |
| 15 | 15 | 6 |
| 49 | 49 | 8 |
| 1000 | 1000 | 18 |




It can be found that as the number of entries increases, the number of node accesses required by the linked list for searching also increases accordingly, and there is a proportional growth. This precisely reflects that the linked list search operation has a time complexity of O(n).

It can be found that as the number of entries increases, the number of node accesses required by the Patricia trie for searching does not change significantly. This indicates that the search operation of the Patricia trie has a little relationship with the number of nodes.

Logically, as the length of the keys increases, the number of bit comparisons required by the Patricia trie for searching also increases. Additionally, when searching within the Patricia tree, the number of node accesses increases as well, which is influenced by the size of the tree.

In theory, the insert operation within linked list can be done in O(1). However, searching for an element within the list has a time complexity of O(n), where n is the number of elements in the list, making it less efficient for large datasets. The Patricia Trie, on the other hand, allows for faster search operations of strings compared to linked lists, with an average time complexity of O(k), where k is the length of the string being searched (“Radix Tree,” 2024).

By comparing the results of linked lists and Patricia tries, we find that when the number of entries is small, there is little difference in performance between the two. However, as the number of entries increases, the performance advantage of the Patricia trie increases.

In conclusion, when handling search operations, especially in large data sets and with repeated prefix keys, the Patricia trie has higher efficiency and better performance than linked lists. This makes the Patricia trie a very attractive choice when dealing with string search problems. However, when choosing a data structure, it is also necessary to consider the specific scenarios and characteristics of the data, because the construction and maintenance costs of the Patricia trie may be higher than linked lists.

**Reference:**

Radix tree. (2024). In *Wikipedia*. https://en.wikipedia.org/wiki/Radix_tree