Assignment 7 - The Great Firewall of Santa Cruz Writeup

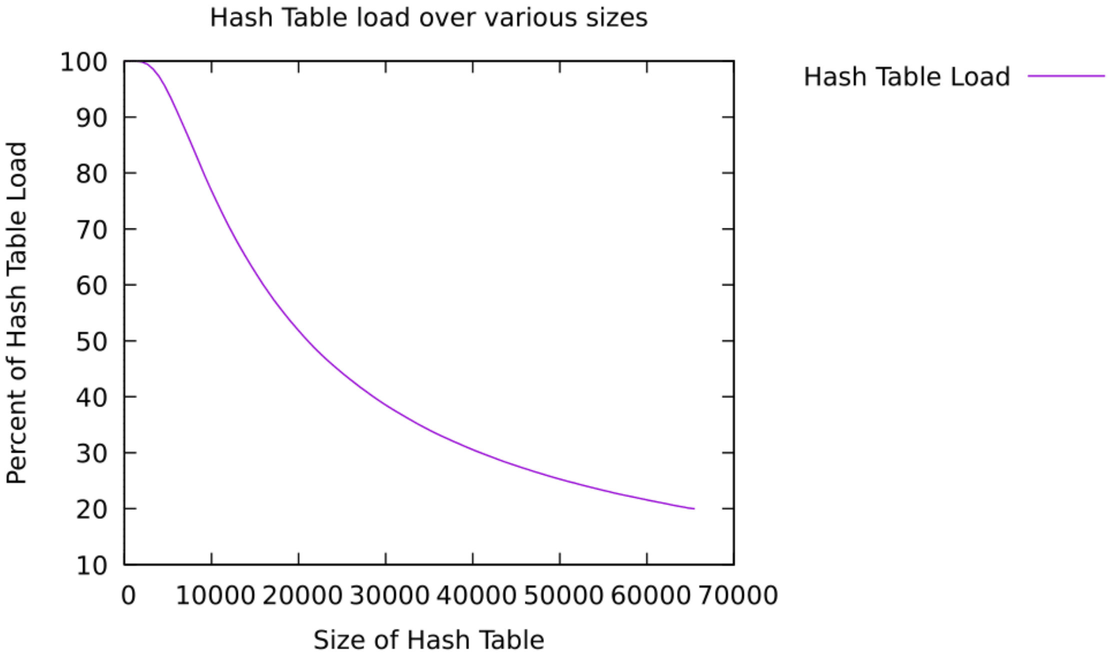
Sean Carlyle

December 5th 2021

1 Introduction

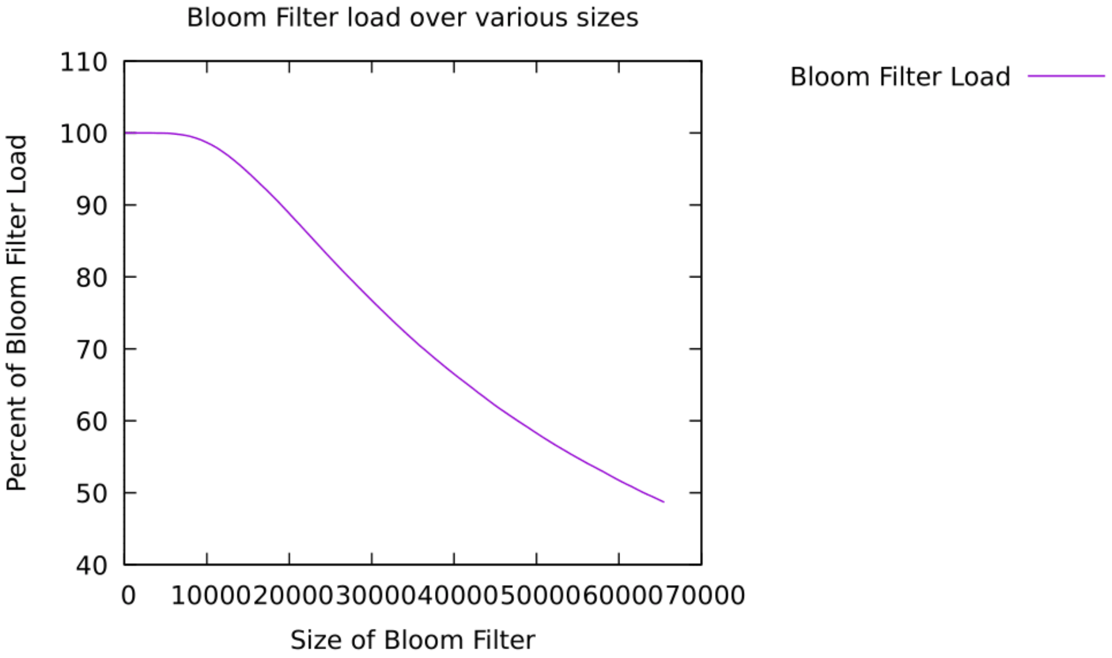
This is my Assignment 7 writeup. This writeup includes 9 graphs looking at the statistics of the banhammer program. It mainly looks at how different statistics change as the Hash table and the Bloom filter size increase. It analyzes statistics such as average binary tree traversal, average height of binary search trees, average size of binary search trees, and the percent load the Bloom filter/Hash table are under. This is what my writeup will analyze.

2 Hash Table Load



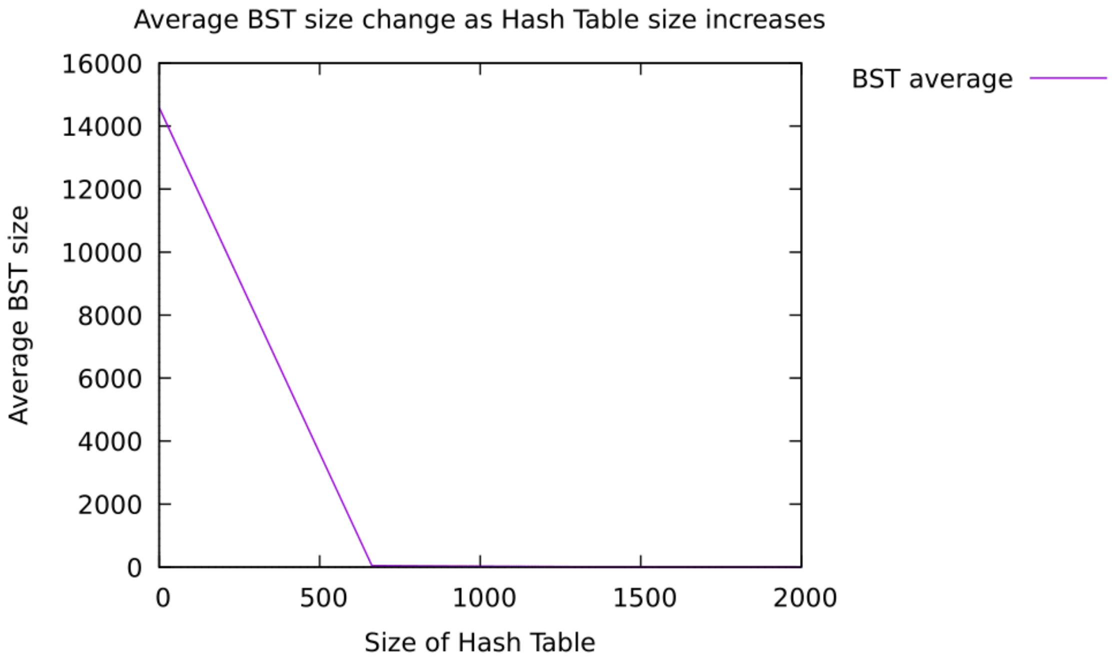
From this graph we can see that as the hash table size is greater there is less load on the hash table. This makes perfect sense because as the hash table gets bigger there are more NULL binary trees that can be used. The ratio between NULL binary trees and hash table size would get smaller and smaller as hash table size increases but NULL binary trees stays the same.

3 Bloom Filter Load



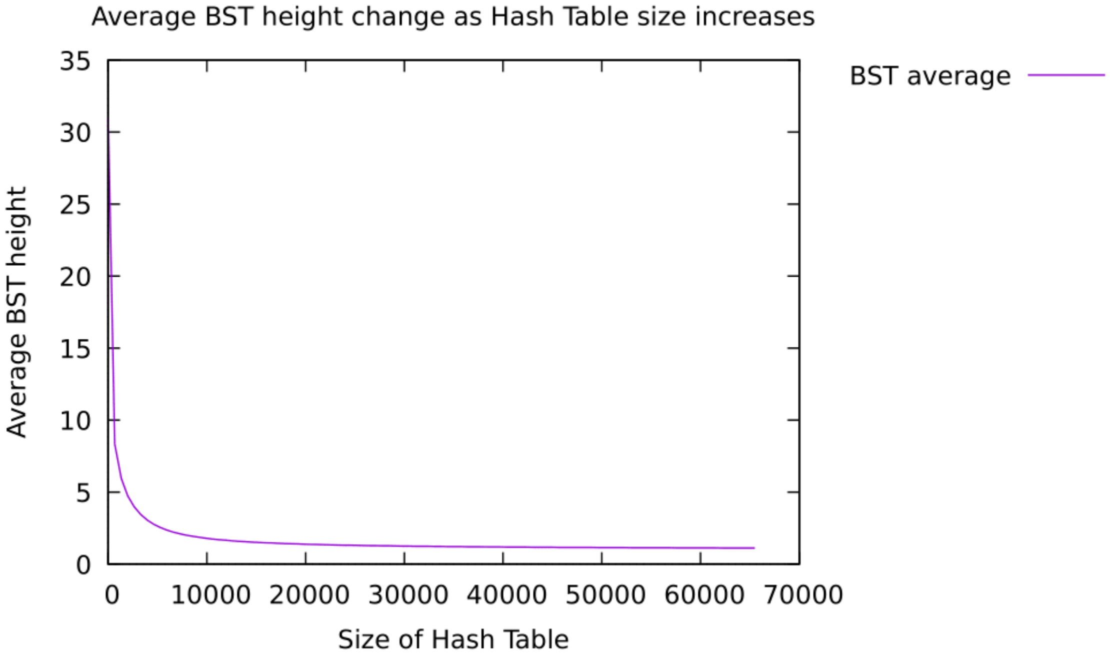
Just like with the Hash table load the Bloom filter load also gets smaller as the size of the bloom filter increases. This makes sense because with more bits that can be set with the same number of set bits, the ratio between set bits and bloom filter size would get smaller. We can also see that for every 10,000 increase in Bloom filter size the load goes down by about 10 percent.

4 Average BST size



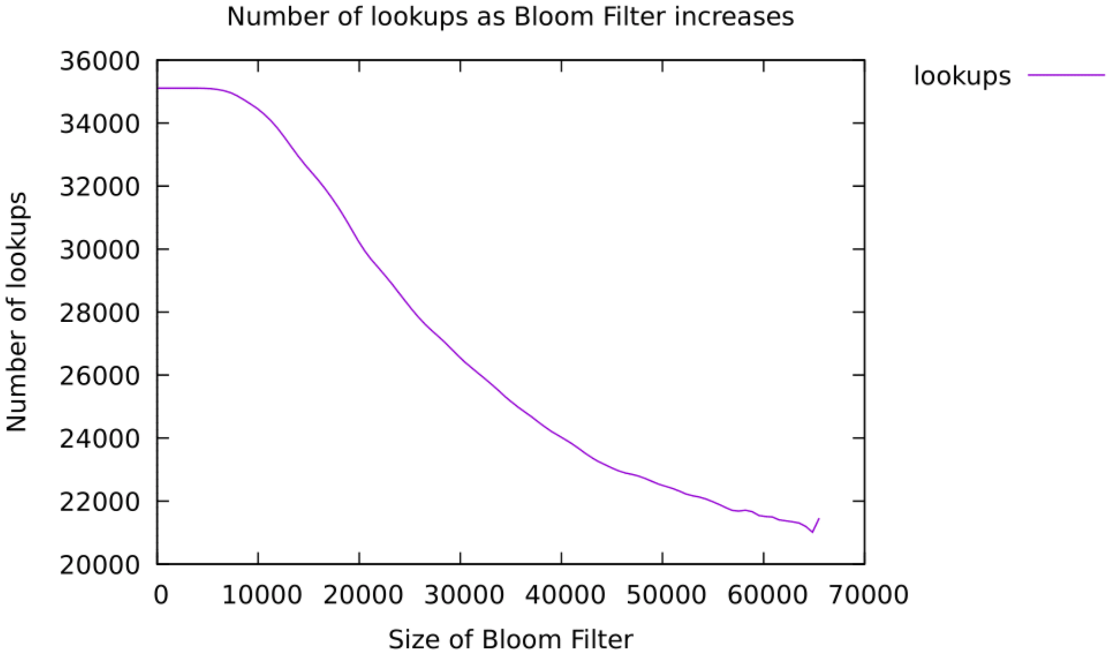
The average binary search tree size steadily decreases as the hash table size increases. Around a hash table size of 600 the average BST size reaches around 1 and decreases by almost nothing. This makes sense because as the hash table increases in size the binary search trees are required to hold less and less Nodes as the hash table size increases. Around 600 the hash table reaches a point where it can hold every badspeak/oldspeak/newspeak with a BST size of about 1 because of the huge size of the hash table.

5 Average BST height



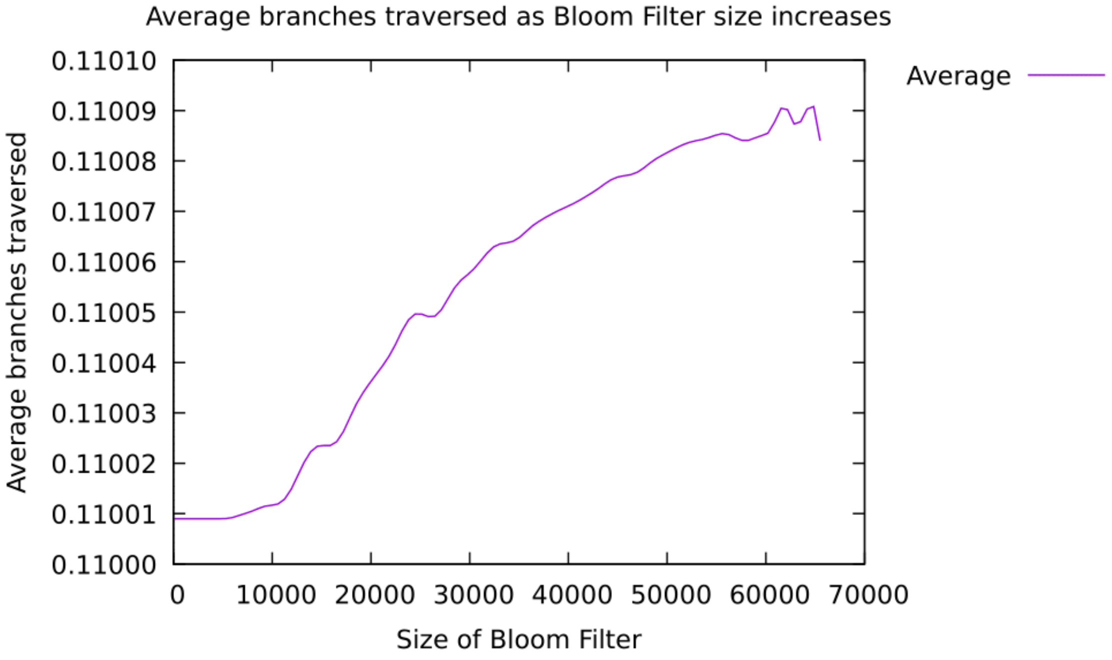
The average binary search height also decreases as the hash table size increases. It much more sharply decreases because by a hash table size of around 1000 the average height of a binary search tree doesn't need to be very tall because the words are more spread across the hash table. When the hash tables size is small all of the words must be inserted into a single BST which makes the height very tall. Eventually there are enough hash table spaces to hold words in very short binary trees. Increasing the amount of words inputted would make it take longer for the Average height to go down.

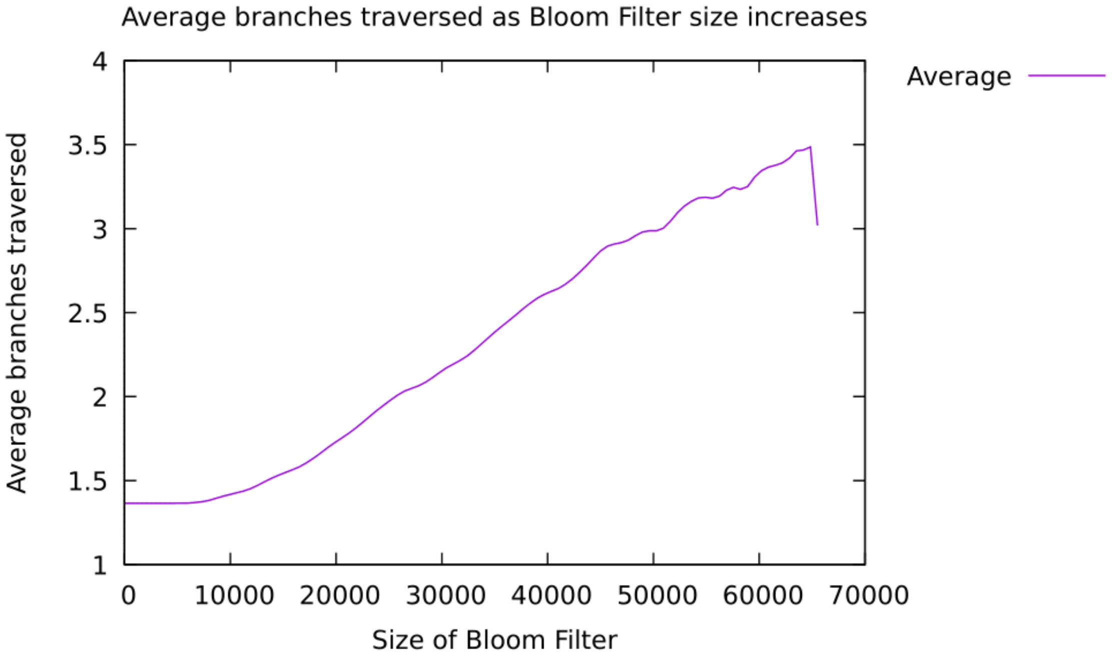
6 Bloom Filter increases affects lookups



The amount of lookups, which is the amount of times ht_insert and ht_lookup are called, decreases as the bloom filter increases. The reason that the bloom filter increasing would affect the number of lookups is that ht_insert and ht_lookup are only called once a word has been declared as in the bloom filter. But, the bloom filter can have false positives. The chance for a false positive is greater when the bloom filter is smaller. This is because there are less bits to choose in the bloom filter so many words may overlap if the bloom filter is small. As the bloom filter increases, there are less chances for false-positives and thus the number of lookups decrease as there are less words in the hash table that need to be looked up.

7 Bloom Filter Average Branch Traversal

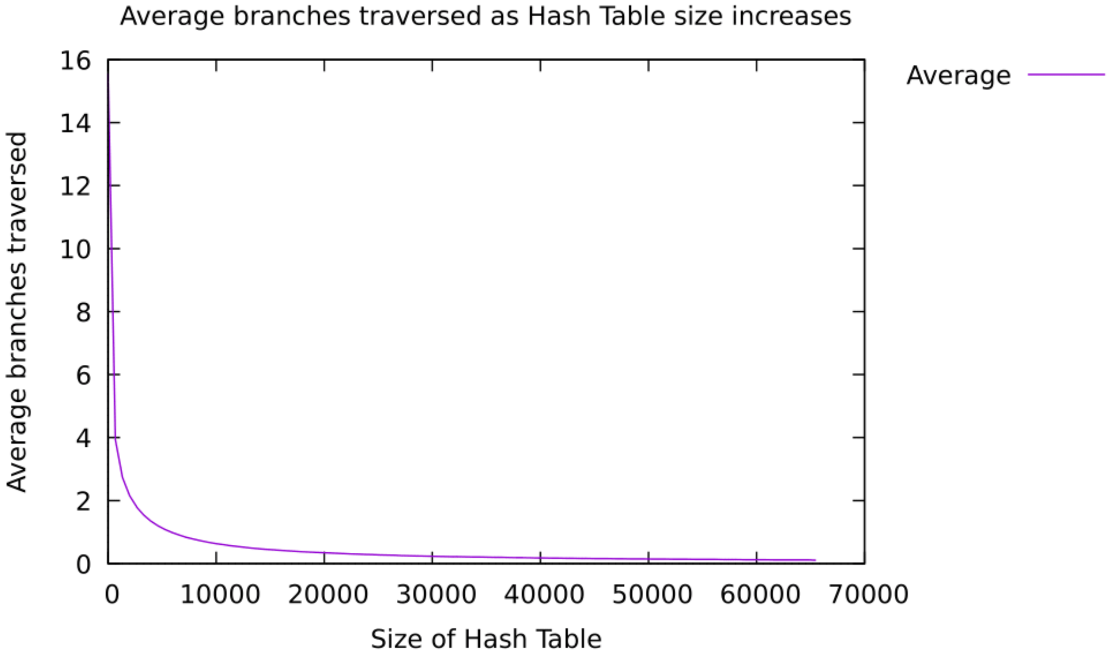


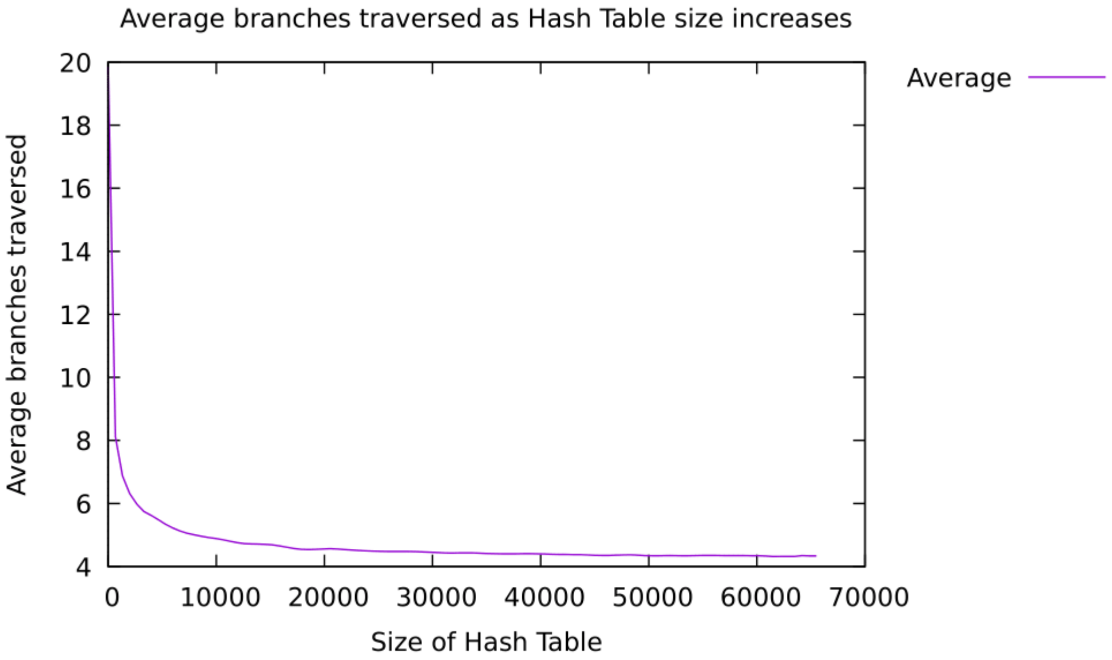


For these graphs, I decided to test on two sample sizes one small and one large. The graphs generally look the same showing that the amount of words entered do not affect

Average branch traversals as the bloom filter increases. We can see from the graphs that as the bloom filter size increases the average branches traversed increase. This makes sense because the Average branch traversal is calculated as branches divided by lookups and as the previous graph shows as the bloom filter gets bigger the amount of lookups get smaller. Thus as lookups gets smaller because it is the divisor the average number of branches traversed increases.

8 Hash Table Average Branch Traversal





For these graphs, we see that as the hash table increases the average branch traversal decreases. This makes sense because as we saw from before, as the hash table increases the height of each BST decreases. If the heights decrease in the BST this means we do not need to traverse as many branches down to find the correct words. Thus as the hash table size increases the average branch traversal decreases. We can also see that it almost exactly matches the graph of Average BST height because once BST height reaches about one, then the average branch traversed would be very small as it wouldn't have to traverse very far to find the word in the BST.

9 Conclusion

In conclusion, these graphs show us how increasing the bloom filter and hash table can affect many different statistics in our program. Mainly we learned that both the hash table load and the bloom filter load are decreased as the bloom filter/hash table sizes are increased respectively. We also learned as the hash table increases the height and size of the binary search trees decrease, because of this when the hash table is increased average branch traversal also goes down. Lastly, we learned the bloom filter decreases the number of lookups needed to find all the words in the hash table. This affects average branch traversals making them go up as the amount of lookups gets smaller. This is what the graphs tell us.