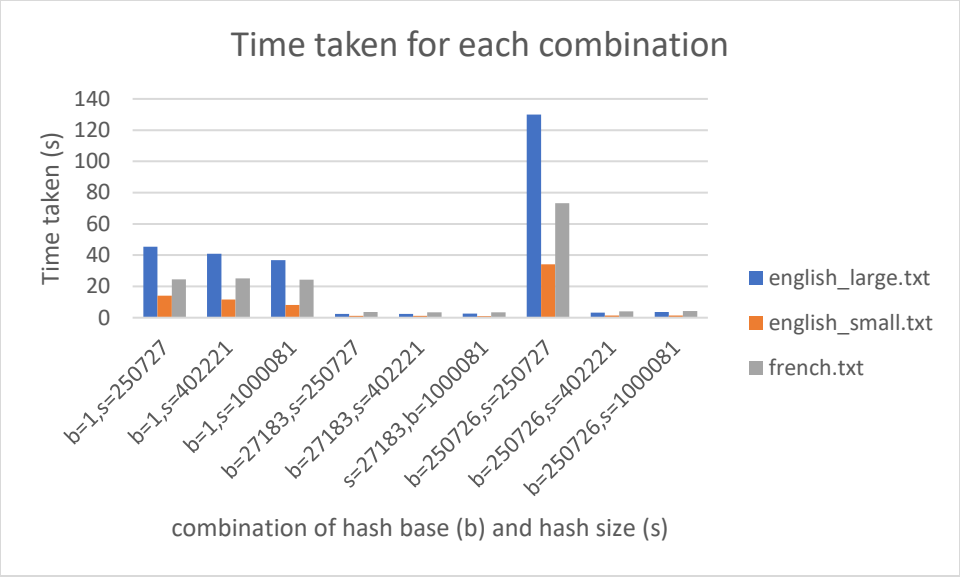
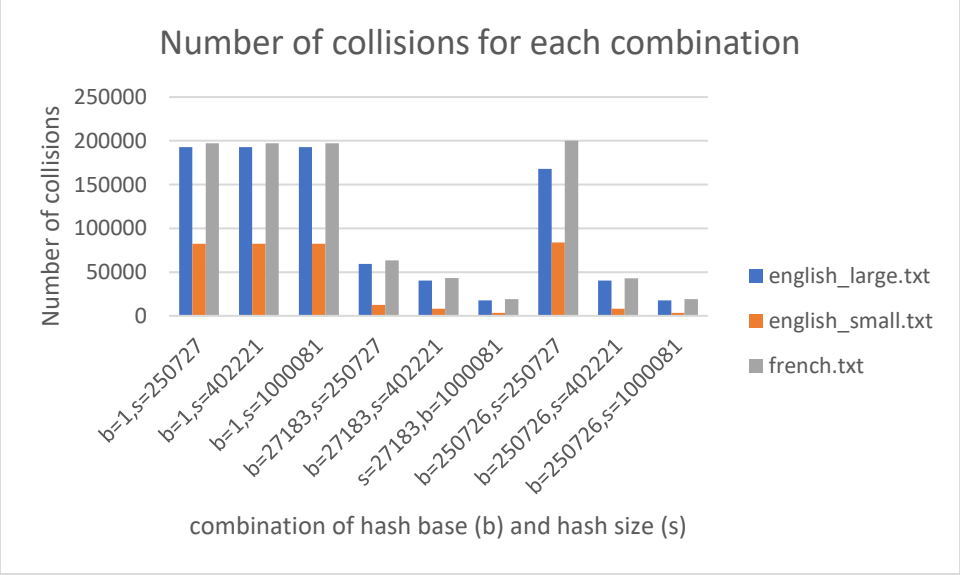
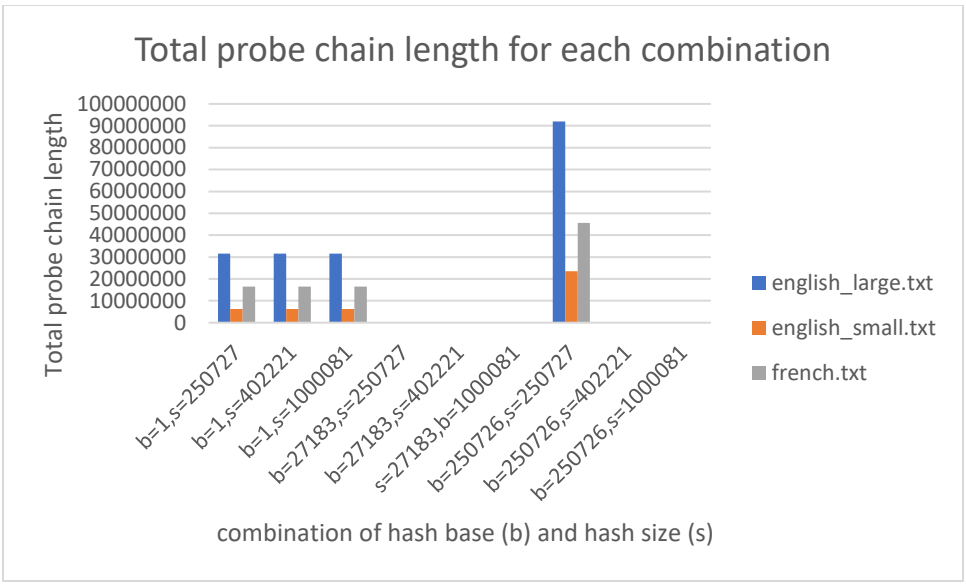
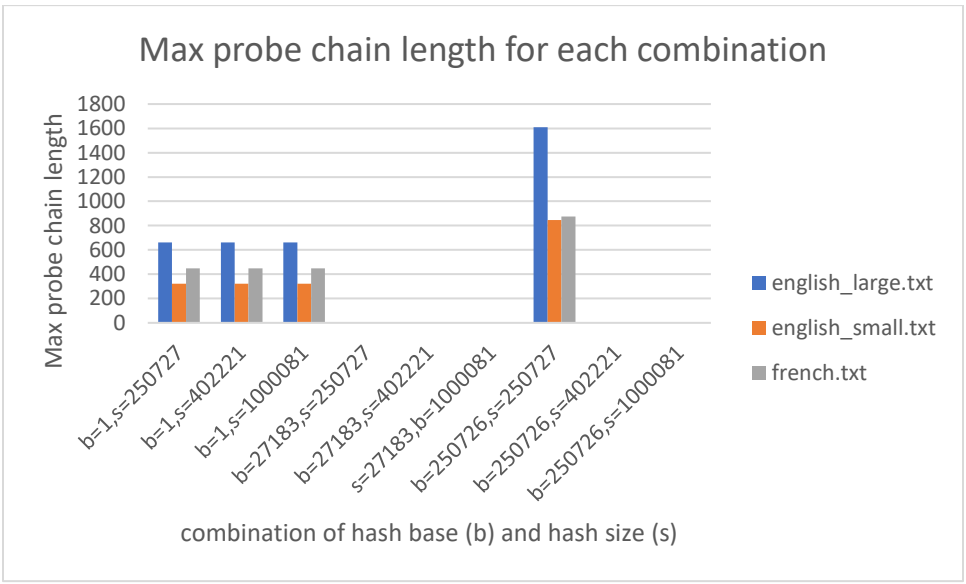
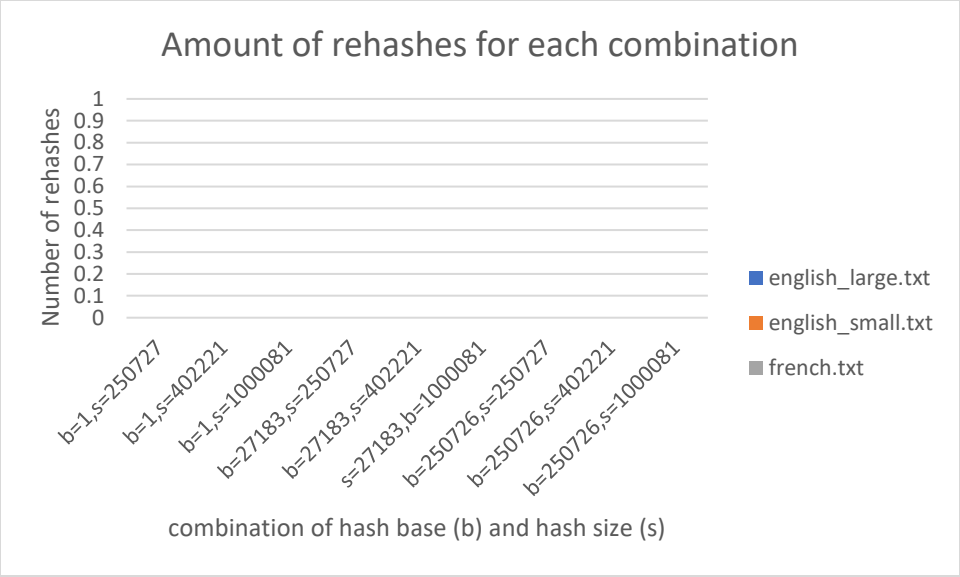

Time taken for each combination
Time taken (s)
140
120
100
80
60
40
20
0
b=1,s=250727
b=1,s=402221
b=1,s=1000081
b=27183,s=250727
b=27183,s=402221
s=27183,b=1000081
b=250726,s=250727
b=250726,s=402221
b=250726,s=1000081
combination of hash base (b) and hash size (s)
english_large.txt
english_small.txt
french.txt


Number of collisions for each combination
Number of collisions
250000
200000
150000
100000
50000
0
b=1,s=250727
b=1,s=402221
b=1,s=1000081
b=27183,s=250727
b=27183,s=402221
s=27183,b=1000081
b=250726,s=250727
b=250726,s=402221
b=250726,s=1000081
combination of hash base (b) and hash size (s)
english_large.txt
english_small.txt
french.txt


Total probe chain length for each combination
Total probe chain length
100000000
90000000
80000000
70000000
60000000
50000000
40000000
30000000
20000000
10000000
0
b=1,s=250727
b=1,s=402221
b=1,s=1000081
b=27183,s=250727
b=27183,s=402221
s=27183,b=1000081
b=250726,s=250727
b=250726,s=402221
b=250726,s=1000081
combination of hash base (b) and hash size (s)
english_large.txt
english_small.txt
french.txt


Max probe chain length for each combination
Max probe chain length
1800
1600
1400
1200
1000
800
600
400
200
0
b=1,s=250727
b=1,s=402221
b=1,s=1000081
b=27183,s=250727
b=27183,s=402221
s=27183,b=1000081
b=250726,s=250727
b=250726,s=402221
b=250726,s=1000081
combination of hash base (b) and hash size (s)
english_large.txt
english_small.txt
french.txt




Using separate chaining appears to speed up the hashing even further compared to quadratic and linear probing. In this case there is only a timeout when the base is similar to the size and it's loading from English_large.txt. The reason why this is considerably faster is because the best time complexity for inserting into a binary search tree is O(Log n) when balanced and worst time complexity O(N) which occurs when a tree is completely unbalanced. This is noticeably better than linear probing where clusters occur and may have to traverse through all data with similar keys to reach an empty slot, making its time complexity O(N), hence you're lowering the time complexity for a lot of cases.

Interestingly, the collisions for base 1 values remained extremely close to the number of collisions for quadratic probing, this shows that separate chaining doesn't make collisions less frequent but is able to resolve them much faster on average, as the probe chain lengths will be smaller.

A majority of the time taken was from retrieving statistical information such as probe lengths from the BST object.