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.