

FIT2004 Assignment 1 – Conor Christensen (27837831)

Task 1:

In task 1 each line of the file is read, stripped of its new line character, and then placed into a hash table. The hash index comes from the multiple of the ascii value for each letter in the word, then multiplied by a prime. Finally, as to stop unwanted collisions, the words are checked for identical length before assigned a place in the table, and linear probing is used to place them where necessary. The hash table looks at each word once and each letter once, making it an $O(NM)$ complexity in all cases. The hashing maintains the order of the words, thus any sorting for the output is unnecessary.

Task 2:

Task 2 uses the built hash table, and fetches all words assigned to the same index as the input key. This means the function runs in $O(M)$ time in all cases, where M is the number of characters in the query string. This is due to the hash table needing to only find the required hash index from each letter, with no string comparisons made.

Task 3:

Task 3 runs very similarly to task two, however runs in 26 extra iterations where each letter of the alphabet is then added and removed to the query, acting as a Wildcard. This means the function still runs in $O(M)$ time as the time complexity is simply multiplied by the constant 26.