**Anagram problem**

First thing, Load the dictionary file as a list of words

Let user input a word (change the user’s words to lowercase)

Create an empty list to hold and store anagrams

Sort the user’s word

Loop through each word in the word list: (change each word in list to lowercase)

If word from list and user’s word have same length then:

Sort the word from list

if word from list sorted is equal to user’s word sorted:

Append word to anagrams list

Print anagrams list

Note: If user input is a list, then loop through user list before looping through each word in the word list and create different list of anagrams for each user’s word from user’s list, and then print all the anagram lists.

Data Structures

We can use lists reason being they are mutable meaning they can be altered even after being created.

We can use maps as we use key – value pair – each user’s word and its anagram list

We can use hash tables or hash maps

HashTable

Slightly faster, as you can partition your values and do binary search to find a match quicker

HashMap

First sort each word, use the sorted word as key and then put original word in a map. The value of the map will be a list containing all the words which have same word after sorting. The Time Complexity is a bit lower.

Algorithm

We can sort and compare words of same length and exhaust all possibilities.

We use the brute force technique since it exhausts all possibilities.