

1 A Wordsearch

Given an N by N wordsearch and N words, devise an algorithm (using pseudocode or describe it in plain English) to solve the wordsearch in $O(N^3)$. For simplicity, assume no word is contained within another, i.e., if the word “**bear**” is given, “**be**” wouldn’t also be given.

If you are unfamiliar with wordsearches or want to gain some wordsearch solving intuition, see below for an example wordsearch.

Example Wordsearch:

```
S F T A T R D V R G K E V I N
A T S K L N X F I H D P X H Z
C N E D X A J Z G U N A I R U
J Y I L C V A N E S S A V P O
A B A R L K F J Q U S Y H I C
V Z U I U A T Q K D A A G R D
F S P E I D S T A A S N M Y N
W C T T S S H Q T W H N G A U
S I W H P E A A E N L I T N V
T Y I A G L D B R K E Y T Y K
A L N N J A J G E T Y A P E A
C X F I K I M U S L I T Y P R
E M U A M N T M A Z X A H U E
Y R A E K E Y W I K O Y O P N
R B C A Q J V Q I A C R O E F
```

Anirudh	Anniyat	Vanessa	Ryan
Ashley	Elaine	Isabel	
David	Stella	Karen	
Ethan	Kevin	Teresa	
Stacey	Dawn		

Hint: Add the words to a **Trie**, and you may find the **longestPrefixOf** operation helpful. Recall that **longestPrefixOf** accepts a **String key** and returns the longest prefix of **key** that exists in the **Trie**, or **null** if no prefix exists.

3 Identifying Sorts

Below you will find intermediate steps in performing various sorting algorithms on the same input list. The steps do not necessarily represent consecutive steps in the algorithm (that is, many steps are missing), but they are in the correct sequence. For each of them, select the algorithm it illustrates from among the following choices: insertion sort, selection sort, mergesort, quicksort (first element of sequence as pivot), and heapsort. When we split an odd length array in half in mergesort, assume the larger half is on the right.

Input list: 1429, 3291, 7683, 1337, 192, 594, 4242, 9001, 4392, 129, 1000

- (a) 1429, 3291, 7683, 1337, 192, 594, 4242, 9001, 4392, 129, 1000
 1429, 3291, 192, 1337, 7683, 594, 4242, 9001, 129, 1000, 4392
 192, 1337, 1429, 3291, 7683, 129, 594, 1000, 4242, 4392, 9001

- (b) 1337, 192, 594, 129, 1000, 1429, 3291, 7683, 4242, 9001, 4392
 192, 594, 129, 1000, 1337, 1429, 3291, 7683, 4242, 9001, 4392
 129, 192, 594, 1000, 1337, 1429, 3291, 4242, 4392, 7683, 9001

- (c) 1337, 1429, 3291, 7683, 192, 594, 4242, 9001, 4392, 129, 1000
 192, 1337, 1429, 3291, 7683, 594, 4242, 9001, 4392, 129, 1000
 192, 594, 1337, 1429, 3291, 7683, 4242, 9001, 4392, 129, 1000

- (d) 1429, 3291, 7683, 9001, 1000, 594, 4242, 1337, 4392, 129, 192
 7683, 4392, 4242, 3291, 1000, 594, 192, 1337, 1429, 129, 9001
 129, 4392, 4242, 3291, 1000, 594, 192, 1337, 1429, 7683, 9001

In all these cases, the final step of the algorithm will be this: 129, 192, 594, 1000, 1337, 1429, 3291, 4242, 4392, 7683, 9001