

PromptScratchpadOur Solution(s)Video Explanation

Run Code

Solution 1Solution 2Solution 3

```
1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved.
2
3 import java.util.*;
4
5 class Program {
6     // O(ns + bs) time | O(ns) space
7     public static List<Boolean> multiStringSearch(String bigString, String[] smallStrings) {
8         Trie trie = new Trie();
9         for (String smallString : smallStrings) {
10             trie.insert(smallString);
11         }
12         Set<String> containedStrings = new HashSet<String>();
13         for (int i = 0; i < bigString.length(); i++) {
14             findSmallStringsIn(bigString, i, trie, containedStrings);
15         }
16         List<Boolean> solution = new ArrayList<Boolean>();
17         for (String str : smallStrings) {
18             solution.add(containedStrings.contains(str));
19         }
20         return solution;
21     }
22
23     public static void findSmallStringsIn(
24         String str, int startIdx, Trie trie, Set<String> containedStrings) {
25         TrieNode currentNode = trie.root;
26         for (int i = startIdx; i < str.length(); i++) {
27             char currentChar = str.charAt(i);
28             if (!currentNode.children.containsKey(currentChar)) {
29                 break;
30             }
31             currentNode = currentNode.children.get(currentChar);
32             if (currentNode.children.containsKey(trie.endSymbol)) {
33                 containedStrings.add(currentNode.word);
34             }
35         }
36     }
37
38     static class TrieNode {
39         Map<Character, TrieNode> children = new HashMap<Character, TrieNode>();
40         String word;
41     }
42
43     static class Trie {
44         TrieNode root = new TrieNode();
45         char endSymbol = '*';
46
47         public void insert(String str) {
48             TrieNode node = root;
49             for (int i = 0; i < str.length(); i++) {
50                 char letter = str.charAt(i);
51                 if (!node.children.containsKey(letter)) {
52                     TrieNode newNode = new TrieNode();
53                     node.children.put(letter, newNode);
54                 }
55                 node = node.children.get(letter);
56             }
57             node.children.put(endSymbol, null);
58             node.word = str;
59         }
60     }
61 }
62
```