

PromptScratchpadOur Solution(s)Video Explanation

Run Code

Solution 1Solution 2Solution 3

```
1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved.
2
3 package main
4
5 // O(ns + bs) time | O(ns) space
6 func MultiStringSearch(bigString string, smallStrings []string) []bool {
7     trie := Trie{children: map[byte]Trie{}}
8     for _, str := range smallStrings {
9         trie.Add(str)
10    }
11    containedStrings := map[string]bool{}
12    for i := range bigString {
13        findSmallStringsIn(bigString, i, trie, containedStrings)
14    }
15    output := make([]bool, len(smallStrings))
16    for i, str := range smallStrings {
17        output[i] = containedStrings[str]
18    }
19    return output
20 }
21
22 func findSmallStringsIn(str string, startIdx int, trie Trie, containedStrings map[string]bool) {
23     current := trie
24     for i := startIdx; i < len(str); i++ {
25         currentChar := str[i]
26         if _, found := current.children[currentChar]; !found {
27             break
28         }
29         current = current.children[currentChar]
30         if end, found := current.children['*']; found {
31             containedStrings[end.word] = true
32         }
33     }
34 }
35
36 type Trie struct {
37     children map[byte]Trie
38
39     word string
40 }
41
42 func (t Trie) Add(word string) {
43     current := t
44     for i := range word {
45         letter := word[i]
46         if _, found := current.children[letter]; !found {
47             current.children[letter] = Trie{
48                 children: map[byte]Trie{},
49             }
50         }
51         current = current.children[letter]
52     }
53     current.children['*'] = Trie{
54         children: map[byte]Trie{},
55         word:     word,
56     }
57 }
58
```