

Our Solution(s)

Run Code

Your Solutions

Run Code

Solution 1

Solution 2

Solution 3

```
1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved.
2
3 package main
4
5 // O(b^2 + ns) time | O(b^2 + n) space
6 func MultiStringSearch(bigString string, smallStrings []string) []bool {
7     trie := NewTrie(bigString)
8     output := make([]bool, len(smallStrings))
9     for i, smallString := range smallStrings {
10         output[i] = trie.Contains(smallString)
11     }
12     return output
13 }
14
15 type ModifiedSuffixTrie map[byte]ModifiedSuffixTrie
16
17 func NewTrie(str string) ModifiedSuffixTrie {
18     trie := ModifiedSuffixTrie{}
19     for i := range str {
20         trie.Add(str, i)
21     }
22     return trie
23 }
24
25 func (trie ModifiedSuffixTrie) Add(str string, startIndex int) {
26     node := trie
27     for j := startIndex; j < len(str); j++ {
28         letter := str[j]
29         if _, found := node[letter]; !found {
30             node[letter] = ModifiedSuffixTrie{}
31         }
32         node = node[letter]
33     }
34 }
```

Solution 1

Solution 2

Solution 3

```
1 package main
2
3 func MultiStringSearch(bigString string, smallStrings []string) []bool {
4     // Write your code here.
5     return nil
6 }
7
```

Our Tests

Custom Output

Submit Code

1 package main

2

3 func MultiStringSearch(bigString string, smallStrings []string) []bool {

4 // Write your code here.

5 return nil

6 }

7

1

2

3

4

5

6

7

```
14 def is_palindrome(word:str) -> bool:
15     reversed = [char for char in word[::-1]]
16     return word == reversed
17
18 # Example usage:
19 word = "level"
20 is_palindrome(word)  # Returns True
21
22
23 def is_palindrome(word:str) -> bool:
24     reversed = [char for char in word[::-1]]
25     return word == reversed
26
27 # Example usage:
28 word = "level"
29 is_palindrome(word)  # Returns True
30
31
32 def is_palindrome(word:str) -> bool:
33     reversed = [char for char in word[::-1]]
34     return word == reversed
```

Run or submit code when you're ready.