

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(bns) time | O(n) space
7     public static List<Boolean> multiStringSearch(String bigString, String[] smallStrings) {
8         List<Boolean> solution = new ArrayList<Boolean>();
9         for (String smallString : smallStrings) {
10             solution.add(isInBigString(bigString, smallString));
11         }
12         return solution;
13     }
14
15     public static boolean isInBigString(String bigString, String smallString) {
16         for (int i = 0; i < bigString.length(); i++) {
17             if (i + smallString.length() > bigString.length()) {
18                 break;
19             }
20             if (isInBigString(bigString, smallString, i)) {
21                 return true;
22             }
23         }
24         return false;
25     }
26
27     public static boolean isInBigString(String bigString, String smallString, int startIdx) {
28         int leftBigIdx = startIdx;
29         int rightBigIdx = startIdx + smallString.length() - 1;
30         int leftSmallIdx = 0;
31         int rightSmallIdx = smallString.length() - 1;
32         while (leftBigIdx <= rightBigIdx) {
33             if (bigString.charAt(leftBigIdx) != smallString.charAt(leftSmallIdx)
34                 || bigString.charAt(rightBigIdx) != smallString.charAt(rightSmallIdx)) {
35                 return false;
36             }
37             leftBigIdx++;
38             rightBigIdx--;
39             leftSmallIdx++;
40             rightSmallIdx--;
41         }
42         return true;
43     }
44 }
45
```