Our Solution(s)

\_\_\_\_\_\_

Run Code

**Your Solutions** 

14рх

Run Code

```
Solution 1 Solution 2
 1 // Copyright © 2020 AlgoExpert, LLC. All rights reserved.
3 // 0(n^3 + m) time | 0(n + m) space - where n is the number of digits
4 function numbersInPi(pi, numbers) {
     const numbersTable = {};
     for (const number of numbers) {
       numbersTable[number] = true;
9
     const minSpaces = getMinSpaces(pi, numbersTable, {}, 0);
     return minSpaces === Infinity ? -1 : minSpaces;
10
11 }
12
13
   function getMinSpaces(pi, numbersTable, cache, idx) {
14
     if (idx === pi.length) return -1;
     if (idx in cache) return cache[idx];
     let minSpaces = Infinity;
16
17
     for (let i = idx; i < pi.length; i++) {</pre>
       const prefix = pi.slice(idx, i + 1);
18
19
       if (prefix in numbersTable) {
20
         const minSpacesInSuffix = getMinSpaces(pi, numbersTable, cache,
21
         minSpaces = Math.min(minSpaces, minSpacesInSuffix + 1);
22
23
24
     cache[idx] = minSpaces;
25
     return cache[idx];
26 }
27
28 exports.numbersInPi = numbersInPi;
```

```
Solution 1 Solution 2
                        Solution 3
1 function numbersInPi(pi, numbers) {
    // Write your code here.
5 // Do not edit the line below.
6 exports.numbersInPi = numbersInPi;
```

Run or submit code when you're ready.

\_\_\_\_\_