Our Solution(s) — Run Code

Solution 1   Solution 2

```python
# Copyright © 2020 AlgoExpert, LLC. All rights reserved.

# O(n^3 + m) time | O(n + m) space - where n is the number of digits
def numbersInPi(pi, numbers):
    numbersTable = {number: True for number in numbers}
    minSpaces = getMinSpaces(pi, numbersTable, {}, 0)
    return -1 if minSpaces == float("inf") else minSpaces


def getMinSpaces(pi, numbersTable, cache, idx):
    if idx == len(pi):
        return -1
    if idx in cache:
        return cache[idx]
    minSpaces = float("inf")
    for i in range(idx, len(pi)):
        prefix = pi[idx : i + 1]
        if prefix in numbersTable:
            minSpacesInSuffix = getMinSpaces(pi, numbersTable, cache,
            minSpaces = min(minSpaces, minSpacesInSuffix + 1)
    cache[idx] = minSpaces
    return cache[idx]
```

Your Solutions — Run Code

Solution 1   Solution 2   Solution 3

```python
def numbersInPi(pi, numbers):
    # Write your code here.
    pass
```

Our Tests

Custom Output — Submit Code