Our Solution(s) Run Code

Solution 1 Solution 2

```go
// Copyright © 2020 AlgoExpert, LLC. All rights reserved.

package main

import "math"

// O(n^3 + m) time | O(n + m) space - where n is the number of digits
// in Pi and m is the number of favorite numbers.
func NumbersInPi(pi string, numbers []string) int {
  numbersTable := map[string]bool{}
  for _, number := range numbers {
    numbersTable[number] = true
  }

  cache := map[int]int{}
  for i := len(pi) - 1; i >= 0; i-- {
    getMinSpaces(pi, numbersTable, cache, i)
  }

  if cache[0] == math.MaxInt32 {
    return -1
  }
  return cache[0]
}

func getMinSpaces(pi string, numbersTable map[string]bool,
  cache map[int]int, idx int) int {
  if idx == len(pi) {
    return -1
  } else if val, found := cache[idx]; found {
    return val
  }
  minSpaces := math.MaxInt32
```

Your Solutions Run Code

Solution 1 Solution 2 Solution 3

```go
package main

func NumbersInPi(pi string, numbers []string) int {
  // Write your code here.
  return -1
}

```

Our Tests

Custom Output Submit Code

Run or submit code when you're ready.