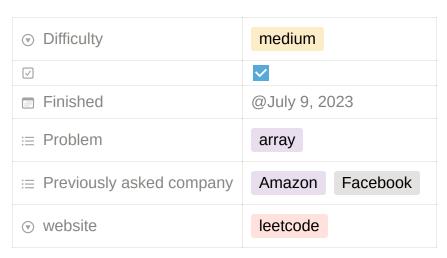
# 974. Subarray Sums Divisible by K



### **Question:**

Given an integer array nums and an integer k, return the number of non-empty subarrays that have a sum divisible by k.

A **subarray** is a **contiguous** part of an array.

#### **Example 1:**

```
Input: nums = [4,5,0,-2,-3,1], k = 5
Output: 7
Explanation: There are 7 subarrays with a sum divisible by k = 5:
[4, 5, 0, -2, -3, 1], [5], [5, 0], [5, 0, -2, -3], [0], [0, -2, -3], [-2, -3]
```

## **Example 2:**

```
Input: nums = [5], k = 9
Output: 0
```

# **Optimal solution:**

Time complexity: O(n)

Space complexity: O(k) k varies for every input but its constant so its O(1)

```
class Solution(object):
    def subarraysDivByK(self, nums, k):
        dict = {}
        sum = 0
        dict[0] = 1
        res = 0
        for i in nums:
            sum += i
            temp = sum % k
        if temp in dict:
            res += dict[temp]
            dict[temp] += 1
        else:
            dict[temp] = 1
        return res
```