

<https://course.acciojob.com/idle?question=6b0355db-2e09-4afa-8be4-045d710113fb>

● EASY

● Max Score: 30 Points

-
-
-
-
-
-
-
-
-

Subarray sum divisible by k

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

A subarray is a contiguous part of an array.

Input Format

Input consists of two lines.

The first line contains two integers `n` (size of the array) and `k`.

The second line contains `n` spaced integers.

Output Format

Print count of subarrays divisible by `k`

Example 1

Input

```
6 5
4 5 0 -2 -3 1
```

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

```
4 2
4 5 0 -2
```

Output

4

Explanation

There are 4 subarrays with a sum divisible by $k = 2$:

[4]

[0]

[0 -2]

[-2]

Constraints

$1 \leq \text{nums.length} \leq 5 * 10^3$

$-10^4 \leq \text{nums}[i] \leq 10^4$

$2 \leq k \leq 10^4$

Topic Tags

- Hashing
- Arrays

My code

// in java

```
import java.io.*;
```

```
import java.util.*;
```

```
public class Main {
```

```
    public static void main(String[] args) throws Exception {
```

```
Scanner scn = new Scanner(System.in);
int n = scn.nextInt();
int m = scn.nextInt();
int c=0;
int[] arr = new int[n];
for (int i = 0; i < n; i++) {
    arr[i] = scn.nextInt();
}

for (int i = 0; i < arr.length; i++) {
    int sum=0;
    for (int j = i; j < arr.length; j++) {

        sum+=arr[j];

        int r=sum%m;
        if(r==0)
            c++;
    }
}
System.out.print(c);
}

}
```