GeeksforGeeks

A computer science portal for geeks

Count all possible groups of size 2 or 3 that have sum as multiple of 3

Given an unsorted integer (positive values only) array of size ‘n’, we can form a group of two or three, the group should be such that the sum of all elements in that group is a multiple of 3. Count all possible number of groups that can be generated in this way.

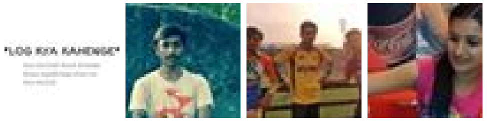
```
Input: arr[] = {3, 6, 7, 2, 9}
Output: 8
// Groups are {3,6}, {3,9}, {9,6}, {7,2}, {3,6,9},
//            {3,7,2}, {7,2,6}, {7,2,9}


Input: arr[] = {2, 1, 3, 4}
Output: 4
// Groups are {2,1}, {2,4}, {2,1,3}, {2,4,3}
```

We strongly recommend to minimize the browser and try this yourself first.

The idea is to see remainder of every element when divided by 3. A set of elements can form a group only if sun of their remainders is multiple of 3. Since the task is to enumerate groups, we count all elements with different remainders.

```
1. Hash all elements in a count array based on remainder, i.e,
   for all elements a[i], do c[a[i]%3]++;
2. Now c[0] contains the number of elements which when divided
   by 3 leave remainder 0 and similarly c[1] for remainder 1
   and c[2] for 2.
```



Interview Experiences
Advanced Data Structures
Dynamic Programming
Greedy Algorithms
Backtracking
Pattern Searching
Divide & Conquer
Mathematical Algorithms
Recursion
Geometric Algorithms

```
3. Now for group of 2, we have 2 possibilities
   a. 2 elements of remainder 0 group. Such possibilities are
      c[0]*(c[0]-1)/2
   b. 1 element of remainder 1 and 1 from remainder 2 group
      Such groups are c[1]*c[2].
4. Now for group of 3,we have 4 possibilities
   a. 3 elements from remainder group 0.
      No. of such groups are c[0]C3
   b. 3 elements from remainder group 1.
      No. of such groups are c[1]C3
   c. 3 elements from remainder group 2.
      No. of such groups are c[2]C3
   d. 1 element from each of 3 groups.
      No. of such groups are c[0]*c[1]*c[2].
5. Add all the groups in steps 3 and 4 to obtain the result.
```

```
#include<stdio.h>
```

```
// Returns count of all possible groups that can be formed from elemen
// of a[].
int findgroups(int arr[], int n)
{
    // Create an array C[3] to store counts of elements with remainder
    // 0, 1 and 2.  c[i] would store count of elements with remainder
    int c[3] = {0}, i;

    int res = 0; // To store the result

    // Count elements with remainder 0, 1 and 2
    for (i=0; i<n; i++)
        c[arr[i]%3]++;

    // Case 3.a: Count groups of size 2 from 0 remainder elements
    res += ((c[0]*(c[0]-1))>>1);

    // Case 3.b: Count groups of size 2 with one element with 1
    // remainder and other with 2 remainder
    res += c[1] * c[2];

    // Case 4.a: Count groups of size 3 with all 0 remainder elements
    res += (c[0] * (c[0]-1) * (c[0]-2))/6;
```

Geometric Algorithms

Popular Posts

- All permutations of a given string
- Memory Layout of C Programs
- Understanding “extern” keyword in C
- Median of two sorted arrays
- Tree traversal without recursion and without stack!
- Structure Member Alignment, Padding and Data Packing
- Intersection point of two Linked Lists
- Lowest Common Ancestor in a BST.
- Check if a binary tree is BST or not
- Sorted Linked List to Balanced BST

```
    // Case 4.b: Count groups of size 3 with all 1 remainder elements
    res += (c[1] * (c[1]-1) * (c[1]-2))/6;

    // Case 4.c: Count groups of size 3 with all 2 remainder elements
    res += ((c[2]*(c[2]-1)*(c[2]-2))/6);

    // Case 4.c: Count groups of size 3 with different remainders
    res += c[0]*c[1]*c[2];

    // Return total count stored in res
    return res;
}

// Driver program to test above functions
int main()
{
    int arr[] = {3, 6, 7, 2, 9};
    int n = sizeof(arr)/sizeof(arr[0]);
    printf("Required number of groups are %d\n", findgroups(arr,n));
    return 0;
}
```

Output:

```
Required number of groups are 8
```

Time Complexity: O(n)
Auxiliary Space: O(1)

This article is contributed by Amit Jain. Please write comments if you find anything incorrect, or you want to share more information about the topic discussed above

Related Tpoics:

- Remove minimum elements from either side such that 2*min becomes more than max
- Divide and Conquer | Set 6 (Search in a Row-wise and Column-wise Sorted 2D Array)
- Bucket Sort
- Kth smallest element in a row-wise and column-wise sorted 2D array | Set 1
- Find the number of zeroes
- Find if there is a subarray with 0 sum
- Divide and Conquer | Set 5 (Strassen's Matrix Multiplication)
- Sort n numbers in range from 0 to n^2 – 1 in linear time

7 Tweet 3 1

Writing code in comment? Please use **ideone.com** and share the link here.

8 Comments GeeksforGeeks

Sort by Newest ▾

705 Subscribe

Recent Comments

kzs please provide solution for the problem...

Backtracking | Set 2 (Rat in a Maze) · 3 minutes ago

Sanjay Agarwal bool tree::Root_to_leaf_path_given_sum(tree...

Root to leaf path sum equal to a given number · 28 minutes ago

GOPI GOPINATH @admin Highlight this sentence "We can easily...

Count trailing zeroes in factorial of a number · 30 minutes ago

newCoder3006 If the array contains negative numbers also. We...

Find subarray with given sum · 54 minutes ago

newCoder3006 Code without using while loop. We can do it...

Find subarray with given sum · 1 hour ago

Sanjay Agarwal You can also use the this method:...

Join the discussion…

AlienOnEarth · 8 days ago

@GeeksforGeeks Please address the issue for duplicates. Or the input is to

^ | ˅ · Reply · Share ›

Sidharth · 25 days ago

Could anyone tell me why is the right shift operator used ?

^ | ˅ · Reply · Share ›

Jumper ➦ Sidharth · 16 days ago

>> 1 is equivalent to dividing by 2. Similarly <<1 is equivalent to multipl

^ | ˅ · Reply · Share ›

Srini · a month ago

For the input arr

{3, 6, 7, 2, 9}

Is the following is not the possible group?

{3, 6, 7, 2}

why was this missed?

^ | ˅ · Reply · Share ›

Alien ➦ Srini · a month ago

Question is to make group of 2-3 members not more than that

^ | ˅ · Reply · Share ›

zzer · a month ago

there lack some boundary checks, such as check if c[1]>=2 and c[0]>=2 whe

Count trailing zeroes in factorial of a number · 1 hour ago

^ | v · Reply · Share ›

Guest · 2 months ago

The question is to count all possible groups , not just the unique ones

1 ^ | v · Reply · Share ›

Kshitij Gupta · 2 months ago

There is an assumption that the numbers in list are unique.

For a list like: {3, 6, 7, 2, 9, 9}

The code would print: 15 (treating both 9 as different)

Whereas the actual unique groups are only 11:

(6, 3), (9, 3), (9, 6), (9, 9), (7, 2), (9, 9, 6), (7, 6, 2), (9, 6, 3), (9, 9, 3), (9, 7, 2),

9 ^ | v · Reply · Share ›

Subscribe   Add Disqus to your site

Contact Us!

Powered by **WordPress** & **MooTools**, customized by geeksforgeeks team