**Count Pairs Odd Xor**

Show Topic Tags   

Given an array Arr[] of N integers. Write a program to find out number of pairs in array whose XOR is odd.  
  
**Input:**  
First line of input contains a single integer T which denotes the number of test cases. Then T test cases follows. First line of each test case contains a single integer N which denotes the size of array. Second line of each test case contains N space separated integers which denotes the elements of the array.  
  
**Output:**  
For each test case print the number of pairs in array whose XOR is odd.  
  
**Constraints:**  
1<=T<=100  
1<=N<=1000  
1<=Arr[i]<=1000  
  
**Example:  
Input:**  
2  
3  
1 2 3  
4  
1 2 3 4  
**Output:**  
2  
4

\*\*For More Examples Use Expected Output\*\*

Contributor: Harsh Agarwal

<http://practice.geeksforgeeks.org/problems/count-pairs-odd-xor/0>

/\*

\* To change this template, choose Tools | Templates

\* and open the template in the editor.

\*/

package javaapplication250;

import java.io.\*;

import java.math.\*;

import java.util.\*;

/\*\*

\*

\* @author Administrador

\*/

public class JavaApplication250 {

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

// TODO code application logic here

BufferedReader br = new BufferedReader(new InputStreamReader(System.in));

int t = Integer.parseInt(br.readLine());

while(t-- > 0) {

int n = Integer.parseInt(br.readLine());

String[] input = br.readLine().trim().split(" ");

int[] arr = new int[n];

for(int i =0; i<n; i++) {

arr[i] = Integer.parseInt(input[i]);

}

int cont =0;

for(int i =0; i<n-1; i++) {

for(int j=i+1; j<n; j++) {

if( (arr[i]^ arr[j])%2 !=0 ) {

cont++;

}

}

}

System.out.println(cont);

}

}

}

**Efficient Approach:**We can observe that:

odd ^ odd = even

odd ^ even = odd

even ^ odd = odd

even ^ even = even

Therefore total pairs in array whose XOR is odd will be equal to count of odd numbers multiplied by count of even numbers.

* C++
* Java

|  |
| --- |
| // C++ program to count pairs in array  // whose XOR is odd  #include <iostream>  using namespace std;    // A function will return number of pair  //  whose XOR is odd  int countXorPair(int arr[], int n)  {      // To store count of odd and even      // numbers      int odd = 0, even = 0;        for (int i = 0; i < n; i++)      {          // Increase even if number is          // even otherwise increase odd          if (arr[i] % 2 == 0)              even++;          else              odd++;      }        // Return number of pairs      return odd \* even;  }    // Driver program to test countXorPair()  int main()  {      int arr[] = { 1, 2, 3 };      int n = sizeof(arr) / sizeof(arr[0]);      cout << countXorPair(arr, n);        return 0;  } |

Run on IDE

Output:

2

Time Complexity : O(n)

This article is contributed by [**nuclode**](http://www.facebook.com/nuclode). If you like GeeksforGeeks and would like to contribute, you can also write an article using [contribute.geeksforgeeks.org](http://www.contribute.geeksforgeeks.org/)or mail your article to contribute@geeksforgeeks.org. See your article appearing on the GeeksforGeeks main page and help other Geeks.

Please write comments if you find anything incorrect, or you want to share more information about the topic discussed above.