Naveej is from a tribe that speaks some weird language - their alphabet consists of N distinct characters. He has an array A=[A1,A2,…,AN], where Ai denotes the number of occurrences of the i-th character with him.

## He wants to make a palindromic string using all the characters he has (every character he has must be used in this string).

In order to make this possible, he can perform the following operation:

Select an i (1≤i≤N) and convert all occurrences of i-th alphabet to any other alphabet of his choice.  
Note that Naveej just wants to be able to make any palindrome, as long as every character is used. For example, if N=2 and A=[2,2] and we consider the characters to be a and b, he can make both abba and baab, but aba is not allowed because it uses only 3 characters.

Find the minimum number of operations required such that Naveej can make a palindromic string using all the characters he has. It can be proven that there always exists at least one sequence of operations allowing for the formation of a palindrome.

**Input Format**  
The first line of input contains a single integer T denoting the number of test cases. The description of T test cases follows.  
The first line of each test case contains a single integer N - the size of the alphabet.  
The second line contains N space-separated integers: A1,A2,...,AN, where Ai is the number of occurrences of the i-th character with Naveej.  
**Output Format**  
For each test case, output a single line containing one integer - the minimum number of operations required so that Naveej can make a palindromic string using all the characters he has.

**Constraints**  
1≤T≤1000  
1≤N≤2⋅10^5  
1≤Ai≤10^9  
It is guaranteed that the sum of N over all test cases does not exceed 2⋅10^5  
Subtasks  
Subtask 1 (100 points): Original constraints  
**Sample Input 1**  
2  
1  
4  
3  
4 3 1  
**Sample Output 1**  
0  
1  
**Explanation**  
In the first test case, N=1. Let the character be a. We can make the following palindromic string: aaaa.

In the second test case, N=3. Let the characters be a, b, c. It is initially not possible to make a palindrome with the given occurrences of the characters. We perform 1 operation: Convert all the occurrences of b to c. Then, we can make the following palindromic string: acaccaca.

Java code

import java.util.\*;

class WeiredPalidromeMAkin

{

public static void main(String[] args)

{

Scanner sc=new Scanner(System.in);

int T=sc.nextInt();

while(T>0)

{

int c=0;

int n=sc.nextInt();

int[] a=new int[n];

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

{

a[i]=sc.nextInt();

}

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

{

if(a[i]%2!=0)

c++;

}

System.out.println(c/2);

T--;

}

}

}