Greatest Common Divisor

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

There is an array of length n, containing only positive numbers.

Now you can add all numbers by 1 many times. Please find out the minimum times you need to perform to obtain an array whose greatest common divisor(gcd) is larger than 1 or state that it is impossible.

You should notice that if you want to add one number by 1, you need to add all numbers by 1 at the same time.

Input

The first line of input file contains an integer T ($1 \leq T \leq 20$), describing the number of test cases.

Then there are $2 \times T$ lines, with every two lines representing a test case.

The first line of each case contains a single integer n ($1 \leq n \leq 10^5$) described above.

The second line of that contains n integers ranging in $[1, 10^9]$.

Output

You should output exactly T lines.

For each test case, print **Case** d**:** (d represents the order of the test case) first. Then output exactly one integer representing the answer. If it is impossible, print **-1** instead.

Example

standard input	standard output
3 1 2 5 2 5 9 5 7 5 3 5 7 9 11	Case 1: 0 Case 2: -1 Case 3: 1

Note

- Sample 1: You do not need to do anything because its gcd is already larger than 1.
- Sample 2: It is impossible to obtain that array.
- Sample 3: You just need to add all number by 1 so that gcd of this array is 2.