Defective Script

Time limit: 2 seconds
Memory limit: 1024 megabytes

Devin is a system administrator at a tech company that manages a network of n servers arranged in a ring topology. Each server is handling a certain amount of computational load, represented by a non-negative integer a_i, where i ranges from 1 to n.

To optimize the network performance and ensure fairness, Devin wants to equalize the load across all servers, making each server handle the same amount of load. He aims to maximize this equal load as much as possible.

Devin has developed a script to reduce the load on any server. When he runs the script on server i, it is supposed to decrease the load on that server by 2 units (down to a minimum of zero). However, due to a known bug in the script, every time it's executed on server i, it inadvertently removes an additional 1 unit of load from the previous server in the network (server $i - 1$). If $i = 1$, the previous server is server n (since the servers form a ring).

Devin can run this buggy script any number of times (including zero), each time choosing any server to run it on. He can run the script on a server even if its current load is less than 2 units, or if the load of the previous server is zero (in both cases the load goes to zero).

Help Devin determine the maximum possible equal load that can be achieved on all servers using his script.

Input

Each test contains multiple test cases. The first line contains the number of test cases t ($1 \le t \le 10^4$). The description of the test cases follows.

The first line of each test case contains a single integer n, denoting the number of servers ($2 \le n \le 2 \cdot 10^5$).

The second line contains n integers $a_1, a_2, \ldots, a_n$, denoting the amounts of load the servers are handling ($0 \le a_i \le 10^9$).

It is guaranteed that the sum of n over all test cases does not exceed $2 \cdot 10^5$.

Output

For each test case, print the maximum possible equal load that can be achieved on all servers.

Example

standard input	standard output

5
4
9 9 6 8
2
3 5
9
9 9 8 2 4 4 3 5 3
3
777 777 777
6
0 1 0 1 0 1
	5
	1
	0
	777
	0

Note

In the first test case, Devin can run the script once on server 1, twice on server 2, and once on server 4. As a result, each server will be handling 5 units of load.