Turtle vs. Rabbit Race: Optimal Trainings

Input file:	`standard input`
Output file:	`standard output`
Time limit:	5 seconds
Memory limit:	256 megabytes

Isaac begins his training. There are n running tracks available, and the i-th track ($1 \le i \le n$) consists of a_i equal-length sections.

Given an integer u ($1 \le u \le 10^9$), finishing each section can increase Isaac's ability by a certain value, described as follows:

- Finishing the 1-st section increases Isaac's performance by u.
- Finishing the 2-nd section increases Isaac's performance by $u - 1$.
- Finishing the 3-rd section increases Isaac's performance by $u - 2$.
- $\ldots$
- Finishing the k-th section ($k \ge 1$) increases Isaac's performance by $u + 1 - k$. (The value $u + 1 - k$ can be negative, which means finishing an extra section decreases Isaac's performance.)

You are also given an integer l. You must choose an integer r such that $l \le r \le n$ and Isaac will finish **each** section of **each** track $l, l+1, \ldots, r$ (that is, a total of $\sum_{i=l}^{r} a_i = a_l + a_{l+1} + \ldots + a_r$ sections).

Answer the following question: what is the optimal r you can choose that the increase in Isaac's performance is maximum possible?

If there are multiple r that maximize the increase in Isaac's performance, output the **smallest** r.

To increase the difficulty, you need to answer the question for q different values of l and u.

Input

The first line of input contains a single integer t ($1 \le t \le 10^4$) — the number of test cases.

The descriptions of the test cases follow.

The first line contains a single integer n ($1 \le n \le 10^5$).

The second line contains n integers $a_1, a_2, \ldots, a_n$ ($1 \le a_i \le 10^4$).

The third line contains a single integer q ($1 \le q \le 10^5$).

The next q lines each contain two integers l and u ($1 \le l \le n, 1 \le u \le 10^9$) — the descriptions to each query.

The sum of n over all test cases does not exceed $2 \cdot 10^5$. The sum of q over all test cases does not exceed $2 \cdot 10^5$.

Output

For each test case, output q integers: the i-th integer contains the optimal r for the i-th query. If there are multiple solutions, output the **smallest** one.

Example

standard input	standard output

5
6
3 1 4 1 5 9
3
1 8
2 7
5 9
1
10
1
1 1
9
5 10 9 6 8 3 10 7 3
5
8 56
1 12
9 3
1 27
5 45
5
7 9 2 5 2
10
1 37
2 9
3 33
4 32
4 15
2 2
4 2
2 19
3 7
2 7
10
9 1 6 7 6 3 10 7 3 10
5
10 43
3 23
9 3
6 8
5 14
	3 4 5
	1
	9 2 9 4 9
	5 2 5 5 5 2 4 5 4 2
	10 6 9 7 7

Note

For the 1-st query in the first test case:

- By choosing $r = 3$, Isaac finishes $a_1 + a_2 + a_3 = 3 + 1 + 4 = 8$ sections in total, hence his increase in performance is $u + (u - 1) + \ldots + (u - 7) = 8 + 7 + 6 + 5 + 4 + 3 + 2 + 1 = 36$.
- By choosing $r = 4$, Isaac finishes $a_1 + a_2 + a_3 + a_4 = 3 + 1 + 4 + 1 = 9$ sections in total, hence his increase in performance is $u + (u - 1) + \ldots + (u - 8) = 8 + 7 + 6 + 5 + 4 + 3 + 2 + 1 + 0 = 36$.

Both choices yield the optimal increase in performance, however we want to choose the **smallest** r. So we choose $r = 3$.

For the 2-nd query in the first test case, by choosing $r = 4$, Isaac finishes $a_2 + a_3 + a_4 = 1 + 4 + 1 = 6$ sections in total, hence his increase in performance is $u + (u - 1) + \ldots + (u - 5) = 7 + 6 + 5 + 4 + 3 + 2 = 27$. This is the optimal increase in performance.

For the 3-rd query in the first test case:

- By choosing $r = 5$, Isaac finishes $a_5 = 5$ sections in total, hence his increase in performance is $u + (u - 1) + \ldots + (u - 4) = 9 + 8 + 7 + 6 + 5 = 35$.
- By choosing $r = 6$, Isaac finishes $a_5 + a_6 = 5 + 9 = 14$ sections in total, hence his increase in performance is $u + (u - 1) + \ldots + (u - 13) = 9 + 8 + 7 + 6 + 5 + 4 + 3 + 2 + 1 + 0 + (-1) + (-2) + (-3) + (-4) = 35$.

Both choices yield the optimal increase in performance, however we want to choose the **smallest** r. So we choose $r = 5$.

Hence the output for the first test case is $[3, 4, 5]$.