



A giant library has just been inaugurated this week. It can be modeled as a **sequence of N consecutive shelves** with **each shelf having some number of books**. Now, think of the following two queries which can be performed on these shelves.

- Change the number of books in one of the shelves.
- Obtain the number of books on the shelf having the k^{th} rank within the range of shelves.

A shelf is said to have the k^{th} rank if its position is k when the shelves are sorted based on the number of the books they contain, in ascending order. Can you write a program to simulate the above queries?

Input Format

The first line contains a single integer T , denoting the number of test cases.

The first line of each test case contains an integer N denoting the number of shelves in the library.

The next line contains N space separated integers where the i^{th} integer represents the number of books on the i^{th} shelf where $1 \leq i \leq N$.

The next line contains an integer Q denoting the number of queries to be performed. Q lines follow with each line representing a query.

Queries can be of two types:

- $1 \ x \ k$ - Update the number of books in the x^{th} shelf to k ($1 \leq x \leq N$).
- $0 \ x \ y \ k$ - Find the number of books on the shelf between the shelves x and y (both inclusive) with the k^{th} rank ($1 \leq x \leq y \leq N$, $1 \leq k \leq y-x+1$).

Output Format

For every test case, output the results of the queries in a new line.

Sample Input

```
2
2
1 2
2
0 1 2 1
0 1 2 2
4
4 3 2 1
4
0 1 1 1
1 1 1
0 1 1 1
0 1 4 3
```

Sample Output

```
1
2
4
1
2
```

You are required to submit the following:

A **zip** file which is named with your INDEX NUMBER (ex: "**18020013.zip**")

The zip file should contain,

a) Project documentation (Include the used algorithm/pseudocode and the test cases.

Do not include the implementation in the document)

b) Soft copy of your system