

Problem C: Classroom

Advanced Algorithms for Programming Contests

Restrictions

Time: 2 seconds

Memory: 512 MB

Problem description

Due to massive budget cuts in his state's education budget, high school teacher Mr. Gregarius is teaching an enormously large class this term. Since in such a setting it is generally pretty hard for *any* teacher to keep the class quiet and get everyone's attention, he always wanders around the classroom aimlessly (to at least be *seen* by most students during a lesson), frequently poses a difficult question and, seemingly at random, picks a nearby student to answer it.

But of course he doesn't *actually* want to pick students at random! Over the course of the first few lessons he numerated the students $1, \dots, N$ and successfully determined each student's *general performance index* (GPI), which is a nonnegative integer that is greater, the better a student performs. Now whenever he has posed a question, he only restricts himself to a certain range of currently nearby students, then tracks down the lowest performing one in that range and asks him to answer it.

Since the class is so large, Mr. Gregarius is having trouble keeping everyone's GPI in mind, therefore he asked you to write him a little program that continuously runs throughout the term, keeps track of changes in the class structure and quickly tells him for any given range of students which is the lowest performing one in that range. Specifically, the program should be able to process queries of the following kinds:

- **QUESTION $l\ r$**
signaling that he just asked a question and now wants to know the index of the lowest performing student within the range l, \dots, r of students
- **DROPOUT i**
signaling that the student with index i dropped out of school and will no longer be present to answer any questions

Input

The input consists of

- one line containing N ($3 \leq N \leq 10^5$) – the number of students in the class (in the beginning of the program's execution, before anyone has dropped out)
- one line containing the GPIs g_1, \dots, g_N ($0 \leq g_i \leq 10^8$, g_i is the GPI of the student with index i). It is guaranteed that no two students share a GPI and that Mr. Gregarius only picks ranges that still include at least one student.
- many lines each containing a query of one of the types described above. The program is supposed to terminate when given the query **DROPOUT** -1 . It is guaranteed that there will be no more than 10^5 queries in a single execution of the program.

Output

For every query of the **QUESTION** type, output the requested student index on a separate line.

Sample input and output

Input	Output
5	2
42 7 19 13 0	5
QUESTION 1 3	4
QUESTION 4 5	
DROPOUT 5	
QUESTION 3 5	
DROPOUT -1	