

## 스택 수열



시간 제한	메모리 제한	제출	정답	맞은 사람	정답 비율
2 초	128 MB	31900	9628	7163	31.211%

## 문제

스택 (stack)은 기본적인 자료구조 중 하나로, 컴퓨터 프로그램을 작성할 때 자주 이용되는 개념이다. 스택은 자료를 넣는 (push) 입구와 자료를 뽑는 (pop) 입구가 같아 제일 나중에 들어간 자료가 제일 먼저 나오는 (LIFO, Last in First out) 특성을 가지고 있다.

1부터  $n$ 까지의 수를 스택에 넣었다가 뽑아 늘어놓음으로써, 하나의 수열을 만들 수 있다. 이때, 스택에 push하는 순서는 반드시 오름차순을 지키도록 한다고 하자. 임의의 수열이 주어졌을 때 스택을 이용해 그 수열을 만들 수 있는지 없는지, 있다면 어떤 순서로 push와 pop 연산을 수행해야 하는지를 알아낼 수 있다. 이를 계산하는 프로그램을 작성하라.

## 입력

첫 줄에  $n$  ( $1 \leq n \leq 100,000$ )이 주어진다. 둘째 줄부터  $n$ 개의 줄에는 수열을 이루는 1이상  $n$ 이하의 정수가 하나씩 순서대로 주어진다. 물론 같은 정수가 두 번 나오는 일은 없다.

理

입력된 수열을 만들기 위해 필요한 연산을 한 줄에 한 개씩 출력한다. push 연산은 +로, pop 연산은 -로 표현하도록 한다. 불가능한 경우 NO를 출력한다.

예제 입력 1 복사

8  
4  
3  
6  
8  
7  
5  
2  
1

8  
7  
6  
5  
4  
3  
2  
1

+++ -- ++ - - - -

## 예제 출력 1 복사

- stack ਤੋਂ ਹਟਾਓ, ਤੁਸੀਂ ਕਰਦੇ

⇒ head < input

→ input이 들어갈 때까지  
push

⇒ head == input

→ pap

## 예제 입력 2 복사

5  
1  
2  
5  
3  
4

$\begin{array}{|c|} \hline 5 \\ \hline 4 \\ \hline 3 \\ \hline 2 \\ \hline 1 \\ \hline \end{array}$

$+-+-++-$

$\hookrightarrow \text{no}$

## 예제 출력 2 복사

힌트

1부터  $n$ 까지에 수에 대해 차례로 [push, push, push, push, pop, pop, push, push, pop, push, push, pop, pop, pop, pop] 연산을 수행하면 수열 [4, 3, 6, 8, 7, 5, 2, 1]을 얻을 수 있다.

## 프린터 큐

시간 제한	메모리 제한	제출	정답	맞은 사람	정답 비율
2 초	128 MB	16634	8551	6823	53.852%

### 문제

여러분도 알다시피 여러분의 프린터 기기는 여러분이 인쇄하고자 하는 문서를 인쇄 명령을 받은 '순서대로', 즉 먼저 요청된 것을 먼저 인쇄한다. 여러 개의 문서가 쌓인다면 Queue 자료구조에 쌓여서 FIFO - First In First Out - 에 따라 인쇄가 되게 된다. 하지만 상근이는 새로운 프린터기 내부 소프트웨어를 개발하였는데, 이 프린터기는 다음과 같은 조건에 따라 인쇄를 하게 된다.

1. 현재 Queue의 가장 앞에 있는 문서의 '중요도'를 확인한다.
2. 나머지 문서들 중 현재 문서보다 중요도가 높은 문서가 하나라도 있다면, 이 문서를 인쇄하지 않고 Queue의 가장 뒤에 재배치 한다. 그렇지 않다면 바로 인쇄를 한다.

예를 들어 Queue에 4개의 문서(A B C D)가 있고, 중요도가 2 1 4 3 라면 C를 인쇄하고, 다음으로 D를 인쇄하고 A, B를 인쇄하게 된다.

여러분이 할 일은, 현재 Queue에 있는 문서의 수와 중요도가 주어졌을 때, 어떤 한 문서가 몇 번째로 인쇄되는지 알아내는 것이다. 예를 들어 위의 예에서 C문서는 1번째로, A 문서는 3번째로 인쇄되게 된다.

### 입력

첫 줄에 test case의 수가 주어진다. 각 test case에 대해서 문서의 수  $N$ (100이하)와 몇 번째로 인쇄되었는지 궁금한 문서가 현재 Queue의 어떤 위치에 있는지를 알려주는  $M$ (0이상  $N$ 미만)이 주어진다. 다음줄에  $N$ 개 문서의 중요도가 주어지는데, 중요도는 1 이상 9 이하이다. 중요도가 같은 문서가 여러 개 있을 수도 있다. 위의 예는  $N=4$ ,  $M=0$ (A 문서가 궁금하다면), 중요도는 2 1 4 3이 된다.

### 출력

각 test case에 대해 문서가 몇 번째로 인쇄되는지 출력한다.

#### 예제 입력 1 복사

```
3
1 0
5
4 2
1 2 3 4
6 0
1 1 9 1 1 1
```

4 2 3 4  
1 2 3 4

#### 예제 출력 1 복사

```
1
2
5
```

ABCD  
2143  
ABCD  
C D A B

queue 활용

⇒ enumerate 함수를 통해  
doc, priority를 같이 저장한다.

## 키로거

☆ 한국어 ▾

시간 제한	메모리 제한	제출	정답	맞은 사람	정답 비율
1 초	256 MB	16139	4020	2534	23.502%

### 문제

창영이는 강산이의 비밀번호를 훔치기 위해서 강산이가 사용하는 컴퓨터에 키로거를 설치했다. 며칠을 기다린 끝에 창영이는 강산이가 비밀번호 창에 입력하는 글자를 얻어냈다.

키로거는 사용자가 키보드를 누른 명령을 모두 기록한다. 따라서, 강산이가 비밀번호를 입력할 때, 화살표나 백스페이스를 입력해도 정확한 비밀번호를 알아낼 수 있다.

강산이가 비밀번호 창에서 입력한 키가 주어졌을 때, 강산이의 비밀번호를 알아내는 프로그램을 작성하시오.

### 입력

첫째 줄에 테스트 케이스의 개수가 주어진다. 각 테스트 케이스는 한 줄로 이루어져 있고, 강산이가 입력한 순서대로 길이가 L인 문자열이 주어진다. ( $1 \leq L$ 의 길이  $\leq 1,000,000$ ) 강산이가 백스페이스를 입력했다면, '.'가 주어진다. 이때 커서의 바로 앞에 글자가 존재한다면, 그 글자를 지운다. 화살표의 입력은 '<'와 '>'로 주어진다. 이때는 커서의 위치를 움직일 수 있다면, 왼쪽 또는 오른쪽으로 1만큼 움직인다. 나머지 문자는 비밀번호의 일부이다. 물론, 나중에 백스페이스를 통해서 지울 수는 있다. 만약 커서의 위치가 줄의 마지막이 아니라면, 그 문자를 입력하고, 커서는 오른쪽으로 한 칸 이동한다.

### 출력

각 테스트 케이스에 대해서, 강산이의 비밀번호를 출력한다. 비밀번호의 길이는 항상 0보다 크다.

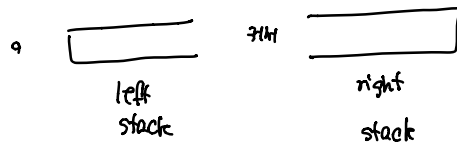
#### 예제 입력 1 복사

```
2
<<BP<A>>Cd-
ThIsIsS3Cr3t
```

#### 예제 출력 1 복사

```
BAPC
ThIsIsS3Cr3t
```

↳ 배열로 이해  $\Rightarrow$  시간 초과



✱

- not in : 시간 초과 ↑
- insert : 시간 초과

↳ append / pop 활용하는 방법