

Monotone Stack.

Monotone Stack

Monotone stack은 스택을 '오름차순'으로 유지하는 기법이다. 일단 어떤 형식으로 돌아가는지 알아보자.

{1, 3, 7, 5, 10}가 있다고 생각해 보자
이 수를 스택에 넣는다고 하면

①③⑦ - 오름차순이니 그냥 push!

①③⑤ - 여기서 중요! 5는 7보다 작으니 pop을 해주고 5를 넣는다.

①③⑤⑩ - 이런식으로 ...

이게 모하...? 라는 의문이 들 수 있으나, 상당히 powerful한 방식이다.
 $O(n^2) \rightarrow O(n)$ 으로 --!

아직 외팔지 않으니 문제를 보자.

시간 제한	메모리 제한	제한	입력	출력 제한	제한
1 초	512 MB	785	278	215	38.66%
문제					
크기가 N인 수열 $A = A_1, A_2, \dots, A_N$ 이 있다. 수열의 각 원소 A_i 에 대해서 오름차순 NGE(N)를 구하려고 한다. A_i 의 오름차순 NGE(N)는 A_i 보다 큰 수 중에서 가장 왼쪽에 있는 수를 의미한다. 그러한 수가 없는 경우에는 오름차순 -1이다.					
예를 들어, $A = [3, 5, 2, 7]$ 인 경우 $NGE(1) = 5$, $NGE(2) = 7$, $NGE(3) = 7$, $NGE(4) = -1$ 이다. $A = [9, 5, 4, 8]$ 인 경우 $NGE(1) = -1$, $NGE(2) = 8$, $NGE(3) = 8$, $NGE(4) = -1$ 이다.					
입력					
첫째 줄에 수열 A의 크기 N ($1 \leq N \leq 1,000,000$)이 주어진다. 둘째 줄에 수열 A의 원소 A_1, A_2, \dots, A_N ($1 \leq A_i \leq 1,000,000$)가 주어진다.					
출력					
출력 줄에 수 $NGE(1), NGE(2), \dots, NGE(N)$ 를 공백으로 구분해 출력한다.					
예제 입력 1 보이기					
4					
3 5 2 7					
예제 출력 1 보이기					
5 7 7 -1					

모약하면, 자신의 오른쪽 수 중에 큰 수 중에 가장 왼쪽의 수를 출력하는 것이다.

필자와 대원원 사람들은 '어? 오른쪽 탐색해서 큰 수 찾으면 출력하면 되지 않아?' 할 수 있다.

```
for (int i=0; i<V.size(); i++) {
    for (int j=i; j<V.size(); j++) {
        if (V[i] < V[j]) {
            // 정답
        }
    }
}
```

간단한 구조이지만.. 시간제한 1초,
 $V.size()$ 는 1,000,000인데
저 코드는 $O(n^2)$ 이니 먹을 법 없다.

여기서 monotone stack을 이용하면
1. $s.top() == -1$ 이 될 때까지 pop하거나
 $s.top() < V[i]$ 가 될 때까지 pop한다.
2. $s.top()$ 를 출력해준다.
3. $s.push(V[i])$ 를 해준다. (오름차순이
이해가 안가는데?)

4
3 5 2 7
stack ④