

S I N C E 2 0 0 7



로그인하세요.

sign in sign up

뉴스 피드

포럼

뉴스

자유게시판

질문과 답변

과거 게시판

위키

페이지 목록

온라인 저지

문제 풀기

랜덤 문제 고르기

최근 제출된 답안

사용자 랭킹

튜토리얼

캘린더

알고스팟 대화방

초대장 받기

이용 안내

검색하기

AOJ 문제 바로가기

다가오는 이벤트들

Hacker Cup 2018 Round 3

(8/19 02:00)

see all



삽입 정렬 뒤집기

문제 정보

문제 ID	시간 제한	메모리 제한	제출 횟수	정답 횟수 (비율)
INSERTION	2000ms	65536kb	1170	544 (46%)
출제자	출처	분류		
JongMan	알고리즘 문제 해결 전략	보기		

문제

유명한 정렬 알고리즘인 삽입 정렬은 정렬된 부분 배열을 유지하며 이 배열에 새 원소를 삽입해 나가는 식으로 동작합니다. 주어진 정수 배열 A를 정렬하는 삽입 정렬의 구현은 다음과 같습니다.

```
void insertionSort(vector<int>& A) {
    for(int i = 0; i < A.size(); ++i) {
        // A[0..i-1] 에 A[i] 를 끼워넣는다
        int j = i;
        while(j > 0 && A[j-1] > A[j]) {
            // 불변식 a: A[j+1..i] 의 모든 원소는 A[j] 보다 크다.
            // 불변식 b: A[0..i] 구간은 A[j] 를 제외하면 정렬되어 있다.
            swap(A[j-1], A[j]);
            --j;
        }
    }
}
```

삽입 정렬은 A[0..i-1] 이 정렬된 배열일 때, A[i] 를 적절한 위치를 만날 때까지 왼쪽으로 한칸씩 움직입니다. 예를 들어 A={5,1,4,3,2} 의 삽입 정렬은 다음과 같이 이루어집니다.

A	비고
5 1 4 3 2	초기 상태
1 5 4 3 2	1을 왼쪽으로 1칸 옮김
1 4 5 3 2	4를 왼쪽으로 1칸 옮김
1 3 4 5 2	3을 왼쪽으로 2칸 옮김
1 2 3 4 5	2를 왼쪽으로 3칸 옮김

1부터 N까지의 자연수가 한 번씩 포함된 길이 N 인 수열 A[] 를 삽입 정렬했습니다. 원래 수열은 알 수 없지만, 그 과정에서 각 원소가 왼쪽으로 몇 칸이나 이동했는지를 알고 있습니다. 예를 들어, 위 예제에서 각 위치에 있는 값들이 움직인 칸수를 표현하면 {0,1,1,2,3} 이 됩니다. 이 때 원래 수열을 찾아내는 프로그램을 작성하세요.

입력

입력의 첫 줄에는 테스트 케이스의 수 C (1 ≤ C ≤ 50) 가 주어집니다. 각 테스트 케이스의 첫 줄에는 원 배열의 길이 N (1 ≤ N ≤ 50000) 이 주어집니다. 그 다음 줄에 N 개의 정수로 A의 각 위치에 있던 값들이 움직인 칸수가 주어집니다. A 는 1부터 N 까지의 정수를 한 번씩 포함합니다.

입력의 양이 많으므로 가능한 빠른 입력 함수를 사용하는 것이 좋습니다.

출력

각 테스트 케이스마다 재구성한 A[] 를 한 줄에 출력합니다.

예제 입력

```
2
5
0 1 1 2 3
4
0 1 2 3
```

예제 출력

```
5 1 4 3 2
4 3 2 1
```

노트

2개의 댓글이 있습니다.