

Week11 – Binary Search Tree

- 주의 사항: 부정행위 금지(채점서버 외 인터넷 사용금지), STL 사용금지 (string, vector는 사용 가능)
- 표준 입출력 사용을 권장 (C는 scanf / printf, C++은 cin / cout)

문제 1

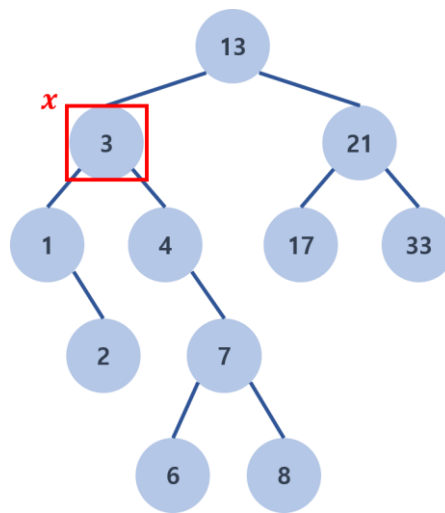


그림 1. 이진 탐색 트리

N개의 자연수로 이루어진 이진 탐색 트리를 만들어보자. 만들어진 이진 탐색 트리의 특정 노드 x 를 찾아서 그 특정 노드 x 의 왼쪽 서브트리의 최솟값과, 오른쪽 서브 트리의 최댓값을 찾아보자.

예로서, 수열 [13, 3, 21, 1, 4, 17, 33, 2, 7, 6, 8] 의 값이 차례대로 이진 탐색 트리에 Insert되면 그림 1과 같은 이진 탐색 트리가 생성된다. 이 때, 정수 3을 가진 노드를 노드 x 라고 하자. 이 때 노드 x 의 왼쪽 서브 트리의 최솟값은 1이고, 노드 x 의 오른쪽 서브 트리의 최댓값은 8이다.

입력

- 1. 첫 번째 줄에는 테스트 케이스의 수 T 가 주어진다. ($1 \leq T \leq 1,000$)
- 2. 두 번째 줄부터, 다음이 T 번 반복된다.
 - 1) 이진 탐색 트리를 구성할 자연수의 개수 P ($1 \leq P \leq 300$)가 주어진다.
 - 2) 이진 탐색 트리를 구성할 P 개의 자연수 N ($1 \leq N \leq 1,000$)이 빈칸을 사이에 두고 주어진다. (단, 값이 중복되어 나오지 않는다.)
 - 3) 이진 탐색 트리를 구성하고 있는 자연수 중 하나인 x 가 주어진다.

출력

매 테스트 케이스마다 주어진 이진 탐색 트리에서 자연수 x 를 가진 노드의 왼쪽 서브 트리의 최솟값과 오른쪽 서브 트리의 최댓값을 공백을 두고 차례대로 출력하라. 단, 왼쪽 서브 트리가 없을 경우 최솟값에 x 를 출력하고, 오른쪽 서브 트리가 없을 경우, 최댓값에 x 를 출력한다.

예제 입출력

예제 입력	예제 출력
5	3 3
7	1 7
4 2 6 1 3 5 7	1 4
3	4 8
7	2 10
4 2 6 1 3 5 7	
4	
8	
5 3 8 1 4 7 9 2	
3	
11	
13 3 21 1 4 17 33 2 7 6 8	
4	
10	
11 8 13 6 10 12 14 5 7 2	
8	