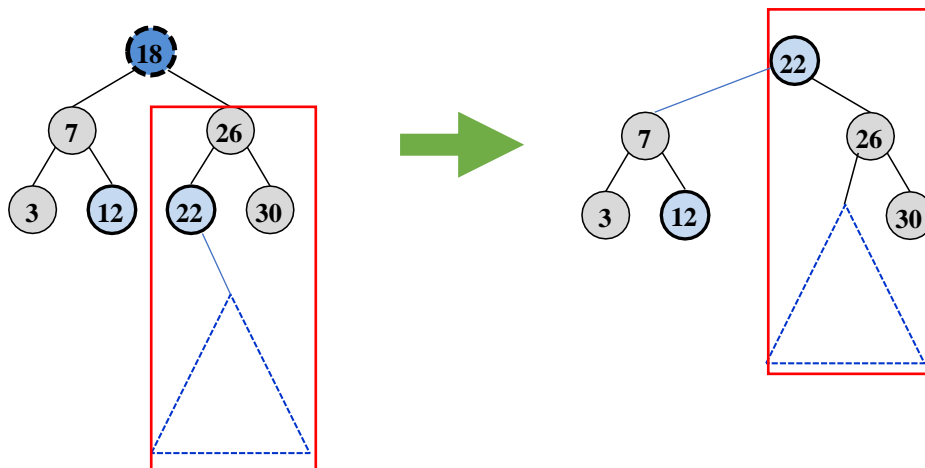


Week11 – Binary Search Tree

- 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 2

N 개의 서로 다른 자연수로 이루어진 이진 탐색 트리(binary search tree)를 만들고, 만들어진 이진 탐색 트리에서 삭제가 가능한 프로그램을 만들어보자. 아래의 그림은 자식이 둘 다 존재하는 노드 z (노드 18)를 삭제하는 하나의 예시를 보여준다. 즉, 노드 z의 오른쪽 서브 트리에서 가장 작은 노드 y (노드 22)를 찾아, z의 위치로 가져오는 과정을 보여주고 있다.



이와 같은 이진 탐색 트리에서의 삭제는 다음의 의사코드를 이용하여 구현할 수 있다.

TREE-DELETE(T, z)

```
1  if  $z.left == NIL$ 
2      TRANSPLANT( $T, z, z.right$ )
3  elseif  $z.right == NIL$ 
4      TRANSPLANT( $T, z, z.left$ )
5  else  $y = TREE-MINIMUM(z.right)$ 
6      if  $y.p \neq z$ 
7          TRANSPLANT( $T, y, y.right$ )
8           $y.right = z.right$ 
9           $y.right.p = y$ 
10     TRANSPLANT( $T, z, y$ )
11      $y.left = z.left$ 
12      $y.left.p = y$ 
```

TRANSPLANT(T, u, v)

```
1  if  $u.p == NIL$ 
2       $T.root = v$ 
3  elseif  $u == u.p.left$ 
4       $u.p.left = v$ 
5  else  $u.p.right = v$ 
6  if  $v \neq NIL$ 
7       $v.p = u.p$ 
```

입력

- 1. 첫 번째 줄에는 테스트케이스의 수 T 가 주어진다. ($1 \leq T \leq 1,000$)
(아래의 과정이 T 회 반복된다.)
- 2. 두 번째 줄에는 트리에 입력될 자연수의 수 N 이 주어진다. ($1 \leq N \leq 1,000$)
- 3. 세 번째 줄에는 이진 탐색 트리를 구성할 N 개의 서로 다른 자연수가 빈칸을 사이에 두고 주어진다.
- 4. 네 번째 줄에는 삭제할 노드의 수 M 이 주어진다. ($1 \leq M \leq N$)
- 5. 다섯 번째 줄에는 이진 탐색 트리에서 삭제할 M 개의 자연수가 빈칸을 사이에 두고 주어진다.

출력

M 개의 노드가 삭제된 후, 이진 탐색 트리를 전위 순회(pre-order traversal)하며 트리의 모든 노드를 빈칸을 사이에 두고 출력한다. 출력할 노드가 없는 경우 0을 출력한다.

예제 입출력

예제 입력	예제 출력
2 // 테스트케이스의 수 (T)	26 7 3 30
7 // 입력될 자연수의 수 (N)	92 87 38
18 7 26 3 12 22 30	
3 // 삭제할 노드의 수 입력 (M)	
18 22 12 // 삭제할 노드 입력	
6 // 입력될 자연수의 수 (N)	
92 80 24 18 38 87	
3 // 삭제할 노드의 수 입력 (M)	
80 24 18 // 삭제할 노드 입력	