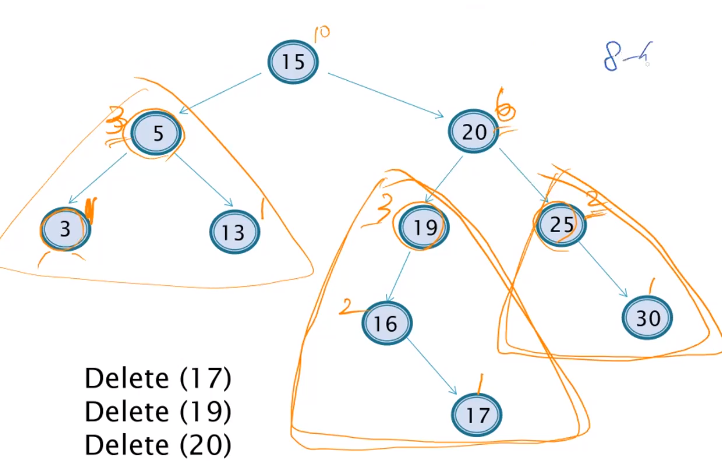
**(크기순으로 정렬했을 때) k번째인 노드는?**

**k-selection**



Size\_field를 이용: 부분트리에 속한 노드의 개수 [자식의 개수+자신의 개수(1)]

**If) 8번째 노드를 찾아라**

: 루트(15)기준으로 왼쪽 서브트리에서 3개, 자신(루트) 포함하면 4개되니까 문제를 아래와 같이 수정 가능

-> 오른쪽 서브 트리에서 4번째를 찾아라

-> 루트(20) 기준으로 왼쪽 서브트리에 3개, 20포함하면 4개 되니까 ‘20’이 답이 됨

**If) 6번째 노드를 찾아라**

: 루트(15)기준으로 왼쪽 서브트리에서 3개, 자신(루트) 포함하면 4개됨

-> 오른쪽 서브트리에서 2번째를 찾아라 (루트(15)기준)

-> 루트(20) 기준으로 왼쪽 서브트리에 3개 있으므로 거기서 조사

( 2번째를 찾아야 되는데 루트(20,4번째)와 오른쪽 서브트리(25,30)은 볼 필요가 없으니까)

( 왼쪽 size가 k이상이면 거기서 조사 )

-> 왼쪽 서브트리에서 2번째를 찾아라 (루트(20)기준)

-> 루트(19) 기준으로 왼쪽 서브트리에 2개 있으므로 거기서 조사

-> 왼쪽 서브트리에서 2번째 찾아라 (루트(19)기준)

-> 루트(16) 기준으로 왼쪽 서브트리(0)과 자신 포함하면 1이 됨 <맨 처음과 같은 로직>

( 왼쪽 size가 k미만이면 ‘왼쪽 size+자기 자신(1)’ 더하고 k에서 뺌, 뺀게 0이면 자기 자신(루트)이고 1이상이면 오른쪽 서브트리에서 뺀 것을 찾으라고 함 )

-> 오른쪽 서브트리에서 1번째 찾아라 (루트(16)기준)

-> 17이다!