

BTS

[문제] 탐색을 위한 이진트리(Binary Tree for Searching¹⁾)에 string 데이터를 추가(insert), 삭제 그리고 몇 가지 작업을 한다. 이 작업 도중에 관리하고 있는 BTS의 depth k 에 있는 node 또는 leaf node를 모두 찾아 왼쪽부터 오른쪽 순으로, 사전식 순서대로 모두 출력해서 작업을 확인한다.

명령어 형식	동작 내용	추가 설명
>> + string "+ box"	문자열 string을 현재 BTS에 들어갈 위치를 찾아 leaf 위치에 이것을 추가한다.	이미 Key값이 Tree에 있는 경우에는 변함이 없다.
>> - string "- soju"	제거할 data가 leaf에 있다면 바로 지운다. 만일 그것이 내부 노드라면 그것을 지운 다음 왼쪽 부트리의 최대값으로 대체한다. 만일 왼쪽 부트리가 없으면 오른쪽 부트리의 최소값으로 교체하고 leaf에 도달할 때까지 계속한다.	해당되는 Key가 없는 경우에는 무시한다. 따라서 BTS는 그대로 유지된다.
>> depth k	depth k 에 있는 노드를 찾아 사전식 순서로 모두 출력한다. 없을 경우에는 "NO"를 출력.	root의 depth는 1이며 depth k 노드의 자식 노드의 depth는 $k+1$.
>> leaf	BTS의 leaf를 왼쪽부터 사전식으로 모두 출력 root만 있을 경우에는 root를 출력	항상 한 개 이상 존재함.

[입출력] 입력 파일 **BTS.inp**의 첫 줄에는 명령어의 개수 N ($5 \leq N \leq 50$)이 주어진다. 그리고 이어지는 N 개의 각 줄에 한 개의 명령어가 주어진다. 여러분은 명령어 "depth k "이나 "leaf"에 대하여 해당되는 원소를 사전식 순서대로 한 줄에 모두 출력해야 한다.

[보기]

BTS.inp	BTS.out
13 // 명령어 개수	chip // leaf의 결과
+ phone	soccer // depth 3
+ banana	banana pizza //depth 2
- cola // 없으면 무시함	banana soccer // leaf
+ chip	
leaf	
+ pizza	
+ soccer	
- phone	
depth 3	
+ machine	
depth 2	
- pizza	
leaf	

[제한조건] 프로그램의 이름은 **BTS.cpp**이다. 과제 마감은 12월 10일(목요일) 23시이며 제출은 12월 2일(목요일)부터 가능하다. 제출 횟수는 최대 15번이며 허용 시간은 1초이다.

1) 1) BTS (Binary Tree for Searching), 방탄소년단이 아니라.