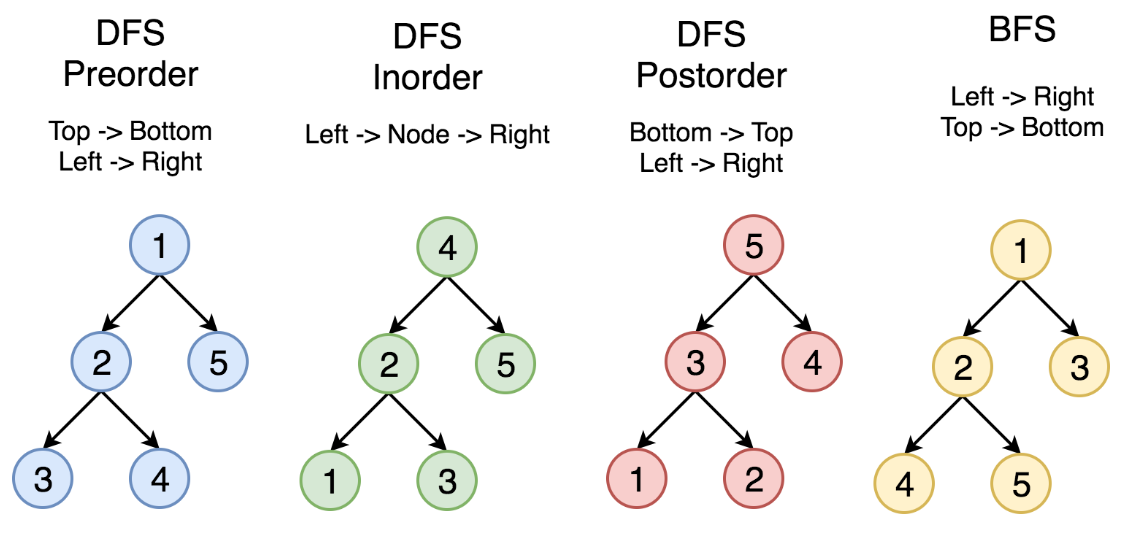
**트리**

1. **트리 순회 ( #1991 )**

노드 A와 LEFT, RIGHT 순서대로 트리가 주어집니다.

해당 트리를 전위순회, 중위순회, 후위순회 하였을 때의 결과를 출력하세요.

****

**[입력]**

**7**

**A B C**

**B D .**

**C E F**

**E . .**

**F . G**

**D . .**

**G . .**

**[출력]**

**ABDCEFG**

**DBAECFG**

**DBEGFCA**

1. **트리 부모 찾기 ( #11725 )**

노드의 개수 N이 주어집니다.

연결된 두 정점이 주어집니다.

루트가 1이라고 가정했을 때, 1을 제외한 노드들의 부모를 출력하세요.

**입력**

**7**

**1 6**

**6 3**

**3 5**

**4 1**

**2 4**

**4 7**

**출력**

**4**

**6**

**1**

**3**

**1**

**4**

**정답코드**

**#1991**

**def** preOrder(start):

**if** start != -18:

print(chr(start+64), end=*""*)

preOrder(tree[start][0])

preOrder(tree[start][1])

**def** inOrder(start):

**if** start != -18:

inOrder(tree[start][0])

print(chr(start+64), end=*""*)

inOrder(tree[start][1])

**def** postOrder(start):

**if** start != -18:

postOrder(tree[start][0])

postOrder(tree[start][1])

print(chr(start+64), end=*""*)

n = int(input())

tree = [[] **for** \_ **in** range(n+1)]

*# 아스키 코드로 넣어주기*

**for** i **in** range(n):

a, b, c = input().split()

a, b, c = ord(a)-64, ord(b)-64, ord(c)-64

tree[a].append(b)

tree[a].append(c)

*# 전위 중위 후위*

preOrder(1)

print()

inOrder(1)

print()

postOrder(1)

**#11725**

**import** **sys**

sys.setrecursionlimit(99999)

**def** dfs(node,prv):

*# parent[node] = prv*

*# child[prv].append(node)*

**for** nxt **in** relation[node]:

**if** nxt == prv:

**continue**

*# 부모 자식 관계*

*# child[node].append(nxt)*

parent[nxt] = node

*# 깊이*

*# depth[nxt] = depth[node] + 1*

*# 자식에게 이동 -- ↑*

dfs(nxt,node)

*# 부모에게 이동 -- ↓*

*# 자식들의 숫자*

*# child\_num[node] += child\_num[nxt]*

n = int(input())

relation = [[] **for** \_ **in** range(n+1)]

parent = [0 **for** \_ **in** range(n+1)]

child = [[] **for** \_ **in** range(n+1)]

depth = [0 **for** \_ **in** range(n+1)]

child\_num = [1 **for** \_ **in** range(n+1)]

*# 입력*

**for** i **in** range(n-1):

a, b = map(int,input().split())

relation[a].append(b)

relation[b].append(a)

*# 출력*

dfs(1, 0)

**for** answer **in** parent[2:]:

print(answer)