

## Problem K

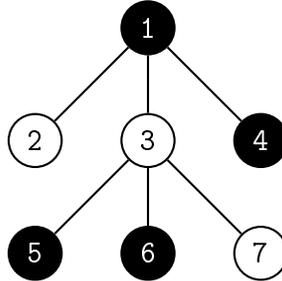
# White-Black Tree

The White-Black Tree is a game played by two players on a rooted tree of  $N$  nodes. The nodes are numbered from 1 to  $N$  and node 1 is the root. Each node in the tree has a color  $C_i$  of either 0 (representing black) or 1 (representing white) that can be changed according to the rule of the game.

Two opposing players play alternately. In their turn, the player chooses a **white** node in the tree; let the chosen node be  $x$ . First, the color of node  $x$  ( $C_x$ ) is changed from white to black. Then, in the same turn, the player is allowed to change the color of zero or more nodes that are a descendant of node  $x$  from white to black or black to white. A node  $y$  is a descendant of a node  $x$  if and only if the parent of node  $y$  is either node  $x$  or a descendant of node  $x$ . The player who cannot make any move loses and the opposing player wins the game.

Your task in this problem is to determine who will win the game assuming both players play optimally; it means that if there exists a move that guarantees their win, then they will surely play that move.

For example, consider the following game with a rooted tree of  $N = 7$  nodes and let the initial colors be  $C_{1..7} = \{0, 1, 1, 0, 0, 0, 1\}$ .



There are three white nodes (node 2, 3, and 7) in which the first player can choose for their first move. In this example, there is a strategy for the first player to win the game. One of the optimal plays for the first player is to choose node 3, change  $C_3$  into 0 (black), and then change the color of node 6 and node 7 (both are node 3's descendant), i.e.  $C_6$  becomes 1 (white) and  $C_7$  becomes 0 (black). When it's the second player's turn, there are only two white nodes to choose from (node 2 and 6) and both of them do not have any descendants. No matter what the second player chooses for their turn, the first player will simply choose the remaining one white node so that the second player will not be able to make any move. Therefore, the second player loses and the first player wins this example game.

### Input

Input begins with a line containing an integer  $N$  ( $2 \leq N \leq 100\,000$ ) representing the number of nodes in the given tree. The second line contains  $N - 1$  integers  $P_i$  ( $1 \leq P_i < i$ ) for  $i = 2..N$  representing the parent of node  $i$ . The third line contains  $N$  integers  $C_i$  ( $C_i \in \{0, 1\}$ ) representing the initial color of node  $i$ . The color black is represented by the integer 0 while the color white is represented by the integer 1.

## Output

Output a string "First" in a line if the first player will win the game assuming both players play the game optimally. Otherwise, output a string "Second" in a line.

### Sample Input #1

```
7
1 1 1 3 3 3
0 1 1 0 0 1
```

### Sample Output #1

```
First
```

*Explanation for the sample input/output #1*

This is the example from the problem description.

### Sample Input #2

```
5
1 1 2 3
0 1 1 0 0
```

### Sample Output #2

```
Second
```

*Explanation for the sample input/output #2*

The black root has two white children with the same subtree structure and colors. Whatever move the first player makes on one child, the second player simply mimics them on the other child.

### Sample Input #3

```
4
1 1 1
1 1 0 1
```

### Sample Output #3

```
First
```

*Explanation for the sample input/output #3*

The first player can make all nodes to be black in one move by choosing node 1.