Question 2
Tree Pruning

Input: from standard input Output: to standard output Source file: prune.c

We are given a rooted tree with N nodes in which each node has at most two children. Each node may be black or white. We define a "prune" as the deletion of a node and the subtree rooted at that node from the tree. Given an integer D, find the minimum number of "prunes" required to obtain a tree in which the number of white nodes minus the number of black nodes is exactly D, or determine that it is impossible to do so.

Input Specification

The first line of input will contain two integers N ($1 \leq N \leq 300$) and D ($-N \leq D \leq N$), representing the number of nodes in the tree and the value of the required difference, respectively. The next N blocks of input each contain the description of a node. The first line of each block contains three integers: the id of the node (a unique integer between 0 and $N - 1$), the colour of the node (1 for a white node, 0 for a black node) and an integer C that represents the number of children of the node. C lines follow, each one containing an integer that represents the id of one child. The root of the tree is the node with id 0.

Output Specification

On one line, output the minimum number of "prunes", as mentioned in the problem description.
If it is impossible to obtain the required difference D, output -1.

Sample Input

```
6 3
0 1 2
1
3
1 1 2
2
5
2 1 1
4
3 1 0
4 0 0
5 1 0
```

Output for Sample Input

```
1
```