

## Problem J. Colorful Tree

Your task is to maintain a colorful tree and process queries.

At the beginning, there is only one vertex numbered 1 with color  $C$  on the tree. Then there are  $q$  operations of two types coming in order:

- $0\ x\ c\ d$ : Add a new vertex indexed  $(n+1)$  with color  $c$  to the tree, where  $n$  is the current number of existing vertices. An edge connecting vertex  $x$  and  $(n+1)$  with length  $d$  will also be added to the tree.
- $1\ x\ c$ : Change the color of vertex  $x$  to  $c$ .

After each operation, you should find a pair of vertices  $u$  and  $v$  ( $1 \leq u, v \leq n$ ) with **different** colors in the current tree so that the distance between  $u$  and  $v$  is as large as possible.

The distance between two vertices  $u$  and  $v$  is the length of the shortest path from  $u$  to  $v$  on the tree.

### Input

There are multiple test cases. The first line of the input contains an integer  $T$  indicating the number of test cases. For each test case:

The first line of the input contains two integers  $q$  and  $C$  ( $1 \leq q \leq 5 \times 10^5$ ,  $1 \leq C \leq q$ ) indicating the number of operations and the initial color of vertex 1.

For the following  $q$  lines, each line describes an operation taking place in order with 3 or 4 integers.

- If the  $i$ -th line contains 4 integers  $0, x_i, c_i$  and  $d_i$  ( $1 \leq x_i \leq n$ ,  $1 \leq c_i \leq q$ ,  $1 \leq d_i \leq 10^9$ ), the  $i$ -th operation will add a new vertex  $(n+1)$  with color  $c_i$  to the tree and connect it to vertex  $x_i$  with an edge of length  $d_i$ .
- If the  $i$ -th line contains 3 integers  $1, x_i$  and  $c_i$  ( $1 \leq x_i \leq n$ ,  $1 \leq c_i \leq q$ ), the  $i$ -th operation will change the color of vertex  $x_i$  to  $c_i$ .

It's guaranteed that the sum of  $q$  of all test cases will not exceed  $5 \times 10^5$ .

### Output

For each operation output the maximum distance between two vertices with different colors. If no valid pair exists output 0 instead.

### Example

standard input	standard output
2	0
1 1	0
0 1 1 1	2
5 1	3
0 1 1 1	2
0 1 2 1	0
0 3 3 1	
1 4 1	
1 3 1	