E. New Year Tree

time limit per test: 3 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

The New Year holidays are over, but Resha doesn't want to throw away the New Year tree. He invited his best friends Kerim and Gural to help him to redecorate the New Year tree.

The New Year tree is an undirected tree with n vertices and root in the vertex 1.

You should process the queries of the two types:

1. Change the colours of all vertices in the subtree of the vertex v to the colour c.
2. Find the number of different colours in the subtree of the vertex v.

Input

The first line contains two integers n, m ($1 \le n, m \le 4 \cdot 10^5$) — the number of vertices in the tree and the number of the queries.

The second line contains n integers c_i ($1 \le c_i \le 60$) — the colour of the i-th vertex.

Each of the next $n - 1$ lines contains two integers x_j, y_j ($1 \le x_j, y_j \le n$) — the vertices of the j-th edge. It is guaranteed that you are given correct undirected tree.

The last m lines contains the description of the queries. Each description starts with the integer t_k ($1 \le t_k \le 2$) — the type of the k-th query. For the queries of the first type then follows two integers v_k, c_k ($1 \le v_k \le n, 1 \le c_k \le 60$) — the number of the vertex whose subtree will be recoloured with the colour c_k. For the queries of the second type then follows integer v_k ($1 \le v_k \le n$) — the number of the vertex for which subtree you should find the number of different colours.

Output

For each query of the second type print the integer a — the number of different colours in the subtree of the vertex given in the query.

Each of the numbers should be printed on a separate line in order of query appearing in the input.

Examples

input

```
7 10
1 1 1 1 1 1 1
1 2
1 3
1 4
3 5
3 6
3 7
1 3 2
2 1
1 4 3
2 1
1 2 5
2 1
1 6 4
2 1
```

```
2 2
2 3
```

output

```
2
3
4
5
1
2
```

input

```
23 30
1 2 2 6 5 3 2 1 1 1 2 4 5 3 4 4 3 3 3 3 3 4 6
1 2
1 3
1 4
2 5
2 6
3 7
3 8
4 9
4 10
4 11
6 12
6 13
7 14
7 15
7 16
8 17
8 18
10 19
10 20
10 21
11 22
11 23
2 1
2 5
2 6
2 7
2 8
2 9
2 10
2 11
2 4
1 12 1
1 13 1
1 14 1
1 15 1
1 16 1
1 17 1
1 18 1
1 19 1
1 20 1
1 21 1
1 22 1
1 23 1
2 1
2 5
2 6
2 7
2 8
2 9
2 10
2 11
2 4
```

output

```
6
1
3
3
2
1
2
3
5
5
1
```

→ Contest materials

- Announcement
- Tutorial

```
2
2
1
1
1
2
3
```