IOITC 2021

Edge subsets

You are given a rooted tree with n vertices. It is rooted at node 1. Additionally there are m tuples of the form (u_i, v_i, l_i, r_i). u_i, v_i are vertices in the tree such that u_i is an ancestor of v_i, and l_i, r_i are integers.

Choose a subset of the edges such that for every such tuple, the number of edges chosen on the path from u_i to v_i lies in the range $[l_i, r_i]$, or state that no such subset exists.

Input

- The first line contains an integer T, the number of test cases. T test cases follow.
- The first line of each testcase contains two space separated integers n and m, the number of vertices and number of tuples respectively.
- The second line contains $n-1$ space separated integers $p_2, p_3, \ldots, p_n$, where p_i is parent of i-th node.
- Next m lines each containing four space separated integers u_i, v_i, l_i, r_i $(1 \le u_i, v_i \le n, 0 \le l_i \le r_i \le n-1)$ — representing m tuples.

Output

For each test case,

- If there is no such subset then print -1 in a single line.
- Otherwise if there exists one such subset then,
- In the first line, print an integer k — the size of chosen edge subset.
- In second line print k space separated integers $v_1, v_2, v_3, \ldots, v_k$ — describing chosen edge subset. Here v_i represents an edge from its parent node to itself (p_{v_i}, v_i).

Test Data

In all inputs,

- $1 \le n \le 1000$.
- $0 \le m \le 1000$.
- $0 \le l_i \le r_i \le n-1$ for each valid i.
- For each valid i, u_i is an ancestor of v_i.
- The sum of n over all testcases doesn't exceed 10^5.
- The sum of m over all testcases doesn't exceed 10^5.

Subtask 1 (10 Points):

- $1 \le n \le 17$.
- $0 \le m \le 50$.
- $0 \le T \le 50$.

Subtask 2 (20 Points):

- $1 \leq n \leq 100$.
- $0 \leq m \leq 100$.
- The sum of n over all testcases doesn't exceed 10^4.
- The sum of m over all testcases doesn't exceed 10^4.

Subtask 3 (20 Points): $u_i = 1$

Subtask 4 (50 Points): No additional constraints

Sample Input

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

Sample Output

```
-1
0

2
2 3
```

Explanation

In test case 1, there are two contradicting conditions between nodes 1 and 3, First one says there should be only one edge in the path from 1 to 3 in chosen subset while second one says, there should be two edges hence there is no such subset. Also you should not print a second line as shown in sample in case of no subset.

In test case 2, there is only one condition that, in chosen subset, there should be 0 edges in the path from 1 to 2. Hence an empty subset works. Also make sure to print an empty line as shown in sample for empty subset.

In test case 3, edge subset 3-2 and 1-3 works because number of edges in path from 1 to 3 is 1 so given condition satisfies

Limits

Time: 2 second

Memory: 256 MB