Problem D. Segments

Input file:	`standard input`
Output file:	`standard output`
Time limit:	3 seconds
Memory limit:	40 megabytes

There is a multiset of segments S. Difference between multiset and set is that multiset allows multiple instances of one segment, unlike a set.

Given two integer numbers n and t. You have n operations of following types that are made over the multiset:

1. Insert segment $[l, r]$ into the multiset S. The segment is assigned with id — minimum positive integer number that was not assigned to any other segment before.
2. Erase the segment with assigned number id from the multiset S. It is guaranteed that at the moment of erasing there is a segment in the multiset S with assigned number id.
3. Count the number of segments from the multiset S that has at least k *integer points* in common with given segment $[l, r]$.

Integer point x is common for both segments $[l_i, r_i]$ and $[l_j, r_j]$, if $l_i \le x \le r_i$ and $l_j \le x \le r_j$.

Input

The first line of input contains two integer numbers n and t ($1 \le n \le 2 \cdot 10^5, 0 \le t \le 1$) — number of operations and constant number. Each of next n lines describes one query.

1. Queries of first type are given in following format: 1 a_i b_i ($0 \le a_i, b_i \le 2 \cdot 10^9$).
2. Queries of second type are given in following format: 2 id_i ($1 \le id_i \le n$).
3. Queries of third type are given in following format: 3 a_i b_i k_i ($0 \le a_i, b_i, k_i \le 2 \cdot 10^9$).

Please note that end points of segments $[l_i, r_i]$ for queries of type 1 and 3 are **encoded**, in order to decode them you need to perform the following transformations:

$$l_i = (a_i \oplus (t * lastans)) \quad r_i = (b_i \oplus (t * lastans))$$

where $lastans$ — last answer to the query of type 3 (initially $lastans$ equals to 0). If it turned out that l_i is greater than r_i, you should swap the values of l_i and r_i.

It is guaranteed that there will be at least one query of type 3 in input.

Here $\oplus$ denotes the bitwise XOR operation.

Consider that problem has **unusual memory limit**.

Output

For each query of type 3 print answer in separate line.

Scoring

This task contains six subtasks:

1. $n \le 5 \cdot 10^3$. Scored 7 points.
2. $n \le 10^5$. First comes queries of type 1, then of type 3 and there is no query of type 2. Scored 15 points.
3. $n \le 2 \cdot 10^5$, $k_i = 1$ for all third type queries. Scored 16 points.
4. $n \le 10^5$, $t = 0$. Scored 17 points.
5. $n \le 10^5$. Scored 20 points.

6. $n \leq 2 \cdot 10^5$. Scored 25 points.

Examples

standard input	standard output
6 1 1 1 2 3 2 4 2 1 3 5 3 2 3 1 2 1 3 0 3 1	0 2 0
6 0 1 3 10 1 3 5 3 6 10 6 2 1 1 3 10 3 6 4 2	0 2

Problem E. Nice sequence

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	256 megabytes

At their leisure time Tima and Kanat play with sequences of integers. Tima considers a sequence *nice* if the sum of any N consecutive numbers of the sequence is negative and Kanat considers a sequence *nice* if the sum of any M consecutive numbers of the sequence is positive. If the sequence does not have N and/or M consecutive numbers, it is considered to be *nice* for Tima and/or Kanat respectively.

Find the sequence of **maximum** possible length that will be *nice* for both of them.

Input

The first line contains one integer $T(1 \le T \le 10)$ — the number of tests.

In the next T lines there are two integers N and M, separated by space.

Output

For each test output 2 lines: in the first line output one integer K — maximum length of the sequence, which is *nice* for both Tima and Kanat. In the second line output K numbers separated by space — the sequence itself. The numbers should not exceed 10^9 by absolute value and should be non-zero. It is guaranteed that it is possible to find a sequence of maximum length that satisfies above condition. When $K = 0$ second line should be empty.

Scoring

This task includes seven subtasks:

1. $1 \le N, M \le 100$, and $max(N, M)$ is divisible by $min(N, M)$. Score 6 points.
2. $1 \le N, M \le 10^4, min(N, M) = 2$. Score 9 points.
3. $1 \le N, M \le 10$. Score 14 points.
4. $1 \le N, M \le 2 \cdot 10^5, |N - M| \le 2$. Score 15 points.
5. $1 \le N, M \le 2000$. Score 14 points.
6. $1 \le N, M \le 5 \cdot 10^4$. Score 18 points.
7. $1 \le N, M \le 2 \cdot 10^5$. Score 24 points.

Example

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

Problem F. Birthday gift

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	256 megabytes

Askhat received from NurlashKO *rooted tree* on his birthday as a gift with n vertexes, numbered from 1 to n. *Tree* — connected unoriented graph without any cycles. The tree root is a vertex with number 1. Vertex v is an ancestor of vertex u if v lies on the minimal path from u to the root. Lowest common ancestor of sequence of vertexes $(x_1, x_2, ..., x_k)$ — farthest vertex from root, which is an ancestor of x_i for all $1 \le i \le k$ ($lca(x_1, x_2, ..., x_k)$).

In addition to the gift, NurlashKO prepared a task for Askhat. At first, he reported a sequence with length m — $(a_1, a_2, ..., a_m)$, each number in the sequence is a vertex from the tree. There may be duplicates of vertexes in the sequence. Then he started asking q queries, each query is one of the two types:

- 1 *pos* v — NurlashKO asks Askhat to change the value at position *pos* to the value v, i.e. $a_{pos} = v$.
- 2 l r v — NurlashKO asks Askhat to find a pair (x, y), such that $l \le x \le y \le r$ and $lca(a_x, a_{x+1}, ..., a_y) = v$. Or say that there is no such pair.

Askhat has spent a lot of time on researching the gift and now he wants your help.

Input

First line of input contains three positive integer numbers n, m and q — size of the tree, length of the sequence and number of queries. Next $n - 1$ lines contain edges of the tree (u_i, v_i) $(u_i \ne v_i)$. Next line contains m integer numbers, $a_1, a_2, ..., a_m$. $(1 \le a_i \le n)$ — sequence, which was gifted to Askhat by NurlashKO. Each of the next q lines describes a query. If first number of query equals to 1, then it is followed by two numbers *pos* and v $(1 \le pos \le m, 1 \le v \le n)$ — query of first type. If first number of query equals to 2, then it is followed by three numbers l, r and v $(1 \le l \le r \le m, 1 \le v \le n)$ — query of second type. It is guaranteed that among q queries at least one is of second type.

Output

Print two numbers x and y — answer to each query of second type, if there is no solution print out "-1 -1" (without quotes). If there are multiple solutions, output any of them.

Scoring

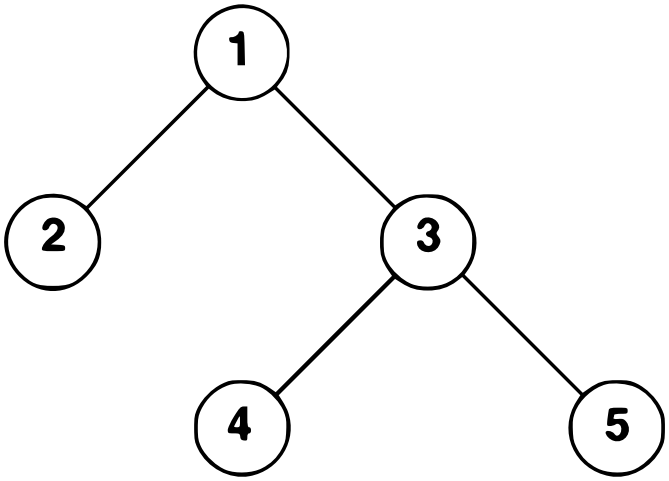
This problem consists of four subtasks, in each subtask tests satisfy constraints in statement:

1. $1 \le n, m, q \le 100$. Score 12 points.
2. $1 \le n, m, q \le 500$. Score 18 points.
3. $1 \le n, m, q \le 2000$. Score 26 points.
4. $1 \le n, m, q \le 2 \cdot 10^5$. Score 44 points.

Example

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

Note



- Sequence: $[4, 5, 2, 3]$
- Subsegment $= [4, 5, 2]$, $v = 1$. $lca(4, 5, 2) = 1$, answer: $(1, 3)$.
- Query on changing, new sequence: $[4, 5, 5, 3]$
- Subsegment $= [5, 3]$, $v = 5$. $lca(5) = 5$, answer: $(3, 3)$.
- Subsegment $= [4, 5, 5]$, $v = 1$. $lca(4) = 4$, $lca(5) = 5$, $lca(4, 5) = 3$, $lca(5, 5) = 5$, $lca(4, 5, 5) = 3$. There is no solution.