Problem I. Set Operations

Input file:	`set-operations.in`
Output file:	`set-operations.out`
Time limit:	2 seconds
Memory limit:	256 mebibytes

In this problem, you have to perform operations with sets according to the given expression.

Consider an expression involving three sets A, B and C and three set operations: union, intersection and complement.

For the purposes of this problem, sets can only consist of numbers $\{1, 2, \ldots, n\}$ for some fixed integer n. A *union* of sets X and Y is a set which contains all numbers present in at least one of the sets X and Y. An *intersection* of sets X and Y is a set which contains all numbers present in both of the sets X and Y. A *complement* of set X is a set which contains all numbers from the range $\{1, 2, \ldots, n\}$ which are not in X. For example, if $n = 5$, $X = \{3, 4\}$ and $Y = \{1, 3\}$, the union of X and Y is $\{1, 3, 4\}$, their intersection is $\{3\}$, and complement of X is $\{1, 2, 5\}$.

An *expression* is recursively defined as follows:

- `A`, `B` and `C` are expressions denoting the three given sets A, B and C respectively.
- If E is an expression, `~`E and `(`E`)` are also expressions.
- If E and F are expressions, E`|`F and E`&`F are also expressions.

Here, `~`X is the complement of set X, X`|`Y is the union of sets X and Y, and X`&`Y is the intersection of sets X and Y. Complement is evaluated before intersection, which in turn is evaluated before union. Parentheses play the usual role of prioritizing operations. So, for example, an expression `A|~B&C` is evaluated in the same way as `A|((~B)&C)`.

You are given one expression and a number of triples of sets A, B and C. Find and output the value of the expression for each of the given triples.

Input

The first line of input contains the expression. Its length is from 1 to 300 000 characters. It is guaranteed that the expression conforms with the recursive definition above. There are no spaces on the first line.

The second line contains two integers n and t separated by a space: the number of elements and the number of triples of sets ($1 \le n \le 20$, $1 \le t \le 10\,000$).

Each of the next t lines describes a triple of sets A, B and C in that order. A set is denoted by a list of integers contained in that set in strictly ascending order followed by a zero. Numbers are separated by one or more spaces.

Output

For each of the t given triples of sets, output a single line containing a single set: the result of evaluating the expression for the given A, B and C. A set must be written as a list of integers contained in that set in strictly ascending order followed by a zero. Separate numbers by one or more spaces.

Example

set-operations.in	set-operations.out
A|~B&C 5 2 1 3 0 3 4 0 2 3 5 0 0 1 3 5 0 0	1 2 3 5 0 0

Explanation

In this example, the expression is evaluated as `A|((~B)&C)`. For the first triple, `~B` $= \{1, 2, 5\}$, `~B&C` $= \{2, 5\}$ and the whole expression evaluates to the set $\{1, 2, 3, 5\}$. For the second triple, `~B` $= \{2, 4\}$, `~B&C` is empty and the result is also an empty set.