A. 24 Game

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Little X used to play a card game called "24 Game", but recently he has found it too easy. So he invented a new game.

Initially you have a sequence of n integers: $1, 2, ..., n$. In a single step, you can pick two of them, let's denote them a and b, erase them from the sequence, and append to the sequence either $a + b$, or $a - b$, or $a \times b$.

After $n - 1$ steps there is only one number left. Can you make this number equal to 24?

Input

The first line contains a single integer n $(1 \le n \le 10^5)$.

Output

If it's possible, print "`YES`" in the first line. Otherwise, print "`NO`" (without the quotes).

If there is a way to obtain 24 as the result number, in the following $n - 1$ lines print the required operations an operation per line. Each operation should be in form: "a op b = c". Where a and b are the numbers you've picked at this operation; op is either "`+`", or "`-`", or "`*`"; c is the result of corresponding operation. Note, that the absolute value of c mustn't be greater than 10^{18}. The result of the last operation must be equal to 24. Separate operator sign and equality sign from numbers with spaces.

If there are multiple valid answers, you may print any of them.

Sample test(s)

input

```
1
```

output

```
NO
```

input

```
8
```

output

```
YES
8 * 7 = 56
6 * 5 = 30
3 - 4 = -1
1 - 2 = -1
30 - -1 = 31
56 - 31 = 25
25 + -1 = 24
```

B. Two Sets

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Little X has n distinct integers: $p_1, p_2, ..., p_n$. He wants to divide all of them into two sets A and B. The following two conditions must be satisfied:

- If number x belongs to set A, then number a - x must also belong to set A.
- If number x belongs to set B, then number b - x must also belong to set B.

Help Little X divide the numbers into two sets or determine that it's impossible.

Input

The first line contains three space-separated integers n, a, b $(1 \le n \le 10^5; 1 \le a, b \le 10^9)$. The next line contains n space-separated distinct integers $p_1, p_2, ..., p_n$ $(1 \le p_i \le 10^9)$.

Output

If there is a way to divide the numbers into two sets, then print "YES" in the first line. Then print n integers: $b_1, b_2, ..., b_n$ (b_i equals either 0, or 1), describing the division. If b_i equals to 0, then p_i belongs to set A, otherwise it belongs to set B.

If it's impossible, print "NO" (without the quotes).

Sample test(s)

input

```
4 5 9
2 3 4 5
```

output

```
YES
0 0 1 1
```

input

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

output

```
NO
```

Note

It's OK if all the numbers are in the same set, and the other one is empty.

C. Hack it!

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Little X has met the following problem recently.

Let's define $f(x)$ as the sum of digits in decimal representation of number x (for example, $f(1234) = 1 + 2 + 3 + 4$). You are to calculate $\sum_{i=l}^{r} f(i) \bmod a$.

Of course Little X has solved this problem quickly, has locked it, and then has tried to hack others. He has seen the following C++ code:

```
ans = solve(l, r) % a;
if (ans <= 0)
  ans += a;
```

This code will fail only on the test with $\sum_{i=l}^{r} f(i) \equiv 0 \pmod{a}$. You are given number a, help Little X to find a proper test for hack.

Input

The first line contains a single integer a $(1 \le a \le 10^{18})$.

Output

Print two integers: l, r $(1 \le l \le r < 10^{200})$ — the required test data. Leading zeros aren't allowed. It's guaranteed that the solution exists.

Sample test(s)

input

```
46
```

output

```
1 10
```

input

```
126444381000032
```

output

```
2333333 2333333333333
```

D. Tree

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Little X has a tree consisting of n nodes (they are numbered from 1 to n). Each edge of the tree has a positive length. Let's define the distance between two nodes v and u (we'll denote it $d(v, u)$) as the sum of the lengths of edges in the shortest path between v and u.

A permutation p is a sequence of n distinct integers $p_1, p_2, ..., p_n$ $(1 \le p_i \le n)$. Little X wants to find a permutation p such that sum $\sum_{i=1}^{n} d(i, p_i)$ is maximal possible. If there are multiple optimal permutations, he wants to find the lexicographically smallest one. Help him with the task!

Input

The first line contains an integer n $(1 \le n \le 10^5)$.

Each of the next n - 1 lines contains three space separated integers u_i, v_i, w_i $(1 \le u_i, v_i \le n; 1 \le w_i \le 10^5)$, denoting an edge between nodes u_i and v_i with length equal to w_i.

It is guaranteed that these edges form a tree.

Output

In the first line print the maximum possible value of the described sum. In the second line print n integers, representing the lexicographically smallest permutation.

Sample test(s)

input

```
2
1 2 3
```

output

```
6
2 1
```

input

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

output

```
32
2 1 4 5 3
```

E. Permanent

time limit per test: 2 seconds
memory limit per test: 512 megabytes
input: standard input
output: standard output

Little X has solved the #P-complete problem in polynomial time recently. So he gives this task to you.

There is a special $n \times n$ matrix A, you should calculate its permanent modulo 1000000007 $(10^9 + 7)$. The special property of matrix A is almost all its elements equal to 1. Only k elements have specified value.

You can find the definition of permanent at the link: `https://en.wikipedia.org/wiki/Permanent`

Input

The first line contains two space-separated integers n, k $(1 \le n \le 10^5; 1 \le k \le 50)$.

The next k lines contain the description of the matrix. The i-th line contains three space-separated integers x_i, y_i, w_i $(1 \le x_i, \ y_i \le \ n; 0 \ \le \ w_i \ \le 10^9)$. These numbers denote that $A_{x_i, y_i} = w_i$. All the elements of the matrix except of the given elements are equal to 1.

It's guaranteed that all the positions (x_i, y_i) are distinct.

Output

Print the permanent of the matrix modulo 1000000007 $(10^9 \ + \ 7)$.

Sample test(s)

input

```
3 1
1 1 2
```

output

```
8
```

input

```
10 10
3 3 367056794
6 2 124561273
1 3 46718146
6 9 415916869
10 5 985968336
3 1 526792265
1 4 386357058
10 4 349304187
2 7 102032499
3 6 502679075
```

output

```
233333333
```