Codeforces Round #279 (Div. 2)

A. Team Olympiad

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

The School №0 of the capital of Berland has n children studying in it. All the children in this school are gifted: some of them are good at programming, some are good at maths, others are good at PE (Physical Education). Hence, for each child we know value t_i:

- $t_i = 1$, if the i-th child is good at programming,
- $t_i = 2$, if the i-th child is good at maths,
- $t_i = 3$, if the i-th child is good at PE

Each child happens to be good at exactly one of these three subjects.

The Team Scientific Decathlon Olympias requires teams of three students. The school teachers decided that the teams will be composed of three children that are good at different subjects. That is, each team must have one mathematician, one programmer and one sportsman. Of course, each child can be a member of no more than one team.

What is the maximum number of teams that the school will be able to present at the Olympiad? How should the teams be formed for that?

Input

The first line contains integer n ($1 \le n \le 5000$) — the number of children in the school. The second line contains n integers $t_1, t_2, \ldots, t_n$ ($1 \le t_i \le 3$), where t_i describes the skill of the i-th child.

Output

In the first line output integer w — the largest possible number of teams.

Then print w lines, containing three numbers in each line. Each triple represents the indexes of the children forming the team. You can print both the teams, and the numbers in the triplets in any order. The children are numbered from 1 to n in the order of their appearance in the input. Each child must participate in no more than one team. If there are several solutions, print any of them.

If no teams can be compiled, print the only line with value w equal to 0.

Examples

input
7 1 3 1 3 2 1 2
output
2 3 5 2 6 7 4

input
4 2 1 1 2
output
0

B. Queue

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

During the lunch break all n Berland State University students lined up in the food court. However, it turned out that the food court, too, has a lunch break and it temporarily stopped working.

Standing in a queue that isn't being served is so boring! So, each of the students wrote down the number of the student ID of the student that stands in line directly in front of him, and the student that stands in line directly behind him. If no one stands before or after a student (that is, he is the first one or the last one), then he writes down number `0` instead (in Berland State University student IDs are numerated from 1).

After that, all the students went about their business. When they returned, they found out that restoring the queue is not such an easy task.

Help the students to restore the state of the queue by the numbers of the student ID's of their neighbors in the queue.

Input

The first line contains integer n ($2 \le n \le 2 \cdot 10^5$) — the number of students in the queue.

Then n lines follow, i-th line contains the pair of integers a_i, b_i ($0 \le a_i, b_i \le 10^6$), where a_i is the ID number of a person in front of a student and b_i is the ID number of a person behind a student. The lines are given in the arbitrary order. Value `0` is given instead of a neighbor's ID number if the neighbor doesn't exist.

The ID numbers of all students are distinct. It is guaranteed that the records correspond too the queue where all the students stand in some order.

Output

Print a sequence of n integers $x_1, x_2, \ldots, x_n$ — the sequence of ID numbers of all the students in the order they go in the queue from the first student to the last one.

Examples

input
4 92 31 0 7 31 0 7 141
output
92 7 31 141

Note

The picture illustrates the queue for the first sample.

C. Hacking Cypher

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

Polycarpus participates in a competition for hacking into a new secure messenger. He's almost won.

Having carefully studied the interaction protocol, Polycarpus came to the conclusion that the secret key can be obtained if he properly cuts the public key of the application into two parts. The public key is a long integer which may consist of even a million digits!

Polycarpus needs to find such a way to cut the public key into two nonempty parts, that the first (left) part is divisible by a as a separate number, and the second (right) part is divisible by b as a separate number. Both parts should be *positive* integers that have no leading zeros. Polycarpus knows values a and b.

Help Polycarpus and find any suitable method to cut the public key.

Input

The first line of the input contains the public key of the messenger — an integer without leading zeroes, its length is in range from 1 to 10^6 digits. The second line contains a pair of space-separated positive integers a, b ($1 \le a, b \le 10^8$).

Output

In the first line print "YES" (without the quotes), if the method satisfying conditions above exists. In this case, next print two lines — the left and right parts after the cut. These two parts, being concatenated, must be exactly identical to the public key. The left part must be divisible by a, and the right part must be divisible by b. The two parts must be positive integers having no leading zeros. If there are several answers, print any of them.

If there is no answer, print in a single line "NO" (without the quotes).

Examples

input
116401024 97 1024
output
YES 11640 1024

input
284254589153928171911281811000 1009 1000
output
YES 2842545891539 28171911281811000

input
120 12 1
output
NO

D. Chocolate

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

Polycarpus likes giving presents to Paraskevi. He has bought two chocolate bars, each of them has the shape of a segmented rectangle. The first bar is $a_1 \times b_1$ segments large and the second one is $a_2 \times b_2$ segments large.

Polycarpus wants to give Paraskevi one of the bars at the lunch break and eat the other one himself. Besides, he wants to show that Polycarpus's mind and Paraskevi's beauty are equally matched, so the two bars must have the same number of squares.

To make the bars have the same number of squares, Polycarpus eats a little piece of chocolate each minute. Each minute he does the following:

- he either breaks one bar exactly in half (vertically **or** horizontally) and *eats exactly a half* of the bar,
- or he chips of exactly one third of a bar (vertically or horizontally) and *eats exactly a third* of the bar.

In the first case he is left with a *half*, of the bar and in the second case he is left with *two thirds* of the bar.

Both variants aren't always possible, and sometimes Polycarpus cannot chip off a half nor a third. For example, if the bar is 16×23, then Polycarpus can chip off a half, but not a third. If the bar is 20×18, then Polycarpus can chip off both a half and a third. If the bar is 5×7, then Polycarpus cannot chip off a half nor a third.

What is the minimum number of minutes Polycarpus needs to make two bars consist of the same number of squares? Find not only the required minimum number of minutes, but also the possible sizes of the bars after the process.

Input

The first line of the input contains integers a_1, b_1 ($1 \le a_1, b_1 \le 10^9$) — the initial sizes of the first chocolate bar. The second line of the input contains integers a_2, b_2 ($1 \le a_2, b_2 \le 10^9$) — the initial sizes of the second bar.

You can use the data of type `int64` (in Pascal), `long long` (in C++), `long` (in Java) to process large integers (exceeding $2^{31} - 1$).

Output

In the first line print m — the sought minimum number of minutes. In the second and third line print the possible sizes of the bars after they are leveled in m minutes. Print the sizes using the format identical to the input format. Print the sizes (the numbers in the printed pairs) in any order. The second line must correspond to the first bar and the third line must correspond to the second bar. If there are multiple solutions, print any of them.

If there is no solution, print a single line with integer `-1`.

Examples

input
2 6 2 3
output
1 1 6 2 3

input
36 5 10 16
output
3 16 5 5 16

input
3 5 2 1
output
-1

E. Restoring Increasing Sequence

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

Peter wrote on the board a strictly increasing sequence of positive integers $a_1, a_2, ..., a_n$. Then Vasil replaced some digits in the numbers of this sequence by question marks. Thus, each question mark corresponds to exactly one lost digit.

Restore the the original sequence knowing digits remaining on the board.

Input

The first line of the input contains integer n ($1 \le n \le 10^5$) — the length of the sequence. Next n lines contain one element of the sequence each. Each element consists only of digits and question marks. No element starts from digit `0`. Each element has length from 1 to 8 characters, inclusive.

Output

If the answer exists, print in the first line "YES" (without the quotes). Next n lines must contain the sequence of positive integers — a possible variant of Peter's sequence. The found sequence must be strictly increasing, it must be transformed from the given one by replacing each question mark by a single digit. All numbers on the resulting sequence must be written without leading zeroes. If there are multiple solutions, print any of them.

If there is no answer, print a single line "NO" (without the quotes).

Examples

input
3 ? 18 1?
output
YES 1 18 19

input
2 ?? ?
output
NO

input
5 12224 12??5 12226 ?0000 ?00000
output
YES 12224 12225 12226 20000 100000

F. Treeland Tour

time limit per test: 5 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

The "Road Accident" band is planning an unprecedented tour around Treeland. The RA fans are looking forward to the event and making bets on how many concerts their favorite group will have.

Treeland consists of n cities, some pairs of cities are connected by bidirectional roads. Overall the country has $n - 1$ roads. We know that it is possible to get to any city from any other one. The cities are numbered by integers from 1 to n. For every city we know its value r_i — the number of people in it.

We know that the band will travel along some path, having concerts in **some** cities along the path. The band's path will not pass one city twice, each time they move to the city that hasn't been previously visited. Thus, the musicians will travel along some path (without visiting any city twice) and in some (not necessarily all) cities along the way they will have concerts.

The band plans to gather all the big stadiums and concert halls during the tour, so every time they will perform in a city which population is *larger* than the population of the previously visited **with concert** city. In other words, the sequence of population in the cities where the concerts will be held is *strictly increasing*.

In a recent interview with the leader of the "road accident" band promised to the fans that the band will **give concert** in the largest possible number of cities! Thus the band will travel along some chain of cities of Treeland and have concerts in some of these cities, so that the population number will increase, and the number of concerts will be the largest possible.

The fans of Treeland are frantically trying to figure out how many concerts the group will have in Treeland. Looks like they can't manage without some help from a real programmer! Help the fans find the sought number of concerts.

Input

The first line of the input contains integer n ($2 \le n \le 6000$) — the number of cities in Treeland. The next line contains n integers $r_1, r_2, \ldots, r_n$ ($1 \le r_i \le 10^6$), where r_i is the population of the i-th city. The next $n - 1$ lines contain the descriptions of the roads, one road per line. Each road is defined by a pair of integers a_j, b_j ($1 \le a_j, b_j \le n$) — the pair of the numbers of the cities that are connected by the j-th road. All numbers in the lines are separated by spaces.

Output

Print the number of cities where the "Road Accident" band will have concerts.

Examples

input
6 1 2 3 4 5 1 1 2 2 3 3 4 3 5 3 6
output
4

input
5 1 2 3 4 5 1 2 1 3 2 4 3 5
output
3