Codeforces Beta Round #97 (Div. 2)

A. Presents

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

Little Petya very much likes gifts. Recently he has received a new laptop as a New Year gift from his mother. He immediately decided to give it to somebody else as what can be more pleasant than giving somebody gifts. And on this occasion he organized a New Year party at his place and invited n his friends there.

If there's one thing Petya likes more that receiving gifts, that's watching others giving gifts to somebody else. Thus, he safely hid the laptop until the next New Year and made up his mind to watch his friends exchanging gifts while he does not participate in the process. He numbered all his friends with integers from 1 to n. Petya remembered that a friend number i gave a gift to a friend number p_i. He also remembered that each of his friends received exactly one gift.

Now Petya wants to know for each friend i the number of a friend who has given him a gift.

Input

The first line contains one integer n ($1 \le n \le 100$) — the quantity of friends Petya invited to the party. The second line contains n space-separated integers: the i-th number is p_i — the number of a friend who gave a gift to friend number i. It is guaranteed that each friend received exactly one gift. It is possible that some friends do not share Petya's ideas of giving gifts to somebody else. Those friends gave the gifts to themselves.

Output

Print n space-separated integers: the i-th number should equal the number of the friend who gave a gift to friend number i.

Examples

input
4 2 3 4 1
output
4 1 2 3

input
3 1 3 2
output
1 3 2

input
2 1 2
output
1 2

B. Ternary Logic

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

Little Petya very much likes computers. Recently he has received a new "Ternatron IV" as a gift from his mother. Unlike other modern computers, "Ternatron IV" operates with ternary and not binary logic. Petya immediately wondered how the *xor* operation is performed on this computer (and whether there is anything like it).

It turned out that the operation does exist (however, it is called *tor*) and it works like this. Suppose that we need to calculate the value of the expression $a\ tor\ b$. Both numbers a and b are written in the ternary notation one under the other one (b under a). If they have a different number of digits, then leading zeroes are added to the shorter number until the lengths are the same. Then the numbers are summed together digit by digit. The result of summing each two digits is calculated modulo 3. Note that there is no carry between digits (i. e. during this operation the digits aren't transferred). For example:
$14_{10}\ tor\ 50_{10} = 0112_3\ tor\ 1212_3 = 1021_3 = 34_{10}$.

Petya wrote numbers a and c on a piece of paper. Help him find such number b, that $a\ tor\ b = c$. If there are several such numbers, print the smallest one.

Input

The first line contains two integers a and c ($0 \le a, c \le 10^9$). Both numbers are written in decimal notation.

Output

Print the single integer b, such that $a\ tor\ b = c$. If there are several possible numbers b, print the smallest one. You should print the number in decimal notation.

Examples

input
14 34
output
50

input
50 34
output
14

input
387420489 225159023
output
1000000001

input
5 5
output
0

C. Replacement

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

Little Petya very much likes arrays consisting of n integers, where each of them is in the range from 1 to 10^9, inclusive. Recently he has received one such array as a gift from his mother. Petya didn't like it at once. He decided to choose exactly one element from the array and replace it with another integer that also lies in the range from 1 to 10^9, inclusive. It is **not allowed** to replace a number with itself or to change no number at all.

After the replacement Petya sorted the array by the numbers' non-decreasing. Now he wants to know for each position: what minimum number could occupy it after the replacement and the sorting.

Input

The first line contains a single integer n ($1 \le n \le 10^5$), which represents how many numbers the array has. The next line contains n space-separated integers — the array's description. All elements of the array lie in the range from 1 to 10^9, inclusive.

Output

Print n space-separated integers — the minimum possible values of each array element after one replacement and the sorting are performed.

Examples

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

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

input
3 2 2 2
output
1 2 2

D. Rectangle and Square

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

Little Petya very much likes rectangles and especially squares. Recently he has received 8 points on the plane as a gift from his mother. The points are pairwise distinct. Petya decided to split them into two sets each containing 4 points so that the points from the first set lay at the vertexes of some square and the points from the second set lay at the vertexes of a rectangle. Each point of initial 8 should belong to exactly one set. It is acceptable for a rectangle from the second set was also a square. If there are several partitions, Petya will be satisfied by any of them. Help him find such partition. Note that the rectangle and the square from the partition should have non-zero areas. The sides of the figures **do not have** to be parallel to the coordinate axes, though it might be the case.

Input

You are given 8 pairs of integers, a pair per line — the coordinates of the points Petya has. The absolute value of all coordinates does not exceed 10^4. It is guaranteed that no two points coincide.

Output

Print in the first output line "YES" (without the quotes), if the desired partition exists. In the second line output 4 space-separated numbers — point indexes from the input, which lie at the vertexes of the square. The points are numbered starting from 1. The numbers can be printed in any order. In the third line print the indexes of points lying at the vertexes of a rectangle in the similar format. All printed numbers should be pairwise distinct.

If the required partition does not exist, the first line should contain the word "NO" (without the quotes), after which no output is needed.

Examples

input
0 0 10 11 10 0 0 11 1 1 2 2 2 1 1 2
output
YES 5 6 7 8 1 2 3 4

input
0 0 1 1 2 2 3 3 4 4 5 5 6 6 7 7
output
NO

input
0 0 4 4 4 0 0 4 1 2 2 3 3 2 2 1
output
YES 1 2 3 4 5 6 7 8

Note

Pay attention to the third example: the figures do not necessarily have to be parallel to the coordinate axes.

E. Zero-One

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

Little Petya very much likes playing with little Masha. Recently he has received a game called "Zero-One" as a gift from his mother. Petya immediately offered Masha to play the game with him.

Before the very beginning of the game several cards are lain out on a table in one line from the left to the right. Each card contains a digit: 0 or 1. Players move in turns and Masha moves first. During each move a player should remove a card from the table and shift all other cards so as to close the gap left by the removed card. For example, if before somebody's move the cards on the table formed a sequence 01**010**101, then after the fourth card is removed (the cards are numbered starting from 1), the sequence will look like that: 01**00**101.

The game ends when exactly two cards are left on the table. The digits on these cards determine the number in binary notation: the most significant bit is located to the left. Masha's aim is to minimize the number and Petya's aim is to maximize it.

An unpleasant accident occurred before the game started. The kids spilled juice on some of the cards and the digits on the cards got blurred. Each one of the spoiled cards could have either 0 or 1 written on it. Consider all possible variants of initial arrangement of the digits (before the juice spilling). For each variant, let's find which two cards are left by the end of the game, assuming that both Petya and Masha play optimally. An ordered pair of digits written on those two cards is called an *outcome*. Your task is to find the set of outcomes for all variants of initial digits arrangement.

Input

The first line contains a sequence of characters each of which can either be a "0", a "1" or a "?". This sequence determines the initial arrangement of cards on the table from the left to the right. The characters "?" mean that the given card was spoiled before the game. The sequence's length ranges from 2 to 10^5, inclusive.

Output

Print the set of outcomes for all possible initial digits arrangements. Print each possible outcome on a single line. Each outcome should be represented by two characters: the digits written on the cards that were left by the end of the game. The outcomes should be sorted lexicographically in ascending order (see the first sample).

Examples

input
????
output
00 01 10 11

input
1010
output
10

input
1?1
output
01 11

Note

In the first sample all 16 variants of numbers arrangement are possible. For the variant 0000 the outcome is 00. For the variant 1111 the outcome is 11. For the variant 0011 the outcome is 01. For the variant 1100 the outcome is 10. Regardless of outcomes for all other variants the set which we are looking for will contain all 4 possible outcomes.

In the third sample only 2 variants of numbers arrangement are possible: 111 and 101. For the variant 111 the outcome is 11. For the variant 101 the outcome is 01, because on the first turn Masha can remove the first card from the left after which the game will end.