Surprise Language Round #8

A. The Check of the Point

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

On the coordinate plane there is a square with sides parallel to the coordinate axes. The length of the square side is equal to a. The lower left corner of the square coincides with the point $(0, 0)$ (the point of the origin). The upper right corner of the square has positive coordinates.

You are given a point with coordinates (x, y). Your task is to determine whether this point is located strictly inside the square, on its side, or strictly outside the square.

Input

The first line contains three integers a, x and y $(1 \le a \le 1000, \ -1000 \le x, y \le 1000)$ — the length of the square side and the coordinates of the point which should be checked.

Output

Print one integer:

- 0, if the point is located strictly inside the square;
- 1, if the point is located on the side of the square;
- 2, if the point is located strictly outside the square.

Examples

input
2 1 1
output
0

input
4 4 4
output
1

input
10 5 -4
output
2

B. The Teacher of Physical Education

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

n pupils came to Physical Education lesson. We know the name and the height of each pupil.

Your task is to help the teacher of Physical Education to line up all pupils in non-decreasing order of their heights.

Input

The first line contains the positive integer n $(1 \le n \le 5000)$ — the number of pupils.

The next n lines contain the pupils' description. In the i-th line there is pupil's name $name_i$ (a non-empty string which consists of uppercase and lowercase Latin letters, the length does not exceed five) and pupil's height x_i $(130 \le x_i \le 215)$. Some pupils can have the same name. Uppercase and lowercase letters of the alphabet should be considered different.

Output

Print n lines — pupils' names in the non-decreasing order of their heights. Each line must contain exactly one name.

If there are several answers, print any of them. Uppercase and lowercase letters of the alphabet should be considered different.

Examples

input
4 Ivan 150 Igor 215 Dasha 158 Katya 150
output
Ivan Katya Dasha Igor

input
2 SASHA 180 SASHA 170
output
SASHA SASHA

C. Symmetric Difference

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

You are given two sets of numbers. Your task is to print all numbers from the sets, that both sets don't contain simultaneously.

Input

The first line contains the description of the first set, the second line contains the description of the second set. Each description begins with the number of elements in this set. All elements of the set follow in the arbitrary order. In each set all elements are distinct and both sets are not empty. The number of elements in each set doesn't exceed 1000. All elements of the sets are integers from -1000 to 1000.

Output

Print the number of the required numbers and then the numbers themselves separated by a space.

Examples

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

input
5 1 4 8 9 10 4 1 2 8 10
output
3 2 4 9

D. Chocolate Bar

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

A chocolate bar has a rectangular shape and consists of $n \times m$ slices. In other words, a bar consists of n rows with m slices of chocolate in each row.

Each slice of chocolate is known to weigh 1 gram. Your task is to determine for each of the q chocolate bars whether it is possible to obtain a piece weighing p grams by breaking the bar several (possibly zero) times. The final piece of the chocolate bar should be whole, and breaks are made along the line of slices' section for the whole length of the current piece.

Input

The first line contains the positive integer q $(1 \le q \le 100)$ — the number of chocolate bars.

Each of the following q lines contains three positive integers n, m and p $(1 \le n, m, p \le 1000)$ — the size of the chocolate bar, and the weight of the piece which should be obtained.

Output

The output should contain q lines and the i-th line must contain "`Yes`" (without the quotes), if it is possible to perform the task for i-th chocolate bar, or "`No`" otherwise.

Example

input
2 3 3 4 4 4 7
output
Yes No

E. Hammer throwing

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

n athletes take part in the hammer throw. Each of them has his own unique identifier — the integer from 1 to n (all athletes have distinct identifiers). After the draw, the order in which the athletes will throw the hammer has been determined (they will do it one by one).

Unfortunately, a not very attentive judge lost the list with the order of athletes, but each of the athletes has remembered how many competitors with identifiers larger than his own will throw the hammer before him.

Your task is to help the organizers as quickly as possible to restore the order of the athletes.

Input

The first line contains the positive integer n ($1 \le n \le 1000$) — the number of athletes.

The next line contains the sequence of integers $a_1, a_2, ..., a_n$ ($0 \le a_i < n$), where a_i is equal to the number of the athletes with identifiers larger than i, who should throw the hammer before the athlete with identifier i.

Output

Print n distinct numbers — the sequence of athletes' identifiers in the order in which they will throw the hammer. If there are several answers it is allowed to print any of them.

Examples

input
4 2 0 1 0
output
2 4 1 3

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

F. Reformat the String

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

In this problem you are given a string s consisting of uppercase and lowercase Latin letters, spaces, dots and commas. Your task is to correct the formatting of this string by removing and inserting spaces, as well as changing the case of the letters.

After formatting, the resulting string must meet the following requirements:

- the string must not start with a space;
- there should be exactly one space between any two consecutive words;
- there should be a Latin letter immediately before a dot or a comma, and there should be a space immediately after a dot or a comma in the case that there is a word after that dot or comma, otherwise that dot or comma must be the last character in the string;
- all letters must be lowercase, except all first letters in the first words of the sentences, they must be capitalized. The first word of a sentence is a word that is either the first word of the string or a word after a dot.

It is guaranteed that there is at least one letter in the given string between any two punctuation marks (commas and dots are punctuation marks). There is at least one letter before the leftmost punctuation mark.

Input

The first line contains a non-empty string s, consisting of uppercase and lowercase Latin letters, spaces, dots and commas. The length of the given string does not exceed 255. The string is guaranteed to have at least one character other than the space.

Output

Output the corrected string which meets all the requirements described in the statement.

Examples

input
hello ,i AM veRy GooD.Boris
output
Hello, i am very good. Boris

input
a. b, C .
output
A. B, c.

G. The Fraction

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

Periodic decimal fraction is usually written as: `[entire_part.non-periodic_part (period)]`. Any simple fraction can be represented as a periodic decimal fraction and vice versa. For example, the decimal fraction $0.2(45)$ corresponds to a fraction $27 / 110$. Your task is to convert the periodic fraction to a simple periodic fraction.

Input

The first line contains the periodic decimal fraction x ($0 < x < 1$) in the format described in the statement. The total number of digits in the period and non-periodic part of the fraction does not exceed 8. Non-periodic part may be absent, the periodic part can't be absent (but it can be equal to any non-negative number).

Output

Print the representation of the fraction x as a simple fraction p / q, where p and q are mutually prime integers.

Examples

input
0.2(45)
output
27/110

input
0.75(0)
output
3/4

H. Exchange of Books

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

n pupils, who love to read books, study at school. It is known that each student has exactly one best friend, and each pupil is the best friend of exactly one other pupil. Each of the pupils has exactly one interesting book.

The pupils decided to share books with each other. Every day, all pupils give their own books to their best friends. Thus, every day each of the pupils has exactly one book.

Your task is to use the list of the best friends and determine the exchange of books among pupils after k days. For simplicity, all students are numbered from 1 to n in all tests.

Input

The first line contains two integers n and k ($2 \le n \le 100000$, $1 \le k \le 10^{16}$) — the number of pupils and days during which they will exchange books.

The second line contains n different integers a_i ($1 \le a_i \le n$), where a_i is equal to the number of the pupil who has the best friend with the number i.

It is guaranteed that no pupil is the best friend of himself.

Output

In a single line print n different integers, where i-th integer should be equal to the number of the pupil who will have the book, which the pupil with the number i had in the beginning, after k days.

Examples

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

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

input
6 18 2 3 5 1 6 4
output
1 2 3 4 5 6

Note

The explanation to the first test.

There are 4 pupils and 1 day. The list of the best friends equals to {2, 4, 1, 3}. It means that:

- the pupil with the number 3 — is the best friend of pupil with the number 1,
- the pupil with the number 1 — is the best friend of pupil with the number 2,
- the pupil with the number 4 — is the best friend of pupil with the number 3,
- the pupil with the number 2 — is the best friend of pupil with the number 4.

After the first day the exchange of books will be {3, 1, 4, 2}.

- the pupil with the number 3 will have the book, which the pupil with the number 1 had in the beginning,
- the pupil with the number 1 will have the book, which the pupil with the number 2 had in the beginning,
- the pupil with the number 4 will have the book, which the pupil with the number 3 had in the beginning
- the pupil with the number 2 will have the book, which the pupil with the number 4 had in the beginning.

Thus, the answer is **3 1 4 2**.

I. Loader

time limit per test: 4 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

A loader works in a warehouse, which is a rectangular field with size $n \times m$. Some cells of this field are free, others are occupied by pillars on which the roof of the warehouse rests.

There is a load in one of the free cells, and the loader in another. At any moment, the loader and the load can not be in the cells with columns, outside the warehouse or in the same cell.

The loader can move to the adjacent cell (two cells are considered adjacent if they have a common side), or move the load. To move the load, the loader should reach the cell adjacent to the load and push the load. In this case the load advances to the next cell in the direction in which the loader pushes it and the loader ends up in the cell in which the load was.

Your task is to determine a sequence of pushes and loader's movements after which the load will reach the given cell (it is guaranteed that this cell is free). The load is rather heavy, so you need to minimize first the number of pushes and second the number of loader's movements.

Input

The first line contains two positive integers n and m ($1 \le n, m \le 40$, $n \cdot m \ge 3$) — the number of rows and columns in the rectangular field.

Each of the next n lines contains m characters — the description of the warehouse. If there is a character in the next cell of the warehouse:

- "`X`", it means, that the current cell contains the column;
- "`.`", it means, that the current cell is free;
- "`Y`", it means, that the loader is in the current cell;
- "`B`", it means, that the load is in the current cell;
- "`T`", it means, that the load should be moved to this cell.

It is guaranteed that there is exactly one load, one loader and one cell to which the load should be moved.

Output

If the loader is not able to move the load to the given cell, print "`NO`" (without the quotes) in the first line of the output.

Otherwise, print "`YES`" (without the quotes) in the first line of the output, and in the second line — the sequence of characters that determines movements and pushes of the loader. Characters `w`, `e`, `n`, `s` shall denote loader's moves to the west, east, north and south, respectively. Characters `W`, `E`, `N`, `S` must denote loader's pushes in the corresponding directions. First of all you need to minimize the number of pushes of the load and second, the number of movements of the loader. If there are several answers, you are allowed to print any of them.

Examples

input
3 3 ..Y .BX ..T
output
YES wSwsE

input
3 3 .BY ... TXX
output
NO

J. The Hero with Bombs

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

In a new computer game you need to help the hero to get out of the maze, which is a rectangular field of size $n \times m$. The hero is located in one of the cells of this field. He knows where the exit of the maze is, and he wants to reach it.

In one move, the hero can either move to the next cell (i.e. the cell which has a common side with the current cell) if it is free, or plant a bomb on the cell where he is, or skip the move and do nothing. A planted bomb explodes after three moves, that is, after the hero makes 3 more actions but does not have time to make the fourth (all three types of moves described above are considered as actions).

The explosion destroys the obstacles in all the cells which have at least one common point with this cell (i.e. in all the cells sharing with the bomb cell a corner or a side). The explosion must not hurt the cell with the exit or the cell with the hero. The hero can not go beyond the boundaries of the maze.

Your task is to determine the sequence of hero's actions to reach the exit. Note that you haven't to minimize the length of the sequence. The only restriction — the length of the resulting sequence should not exceed 100,000 symbols.

Input

The first line contains two integers n and m ($1 \le n, m \le 100$, $n \cdot m > 1$) — sizes of the maze.

Each of the following n lines contains m characters — description of the maze. The character "`.`" means a free cell, "`E`" — the hero, "`T`" — the exit, "`X`" — the obstacle.

It is guaranteed that there is exactly one hero and exactly one exit in the maze.

Output

Print the hero's actions which will help him get out of the maze ("`M`" — to plant a bomb, "`T`" — to skip the move, "`S`" — to go down, "`W`" — to go left, "`N`" — to go up, "`E`" — to go right). If the hero can not reach the exit, print "`No solution`" (without quotes).

The length of the resulting sequence should not exceed 100,000 symbols. If there are several solutions it is allowed to print any of them.

Example

input
3 5 XEX.X X.XXT X.X.X
output
SSMNNTSSNEMWWTEEEE