Codeforces β Sponsored by Telegram

Coder-Strike 2014 - Round 1

A. Poster

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

The R1 company has recently bought a high rise building in the centre of Moscow for its main office. It's time to decorate the new office, and the first thing to do is to write the company's slogan above the main entrance to the building.

The slogan of the company consists of n characters, so the decorators hung a large banner, n meters wide and 1 meter high, divided into n equal squares. The first character of the slogan must be in the first square (the leftmost) of the poster, the second character must be in the second square, and so on.

Of course, the R1 programmers want to write the slogan on the poster themselves. To do this, they have a large (and a very heavy) ladder which was put exactly opposite the k-th square of the poster. To draw the i-th character of the slogan on the poster, you need to climb the ladder, standing in front of the i-th square of the poster. This action (along with climbing up and down the ladder) takes one hour for a painter. The painter is not allowed to draw characters in the adjacent squares when the ladder is in front of the i-th square because the uncomfortable position of the ladder may make the characters untidy. Besides, the programmers can move the ladder. In one hour, they can move the ladder either a meter to the right or a meter to the left.

Drawing characters and moving the ladder is very tiring, so the programmers want to finish the job in as little time as possible. Develop for them an optimal poster painting plan!

Input

The first line contains two integers, n and k $(1 \leq k \leq n \leq 100)$ — the number of characters in the slogan and the initial position of the ladder, correspondingly. The next line contains the slogan as n characters written without spaces. Each character of the slogan is either a large English letter, or digit, or one of the characters: '.', '!', ',', '?'.

Output

In t lines, print the actions the programmers need to make. In the i-th line print:

- "LEFT" (without the quotes), if the i-th action was "move the ladder to the left";
- "RIGHT" (without the quotes), if the i-th action was "move the ladder to the right";
- "PRINT x" (without the quotes), if the i-th action was to "go up the ladder, paint character x, go down the ladder".

The painting time (variable t) must be minimum possible. If there are multiple optimal painting plans, you can print any of them.

Examples

input

```
2 2
R1
```

output

```
PRINT 1
LEFT
PRINT R
```

input

```
2 1
R1
```

output

```
PRINT R
RIGHT
PRINT 1
```

input

```
6 4
GO?GO!
```

output

```
RIGHT
RIGHT
PRINT !
```

```
LEFT
PRINT O
LEFT
PRINT G
LEFT
PRINT ?
LEFT
PRINT O
LEFT
PRINT G
```

Note

Note that the ladder cannot be shifted by less than one meter. The ladder can only stand in front of some square of the poster. For example, you cannot shift a ladder by half a meter and position it between two squares. Then go up and paint the first character and the second character.

B. Network Configuration

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

The R1 company wants to hold a web search championship. There were n computers given for the competition, each of them is connected to the Internet. The organizers believe that the data transfer speed directly affects the result. The higher the speed of the Internet is, the faster the participant will find the necessary information. Therefore, before the competition started, each computer had its maximum possible data transfer speed measured. On the i-th computer it was a_i kilobits per second.

There will be k participants competing in the championship, each should get a separate computer. The organizing company does not want any of the participants to have an advantage over the others, so they want to provide the same data transfer speed to each participant's computer. Also, the organizers want to create the most comfortable conditions for the participants, so the data transfer speed on the participants' computers should be as large as possible.

The network settings of the R1 company has a special option that lets you to cut the initial maximum data transfer speed of any computer to any lower speed. How should the R1 company configure the network using the described option so that at least k of n computers had the same data transfer speed and the data transfer speed on these computers was as large as possible?

Input

The first line contains two space-separated integers n and k $(1 \le k \le n \le 100)$ — the number of computers and the number of participants, respectively. In the second line you have a space-separated sequence consisting of n integers: $a_1, a_2, \dots, a_n$ $(16 \le a_i \le 32768)$; number a_i denotes the maximum data transfer speed on the i-th computer.

Output

Print a single integer — the maximum Internet speed value. It is guaranteed that the answer to the problem is always an integer.

Examples

input

```
3 2
40 20 30
```

output

```
30
```

input

```
6 4
100 20 40 20 50 50
```

output

```
40
```

Note

In the first test case the organizers can cut the first computer's speed to 30 kilobits. Then two computers (the first and the third one) will have the same speed of 30 kilobits. They should be used as the participants' computers. This answer is optimal.

C. Pattern

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

Developers often face with regular expression patterns. A pattern is usually defined as a string consisting of characters and metacharacters that sets the rules for your search. These patterns are most often used to check whether a particular string meets the certain rules.

In this task, a pattern will be a string consisting of small English letters and question marks ('?'). The question mark in the pattern is a metacharacter that denotes an arbitrary small letter of the English alphabet. We will assume that a string matches the pattern if we can transform the string into the pattern by replacing the question marks by the appropriate characters. For example, string aba matches patterns: ???, ??a, a?a, aba.

Programmers that work for the R1 company love puzzling each other (and themselves) with riddles. One of them is as follows: you are given n patterns of the same length, you need to find a pattern that contains as few question marks as possible, and intersects with each of the given patterns. Two patterns intersect if there is a string that matches both the first and the second pattern. Can you solve this riddle?

Input

The first line contains a single integer n ($1 \le n \le 10^5$) — the number of patterns. Next n lines contain the patterns.

It is guaranteed that the patterns can only consist of small English letters and symbols '?'. All patterns are non-empty and have the same length. The total length of all the patterns does not exceed 10^5 characters.

Output

In a single line print the answer to the problem — the pattern with the minimal number of signs '?', which intersects with each of the given ones. If there are several answers, print any of them.

Examples

input

```
2
?ab
??b
```

output

```
xab
```

input

```
2
a
b
```

output

```
?
```

input

```
1
?a?b
```

output

```
cacb
```

Note

Consider the first example. Pattern xab intersects with each of the given patterns. Pattern ??? also intersects with each of the given patterns, but it contains more question signs, hence it is not an optimal answer. Clearly, xab is the optimal answer, because it doesn't contain any question sign. There are a lot of other optimal answers, for example: aab, bab, cab, dab and so on.

D. Giving Awards

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

The employees of the R1 company often spend time together: they watch football, they go camping, they solve contests. So, it's no big deal that sometimes someone pays for someone else.

Today is the day of giving out money rewards. The R1 company CEO will invite employees into his office one by one, rewarding each one for the hard work this month. The CEO knows who owes money to whom. And he also understands that if he invites person x to his office for a reward, and then immediately invite person y, who has lent some money to person x, then they can meet. Of course, in such a situation, the joy of person x from his brand new money reward will be much less. Therefore, the R1 CEO decided to invite the staff in such an order that the described situation will not happen for any pair of employees invited one after another.

However, there are a lot of employees in the company, and the CEO doesn't have a lot of time. Therefore, the task has been assigned to you. Given the debt relationships between all the employees, determine in which order they should be invited to the office of the R1 company CEO, or determine that the described order does not exist.

Input

The first line contains space-separated integers n and m $(2 \le n \le 3 \cdot 10^4; 1 \le m \le min(10^5, \frac{n(n-1)}{2}))$ — the number of employees in R1 and the number of debt relations. Each of the following m lines contains two space-separated integers a_i, b_i $(1 \le a_i, b_i \le n; a_i \ne b_i)$, these integers indicate that the person number a_i owes money to a person a number b_i. Assume that all the employees are numbered from 1 to n.

It is guaranteed that each pair of people p, q is mentioned in the input data at most once. In particular, the input data will not contain pairs p, q and q, p simultaneously.

Output

Print -1 if the described order does not exist. Otherwise, print the permutation of n distinct integers. The first number should denote the number of the person who goes to the CEO office first, the second number denote the person who goes second and so on.

If there are multiple correct orders, you are allowed to print any of them.

Examples

input

```
2 1
1 2
```

output

```
2 1
```

input

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

output

```
2 1 3
```

E. E-mail Addresses

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

One of the most important products of the R1 company is a popular @r1.com mail service. The R1 mailboxes receive and send millions of emails every day.

Today, the online news thundered with terrible information. The R1 database crashed and almost no data could be saved except for one big string. The developers assume that the string contains the letters of some users of the R1 mail. Recovering letters is a tedious mostly manual work. So before you start this process, it was decided to estimate the difficulty of recovering. Namely, we need to calculate the number of different substrings of the saved string that form correct e-mail addresses.

We assume that valid addresses are only the e-mail addresses which meet the following criteria:

- the address should begin with a non-empty sequence of letters, numbers, characters '_', starting with a letter;
- then must go character '@';
- then must go a non-empty sequence of letters or numbers;
- then must go character '.';
- the address must end with a non-empty sequence of letters.

You got lucky again and the job was entrusted to you! Please note that the substring is several consecutive characters in a string. Two substrings, one consisting of the characters of the string with numbers $l_1, l_1 + 1, l_1 + 2, \ldots, r_1$ and the other one consisting of the characters of the string with numbers $l_2, l_2 + 1, l_2 + 2, \ldots, r_2$, are considered distinct if $l_1 \neq l_2$ or $r_1 \neq r_2$.

Input

The first and the only line contains the sequence of characters $s_1s_2... s_n$ $(1 \leq n \leq 10^6)$ — the saved string. It is guaranteed that the given string contains only small English letters, digits and characters '.', '_', '@'.

Output

Print in a single line the number of substrings that are valid e-mail addresses.

Examples

input

```
gerald.agapov1991@gmail.com
```

output

```
18
```

input

```
x@x.x@x.x_e_@r1.com
```

output

```
8
```

input

```
a___@1.r
```

output

```
1
```

input

```
.asd123__..@
```

output

```
0
```

Note

In the first test case all the substrings that are correct e-mail addresses begin from one of the letters of the word agapov and end in one of the letters of the word com.

In the second test case note that the e-mail x@x.x is considered twice in the answer. Note that in this example the e-mail entries overlap inside the string.