# A. Bar

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

According to Berland laws it is only allowed to sell alcohol to people **not younger than** 18 years. Vasya's job is to monitor the law's enforcement. Tonight he entered a bar and saw $n$ people sitting there. For every one of them Vasya happened to determine either the age or the drink the person is having. Vasya can check any person, i.e. learn his age and the drink he is having at the same time. What minimal number of people should Vasya check additionally to make sure that there are no clients under 18 having alcohol drinks?

The list of all alcohol drinks in Berland is: ABSINTH, BEER, BRANDY, CHAMPAGNE, GIN, RUM, SAKE, TEQUILA, VODKA, WHISKEY, WINE

## Input
The first line contains an integer $n$ ($1 \le n \le 100$) which is the number of the bar's clients. Then follow $n$ lines, each describing one visitor. A line either contains his age (an integer from 0 to 1000) or his drink (a string of capital Latin letters from 1 to 100 in length). It is guaranteed that the input data does not contain spaces and other unnecessary separators.

Only the drinks from the list given above should be considered alcohol.

## Output
Print a single number which is the number of people Vasya should check to guarantee the law enforcement.

**Examples**

| input |
| --- |
| 5<br>18<br>VODKA<br>COKE<br>19<br>17 |
| **output** |
| 2 |

## Note
In the sample test the second and fifth clients should be checked.

# B. Spoilt Permutation

Vasya collects coins: he has exactly one coin for every year from $1$ to $n$. Naturally, Vasya keeps all the coins in his collection in the order in which they were released. Once Vasya's younger brother made a change — he took all the coins whose release year dated from $l$ to $r$ inclusively and put them in the reverse order. That is, he took a certain segment $[l, r]$ and reversed it. At that the segment's endpoints did not coincide. For example, if $n = 8$, then initially Vasya's coins were kept in the order $1\ 2\ 3\ 4\ 5\ 6\ 7\ 8$. If Vasya's younger brother chose the segment $[2, 6]$, then after the reversal the coin order will change to $1\ 6\ 5\ 4\ 3\ 2\ 7\ 8$. Vasya suspects that someone else could have spoilt the permutation after his brother. Help him to find that out. Check if the given permutation can be obtained from the permutation $1\ 2\ \ldots\ n$ using exactly one segment reversal. If it is possible, find the segment itself.

## Input

The first line contains an integer $n$ ($1 \leq n \leq 1000$) which is the number of coins in Vasya's collection. The second line contains space-separated $n$ integers which are the spoilt sequence of coins. It is guaranteed that the given sequence is a permutation, i.e. it contains only integers from $1$ to $n$, and every number is used exactly 1 time.

## Output

If it is impossible to obtain the given permutation from the original one in exactly one action, print $0\ \ 0$. Otherwise, print two numbers $l\ r$ ($1 \leq l < r \leq n$) which are the endpoints of the segment that needs to be reversed to obtain from permutation $1\ 2\ \ldots\ n$ the given one.

## Examples

| input |
|---|
| 8<br>1 6 5 4 3 2 7 8 |
| **output** |
| 2 6 |

| input |
|---|
| 4<br>2 3 4 1 |
| **output** |
| 0 0 |

| input |
|---|
| 4<br>1 2 3 4 |
| **output** |
| 0 0 |

# C. Corporation Mail

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

The Beroil corporation structure is hierarchical, that is it can be represented as a tree. Let's examine the presentation of this structure as follows:

- $employee ::= name\,.\,|\,name : employee_1 , employee_2 , \ldots , employee_k\,.$
- $name ::=$ name of an employee

That is, the description of each employee consists of his name, a colon (`:`), the descriptions of all his subordinates separated by commas, and, finally, a dot. If an employee has no subordinates, then the colon is not present in his description.

For example, line `MIKE:MAX.,ARTEM:MIKE..,DMITRY:DMITRY.,DMITRY...` is the correct way of recording the structure of a corporation where the director `MIKE` has subordinates `MAX`, `ARTEM` and `DMITRY`. `ARTEM` has a subordinate whose name is `MIKE`, just as the name of his boss and two subordinates of `DMITRY` are called `DMITRY`, just like himself.

In the Beroil corporation every employee can only correspond with his subordinates, at that the subordinates are not necessarily direct. Let's call an uncomfortable situation the situation when a person whose name is $S$ writes a letter to another person whose name is also $S$. In the example given above are two such pairs: a pair involving `MIKE`, and two pairs for `DMITRY` (a pair for each of his subordinates).

Your task is by the given structure of the corporation to find the number of uncomfortable pairs in it.

## Input

The first and single line contains the corporation structure which is a string of length from 1 to 1000 characters. It is guaranteed that the description is correct. Every name is a string consisting of capital Latin letters from 1 to 10 symbols in length.

## Output

Print a single number — the number of uncomfortable situations in the company.

## Examples

| input |
|---|
| MIKE:MAX.,ARTEM:MIKE..,DMITRY:DMITRY.,DMITRY... |
| **output** |
| 3 |

| input |
|---|
| A:A.. |
| **output** |
| 1 |

| input |
|---|
| A:C:C:C:C..... |
| **output** |
| 6 |

# D. Changing a String

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

There is a string $s$, consisting of capital Latin letters. Let's denote its current length as $|s|$. During one move it is allowed to apply one of the following operations to it:

- INSERT $pos$ $ch$ — insert a letter $ch$ in the string $s$ in the position $pos$ ($1 \le pos \le |s| + 1, A \le ch \le Z$). The letter $ch$ becomes the $pos$-th symbol of the string $s$, at that the letters shift aside and the length of the string increases by 1.
- DELETE $pos$ — delete a character number $pos$ ($1 \le pos \le |s|$) from the string $s$. At that the letters shift together and the length of the string decreases by 1.
- REPLACE $pos$ $ch$ — the letter in the position $pos$ of the line $s$ is replaced by $ch$ ($1 \le pos \le |s|, A \le ch \le Z$). At that the length of the string does not change.

Your task is to find in which minimal number of moves one can get a $t$ string from an $s$ string. You should also find the sequence of actions leading to the required results.

## Input

The first line contains $s$, the second line contains $t$. The lines consist only of capital Latin letters, their lengths are positive numbers from $1$ to $1000$.

## Output

In the first line print the number of moves $k$ in the given sequence of operations. The number should be the minimal possible one. Then print $k$ lines containing one operation each. Print the operations in the format, described above. If there are several solutions, print any of them.

## Examples

| input |
|---|
| ABA<br>ABBBA |
| **output** |
| 2<br>INSERT 3 B<br>INSERT 4 B |

| input |
|---|
| ACCEPTED<br>WRONGANSWER |
| **output** |
| 10<br>REPLACE 1 W<br>REPLACE 2 R<br>REPLACE 3 O<br>REPLACE 4 N<br>REPLACE 5 G<br>REPLACE 6 A<br>INSERT 7 N<br>INSERT 8 S<br>INSERT 9 W<br>REPLACE 11 R |

# E. Domino Principle

Vasya is interested in arranging dominoes. He is fed up with common dominoes and he uses the dominoes of different heights. He put $n$ dominoes on the table along one axis, going from left to right. Every domino stands perpendicular to that axis so that the axis passes through the center of its base. The $i$-th domino has the coordinate $x_i$ and the height $h_i$. Now Vasya wants to learn for every domino, how many dominoes will fall if he pushes it to the right. Help him do that.

Consider that a domino falls if it is touched strictly above the base. In other words, the fall of the domino with the initial coordinate $x$ and height $h$ leads to the fall of all dominoes on the segment $[x + 1, x + h - 1]$.

## Input

The first line contains integer $n$ ($1 \le n \le 10^5$) which is the number of dominoes. Then follow $n$ lines containing two integers $x_i$ and $h_i$ ($-10^8 \le x_i \le 10^8, 2 \le h_i \le 10^8$) each, which are the coordinate and height of every domino. No two dominoes stand on one point.

## Output

Print $n$ space-separated numbers $z_i$ — the number of dominoes that will fall if Vasya pushes the $i$-th domino to the right (including the domino itself).

## Examples

| input |
|---|
| 4<br>16 5<br>20 5<br>10 10<br>18 2 |
| **output** |
| 3 1 4 1 |

| input |
|---|
| 4<br>0 10<br>1 5<br>9 10<br>15 10 |
| **output** |
| 4 1 2 1 |

---