COCI 2011/2012
5th round, March 17th, 2012

Task KRIŽALJKA
Author: Bruno Rahle

Since ACTA has entered into force, Slavko has been spending his time offline, solving crosswords. Having solved almost all that he could get his hands on, he wants to make a few crosswords of his own. However, he is too sloppy for such fine work, so he has asked you to help him generate the crosswords.

You are given two words, **A** and **B**. The word **A** must be output **horizontally**, and the word **B** **vertically**, so that the two words cross (i.e. share exactly one letter). The shared letter must be the first letter in **A** that is also contained in **B**, more specifically the first occurrence of that letter in each word.

For example, given the words **A** = “ABBA” and **B** = “CCBB”, you need to output 4 lines as shown below:

```
.C..
.C..
ABBA
.B..
```

INPUT

The first and only line of input contains two words, **A** and **B**, not more than 30 characters long, separated by a single space. Both words will contain only uppercase English letters. There will be at least one letter contained in both words.

OUTPUT

Let **N** be the length of word **A**, and **M** the length of word **B**. The output must contain **M** lines, each containing **N** characters. The character grid must contain the two words crossed as described above. All other characters in the grid must be periods (the character ‘.’, without quotes), thus padding all lines to the length of **N** characters.

SAMPLE TESTS

input	**input**	**input**
BANANA PIDZAMA	MAMA TATA	REPUBLIKA HRVATSKA
output	**output**	**output**
.P.... .I.... .D.... .Z.... BANANA .M.... .A....	.T.. MAMA .T.. .A..	H........ REPUBLIKA V........ A........ T........ S........ K........ A........

Lumberjack Mirko needs to chop down **M** metres of wood. It is an easy job for him since he has a nifty new woodcutting machine that can take down forests like wildfire. However, Mirko is only allowed to cut a single row of trees.

Mirko's machine works as follows: Mirko sets a height parameter **H** (in metres), and the machine raises a giant sawblade to that height and cuts off all tree parts higher than **H** (of course, trees not higher than **H** meters remain intact). Mirko then takes the parts that were cut off. For example, if the tree row contains trees with heights of 20, 15, 10, and 17 metres, and Mirko raises his sawblade to 15 metres, the remaining tree heights after cutting will be 15, 15, 10, and 15 metres, respectively, while Mirko will take 5 metres off the first tree and 2 metres off the fourth tree (7 metres of wood in total).

Mirko is **ecologically** minded, so he doesn't want to cut off more wood than necessary. That's why he wants to set his sawblade as high as possible. Help Mirko find the **maximum integer height** of the sawblade that still allows him to cut off **at least M** metres of wood.

INPUT

The first line of input contains two space-separated positive integers, **N** (the number of trees, 1 ≤ **N** ≤ 1 000 000) and **M** (Mirko's required wood amount, 1 ≤ **M** ≤ 2 000 000 000).

The second line of input contains **N** space-separated positive integers less than 1 000 000 000, the heights of each tree (in metres). The sum of all heights will exceed **M**, thus Mirko will always be able to obtain the required amount of wood.

OUTPUT

The first and only line of output must contain the required height setting.

SAMPLE TESTS

input 4 7 20 15 10 17 **output** 15	**input** 5 20 4 42 40 26 46 **output** 36

Task DNA
Author: Adrian Satja Kurdija

Biologists have discovered a strange DNA molecule, best described as a sequence of **N** characters from the set {A, B}. An unlikely sequence of mutations has resulted in a DNA strand consisting only of A's. Biologists found that very odd, so they began studying the mutations in greater detail.

They discovered two types of mutations. One type results in changing a single character of the sequence (A → B or B → A). The second type changes a whole **prefix** of the sequence, specifically replacing all characters in positions from 1 to **K** (for some **K** between 1 and **N**, inclusive) with the other character (A with B, B with A).

Compute the least possible number of mutations that could convert the starting molecule to its end state (containing only A characters). Mutations can occur in any order.

INPUT

The first line of input contains the positive integer **N** (1 ≤ **N** ≤ 1 000 000), the length of the molecule.

The second line of input contains a string with **N** characters, with each character being either A or B. This string represents the starting state of the molecule.

OUTPUT

The first and only line of output must contain the required minimum number of mutations.

SAMPLE TESTS

input 4 ABBA **output** 2	**input** 5 BBABB **output** 2	**input** 12 AAABBBAAABBB **output** 4

In the popular show "Dinner for Five", five contestants compete in preparing culinary delights. Every evening one of them makes dinner and each of other four then grades it on a scale from 1 to 5.

The number of points a contestant gets is equal to the sum of grades they got. The winner of the show is of course the contestant that gets the most points.

Write a program that determines the winner and how many points they got.

INPUT

Five lines, each containing 4 integers, the grades a contestant got.

The contestants are numbered 1 to 5 in the order in which their grades were given.

The input data will guarantee that the solution is unique.

OUTPUT

Output on a single line the winner's number and their points, separated by a space.

EXAMPLES

input
 	
`5 4 4 5`
`5 4 4 4`
`5 5 4 4`
`5 5 5 4`
`4 4 4 5`
 	
output
 	
`4 19`
	input
	
	`4 4 3 3`
	`5 4 3 5`
	`5 5 2 4`
	`5 5 5 1`
	`4 4 4 4`
	
	output
	
	`2 17`

Luka is fooling around in chemistry class again! Instead of balancing equations he is writing coded sentences on a piece of paper. Luka modifies every word in a sentence by adding, after each vowel (letters 'a', 'e', 'i', 'o' and 'u'), the letter 'p' and then that same vowel again.

For example, the word "kemija" becomes "kepemipijapa" and the word "paprika" becomes "papapripikapa". The teacher took Luka's paper with the coded sentences and wants to decode them.

Write a program that decodes Luka's sentence.

INPUT

The coded sentence will be given on a single line. The sentence consists only of lowercase letters of the English alphabet and spaces. The words will be separated by exactly one space and there will be no leading or trailing spaces. The total number of character will be at most 100.

OUTPUT

Output the decoded sentence on a single line.

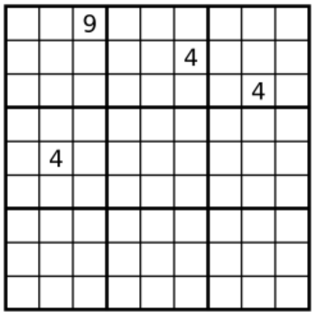
EXAMPLES

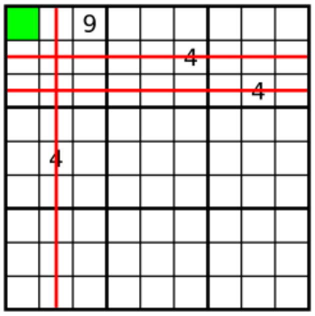
input	**input**
`zepelepenapa papapripikapa`	`bapas jepe doposapadnapa opovapa kepemipijapa`
output	**output**
`zelena paprika`	`bas je dosadna ova kemija`

In the game of Sudoku, the objective is to place integers between 1 and 9 (inclusive) into a 9x9 grid so that each row, each column, and each of the nine 3x3 boxes contains all nine numbers. The starting board is partially filled in so that it is possible to logically deduce the values of the other cells. Sudoku puzzles range in difficulty, and complex analysis methods are required to solve the hardest puzzles. In this problem, however, you will implement one of the simplest methods, cross-hatching.

In cross-hatching, we select one of the nine numbers and, for each of its occurrences in the grid, cross out the corresponding row, column and 3x3 box. Now look for any 3x3 boxes where there is only one possible placement for the number and place it there.

The first image below shows a very sparsely filled in Sudoku grid. However, even in this grid it is possible to deduce using cross-hatching that the number in the top left cell is 4, as illustrated in the second image.





You will be given a partially filled-in grid. Your task is to repeatedly apply the cross-hatching method for different numbers until no more deductions can be made about any number.

The initial placement of the numbers in the grid may be invalid. It is also possible that there will be no available cell for a number in a 3x3 box. In both cases, you are to report an error.

INPUT

Input will consist of 9 lines, each containing exactly 9 characters. Each character will either be a digit between 1 and 9, or a period ('.') denoting an empty cell.

OUTPUT

If the input is valid and there is no contradiction while solving, you should output the grid in the same format it was given in, with cells filled in if their value can be deduced using cross-hatching. Otherwise, output "ERROR" (quotes for clarity).

EXAMPLES

input	**input**	**input**	**input**
..9...... .....4... ....4. .... .4.......	...1...6. 18...9... ..7.642.. 2.9..6.5. .43...72. .6.3..9.1 ..265.1.. ...2...97 .5...3...	1........ ..1...... ....1.	........2 ....1.... 1........ ......1.. ....1. ....
output	**output**	**output**	**output**
4.9...... .....4... ....4. .... .4.......	524137869 186529473 397864215 219476358 843915726 765382941 972658134 638241597 451793682	ERROR	ERROR

A binary search tree is a tree in which every node has **at most** two children nodes (a left and a right child). Each node has an integer written inside it. If the number X is written inside a node, then the numbers in its left subtree are less than X and the numbers in its right subtree are greater than X.

You will be given a sequence of integers between 1 and N (inclusive) such that each number appears in the sequence exactly once. You are to create a binary search tree from the sequence, putting the first number in the root node and inserting every other number in order. In other words, run insert(X, root) for every other number:

```
insert( number X, node N )
    increase the counter C by 1
    if X is less than the number in node N
        if N has no left child
            create a new node with the number X and set it to be the left child of node N
        else
            insert(X, left child of node N)
    else (X is greater than the number in node N)
        if N has no right child
            create a new node with the number X and set it to be the right child of node N
        else
            insert(X, right child of node N)
```

Write a program that calculates the value of the counter C after every number is inserted. The counter is initially 0.

INPUT

The first line contains the integer N ($1 \leq N \leq 300000$), the length of the sequence.

The remaining N lines contain the numbers in the sequence, integers in the interval [1, N]. The numbers will be distinct.

OUTPUT

Output N integers each on its own line, the values of the counter C after each number is inserted into the tree.

SCORING

In test cases worth 50% of points, N will be at most 1000.

EXAMPLES

input 4 1 2 3 4 **output** 0 1 3 6	**input** 5 3 2 4 1 5 **output** 0 1 2 4 6	**input** 8 3 5 1 6 8 7 2 4 **output** 0 1 2 4 7 11 13 15