Tasks

Task	TRI	PASCAL	JABUKE	AVOGADRO	BARICA	BAZA
Input	standard input (keyboard)					
Output	standard output (screen)					
Memory limit (heap+stack)	32 MB					
Time limit (per test)	1 second					2 seconds
Number of tests	5	10	5	10	10	10
Points per test	4	3	6	6	7	9
Total points	**20**	**30**	**30**	**60**	**70**	**90**
	300					

Note: The evaluation system has two Intel Pentium 4 3.0 GHz processors and is running the Linux operating system. The following compile options are used for different languages:

- C: –O2 –s –static –std=c99 –lm
- C++: –O2 –s –static –lm
- Pascal: –O1 –XS

1. TRI

Little Mirko wrote into his math notebook an equation containing three positive integers, the equals sign and one of the four basic arithmetic operations (addition, subtraction, multiplication and division).

During another class, his friend Slavko erased the equals sign and the operations from Mirko's notebook.

Help Mirko by reconstructing the equation from the three integers.

Input

The first line of input contains three integers less than 100, separated by spaces.

Note: The input data will guarantee that a solution, although not necessarily unique, will always exist.

Output

On a single line, output a valid equation containing the three integers (in the same order), an equals sign and one of the four operations. If there are multiple solutions, output any of them.

Sample test data

input

```
5 3 8
```

output

```
5+3=8
```

input

```
5 15 3
```

output

```
5=15/3
```

2. PASCAL

Little Frane is already in tenth grade, but is still struggling with Pascal in computer class. For homework, his teacher wrote the following program into his notebook, and he needs to determine the output, given the integer N.

```
readln(N);
counter := 0;
for i := N-1 downto 1 do begin
     counter := counter + 1;
     if N mod i = 0 then break;
end;
writeln(counter);
```

Write a program which solves Frane's problem.

Input

The first line of input contains the integer N ($1 \le N \le 10^9$).

Output

Output the result on a single line.

Sample test data

input	**input**	**input**
1	10	27
output	**output**	**output**
0	5	18

3. JABUKE

Ante bought a piece of land. The land contains N apple trees, but his piece is triangular and it is not easy for him to determine which apple trees belong to him.

Your program will be given the coordinates of the vertices of the triangle forming Ante's piece, and the coordinates of all apple trees. Determine the area of land belonging to Ante, and the number of trees belonging to him. We consider apple trees on the very border of his piece to belong to him.

The area of a triangle with vertices (x_A, y_A), (x_B, y_B) and (x_C, y_C) is given by the following formula:

$$\frac{\left|x_A(y_B - y_C) + x_B(y_C - y_A) + x_C(y_A - y_B)\right|}{2}$$

Input

The first three lines contain the coordinates of the vertices of the triangle.

The following line contains the integer N (1 ≤ N ≤ 100), the number of apple trees.

Each of the following N lines contains the coordinates of one apple tree.

All coordinates are pairs of positive integers less than 1000, separated by a space.

Output

Output the area of land belonging to Ante on the first line, with one digit after the decimal point.

Output the number of tree belonging to Ante on the second line.

Sample test data

input

```
1 1
5 1
3 3
4
3 1
3 2
3 3
3 4
```

output

```
4.0
3
```

input

```
3 2
5 4
1 6
3
2 4
3 5
4 3
```

output

```
6.0
3
```

input

```
2 6
5 1
7 8
5
1 4
3 5
6 4
6 5
4 7
```

output

```
15.5
2
```

4. AVOGADRO

Luka is slacking again during chemistry class, while the teacher is explaining Avogadro's law.

Luka first drew a table consisting of 3 rows and N columns. Then he wrote the numbers 1 to N into the first row in arbitrary order, each number appearing exactly once. In the other two rows he also wrote integers between 1 and N, but didn't care how many times a number appeared.

Luka can now delete any set of columns from the table. After doing so, he sorts the numbers in each row in ascending order.

He wants to obtain a table in which all three rows are identical after sorting. Write a program that determines the smallest number of columns he must delete.

Input

The first line of input contains the integer N ($1 \leq N \leq 100000$), the number of columns in the table.

The following three lines contain N integers each, separated by single spaces. The numbers will be between 1 and N, and there will be no duplicates in the first row.

Output

Output the smallest number of columns Luka must delete.

Scoring

In test cases worth 40% of points, N will be less than 100.

In test cases worth 70% of points, N will be less than 10000.

Sample test data

input

```
7
5 4 3 2 1 6 7
5 5 1 1 3 4 7
3 7 1 4 5 6 2
```

output

```
4
```

input

```
9
1 3 5 9 8 6 2 4 7
2 1 5 6 4 9 3 4 7
3 5 1 9 8 6 2 8 7
```

output

```
2
```

In the first example, Luka needs to delete the second, fourth, sixth and seventh columns. After deleting the columns and sorting each row, all three rows contain the numbers 1, 3 and 5.

5. BARICA

Barica is an unusual frog. She lives in a pond where N plants float on the surface of the water. The plants are numbered 1 through N. When viewing from above, the location of each plant is given by a pair of coordinates. What makes Barica unusual is her fear of jumping diagonally and in the negative direction. More precisely, she can jump from a plant at coordinates (x_1, y_1) to another at coordinates (x_2, y_2) only if:

- $x_2 > x_1$ and $y_2 = y_1$, or
- $y_2 > y_1$ and $x_2 = x_1$

For each plant, we know the number of flies in its immediate vicinity. Barica can use her swift tongue to eat all flies near the plant she is at.

Barica absorbs one energy unit for each fly she eats, and uses K energy units for each jump she makes. Barica can not make a jump if she doesn't have enough energy units beforehand.

Barica wants to go from plant 1 to plant N and have the **largest amount of energy** possible after arriving. Barica initially has no energy and must gather energy for her first jump from the flies around plant 1.

Find the sequence of plants Barica should travel to achieve her goal.

Input

The first line of input contains two integers N and K ($2 \le N \le 300\,000$, $1 \le K \le 1000$) separated by a space.

Each of the following N lines contains three integers X, Y and F ($0 \le X, Y \le 100\,000$, $0 \le F \le 1000$) separated by spaces, meaning that there is a plant at coordinates (X, Y) with F flies around it.

The first plant in the input is plant 1, the second is plant 2 etc.

No two plants will share the same pair of coordinates.

Note: The input data will guarantee that a sequence of jumps, although not necessarily unique, will always exist.

Output

Output the final energy level on the first line.

Output an integer L, the number of plants Barica should travel, including plants 1 and N.

On the following L lines, output the sequence of plants Barica should travel.

Sample test data

```
input

6 5
1 1 5
2 1 5
1 2 4
2 3 5
3 2 30
3 3 5

output

5
4
1 1
2 1
2 3
3 3
```

```
input

8 10
1 1 15
2 2 30
1 2 8
2 1 7
3 2 8
2 3 7
4 2 100
3 3 15

output

36
5
1 1
1 2
2 2
3 2
3 3
```

```
input

9 5
5 5 10
6 5 2
7 5 1
5 6 2
6 6 6
7 6 2
5 7 1
6 7 2
7 7 1

output

2
3
5 5
7 5
7 7
```

The longest common prefix of two words is the longest word that both words start with. For example, the longest common prefix of the words "identity" and "idealistic" is the word "ide".

A database contains N words.

The algorithm to search for a query word W in the database is primitive. It compares the word W one by one with each word in the database. Two words are compared letter by letter until a letter in which they differ is found or until the end of one of the words is reached (it is then established either that the words are equal or that one is longer than the other). When the algorithm finds the word W in the database, it terminates.

Analysing the algorithm shows that the number of steps needed to find a word W is equal to the number of words W is compared to, plus the sum of the lengths of the longest common prefixes of W and each of the words it was compared to.

Write a program that calculates the number of steps the algorithm uses to find each of the Q query words.

Input

The first line contains an integer N ($1 \le N \le 30000$), the number of words in the database.

Each of the following N lines contains a single word from the database. The words are given in the order the algorithm compares them to a query word. All words in the database will be distinct.

The following line contains an integer Q ($1 \le Q \le 30000$), the number of words searched for.

Each of the following Q lines contains a single query word.

All words in the input will be strings of less than 30 lowercase letters of the English alphabet.

Output

Output one integer per line for each query word, the number of steps the algorithm uses when searching for the word.

Sample test data

input

```
5
hobotnica
robot
hobi
hobit
robi
4
robi
hobi
hobit
rakija
```

output

```
12
10
16
7
```

input

```
8
majmunica
majmun
majka
malina
malinska
malo
maleni
malesnica
3
krampus
malnar
majmun
```

output

```
8
29
14
```

In the second example, the number of steps to search for the word "krampus" is 8 because the algorithm only needs to compare its first letter to each word in the database.

When searching for the word "malnar", we need three steps for each of the first three words, and four steps for each of the remaining five words, for a total of 29 steps.

To find the word "majmun" we use a total of 14 steps. For the first word in the database, we have six successful comparisons and one step in which we determine that the word "majmunica" is longer than the query word. For the second word, we also have six successful comparisons and a final step in which it is established that the words are equal. After finding the word, the algorithm terminates with great joy.