COCI 2018/2019

Round #5, February 9th, 2018

Tasks

Task	Time limit	Memory limit	Score
Titlovi	1 s	64 MB	50
Jarvis	1 s	64 MB	70
Ispit	2 s	64 MB	90
Parametriziran	3 s	64 MB	110
Transport	1 s	64 MB	130
Total			450

In order to understand speech in foreign films, the translation of the spoken text is displayed at the bottom of the screen, colloquially called subtitles. In the digital world, subtitles are often written in .SRT files that consist of blocks of the following format:

```
[Ordinal number of the block]
[Beginning time of the subtitles --> Ending time of the subtitles]
[Subtitle text in one or more lines]
```

These blocks are separated by one blank line. The time is in the form of *HH:MM:SS,TTT* (note the colon and the comma), meaning *hour:minute:second,millisecond*. One second contains 1000 milliseconds. Here's an example of a .SRT file:

```
1
00:00:01,600 --> 00:00:04,200
Good day!

2
00:00:05,900 --> 00:00:07,999
Good day to you too!
Here you go!

3
00:00:10,000 --> 00:00:14,000
May I please have ten garlic sausages?
```

Sometimes subtitles are not fully synchronized with the associated movie, and appear for example two seconds too soon or too late. Because of that we want all the times in the appropriate .SRT file, or in some part of that file, to shift forward or backward for a specified number of (milli)seconds. Write a program which does that.

INPUT

Input data contains up to 30 lines that represent part of some .SRT file.
The first line begins with a block with the ordinal number X, followed by blocks with ordinal numbers $X + 1$, $X + 2$, ..., $X + K$, in the form described in the task statement.
Blocks are separated by a single blank line. The subtitles text within the block contains one or more non-empty lines containing the letters of the english alphabet and the punctuation marks `',.?!`.
The line after the last block contains only the '#' sign that indicates the end of the observed part of the file and that character will not appear elsewhere.
The following (last) line contains the time in milliseconds ($-10\ 000 \le T \le 10\ 000$), which needs to be added to all subtitle blocks.

OUTPUT

Print part of the .SRT file given in the input data, whereby the time stamps (beginning and ending time of each block) should be increased by T milliseconds. Test samples will ensure that the required time stamps are not negative.

SAMPLE TESTS

input

```
8
00:00:01,600 --> 00:00:04,200
We thought you was...

9
00:00:05,900 --> 00:00:07,999
a toad.
#
300
```

output

```
8
00:00:01,900 --> 00:00:04,500
We thought you was...

9
00:00:06,200 --> 00:00:08,299
a toad.
#
```

input

```
624
00:43:30,566 --> 00:43:32,108
Howdy do, ladies?

625
00:43:32,276 --> 00:43:33,943
Name of Pete.

626
00:43:47,124 --> 00:43:48,082
Ain't you gonna
introduce us, Pete?
#
-500
```

output

```
624
00:43:30,066 --> 00:43:31,608
Howdy do, ladies?

625
00:43:31,776 --> 00:43:33,443
Name of Pete.

626
00:43:46,624 --> 00:43:47,582
Ain't you gonna
introduce us, Pete?
#
```

Ivan sent N drone warriors to the final battle against Tony Stark, also known as Iron Man. Each drone has a defined frequency, expressed as an integer number, on which it receives commands from Ivan during the fight. Jarvis, the artificial intelligence developed by Toni, has to determine which frequencies those are and thereby take control over as many drones as possible.
Jarvis knows the original factory values of the frequency for each drone, but the frequencies required for each drone, unfortunately, have been changed in the meantime.
Jarvis has only one attempt. He can choose an integer number X and increase each of the factory frequencies by X (X may be negative as well). After that, Jarvis will take over control of each drone whose modified factory frequencies and the one required by the specific drone is equal.
Write a program that will determine how much drone warriors Jarvis can take control over.

INPUT

The first line contains the integer number N ($1 \le N \le 100\ 000$), the number of drones from the task statement.
In the second line there are N integers A_i ($-1\ 000\ 000 \le A_i \le 1\ 000\ 000$) representing the factory frequency values of the drone warrior.
In the third line there are N integers B_i ($-1\ 000\ 000 \le B_i \le 1\ 000\ 000$) representing the required frequency values of the drone warriors.

OUTPUT

In the only line, print out the largest number of drone warriors Jarvis can take control over.

SCORING

In the test samples totally worth 40% of the points all absolute values of the frequencies will be less than or equal to 10.

SAMPLE TESTS

input	input	input
1 1 2	2 0 0 1 1	2 1 2 5 5
output	**output**	**output**
1	2	1

Explanation of the third example:
If we choose X = 3, the factory frequencies will be 4 and 5, respectively (1 + 3 and 2 + 3), then Jarvis would take control only over the second drone warrior. If we choose X = 4, the factory frequencies will be 5 and 6 and then Jarvis would only take control over the first drone warrior. There is also no X such that Jarvis simultaneously takes control over both drone warriors.

After 26 years of studying, little Mirko took his potentially last exam. He confidently took his seat, sharpened his pencil and waited calmly for the professor's permission to start writing – after all, that was his favorite subject, *Data Structures and Algorithms*. But, as in any good story, this one also has *that but*... Namely, when he got his exam, Mirko could not even comprehend what was written in it. He only saw a meaningless matrix of letters with *N* rows and *N* columns.

Since the professor forbid leaving the classroom during the exam, Mirko decided to spend 2 hours coming up with his own task. Mirko was wondering if it is possible to select *K* consecutive columns of the matrix so that, after arbitrarily shuffling letters in the *K* selected columns' rows, there are two equal rows of the matrix. Shuffling is allowed only inside of the same row within selected columns and it is possible that a row remains unchanged after such operation.

Can you solve Mirko's task?

INPUT

In the first line of the input there are two integer numbers *N* and *K* ($2 \le K \le N \le 500$).
The following *N* rows contain *N* lowercase letters of the english alphabet describing the matrix of the letters Mirko saw in the exam.

OUTPUT

Print “DA” (Croatian for yes, without the quotation marks) if it is possible to select the *K* consecutive columns that meet the conditions of the task. Otherwise print “NE” (Croatian for no, also without quotation marks).

SCORING

In the test samples totally worth 30% of the points it will hold $N \le 10$.
In the test samples totally worth additional 40% of the points it will hold $N \le 200$.

SAMPLE TESTS

input
```
4 2
abcd
acbd
enaa
moze
```
output
```
DA
```

input
```
2 2
aa
aa
```
output
```
DA
```

input
```
3 2
nec
uuc
iti
```
output
```
NE
```

Explanation of the first test sample:

E.g. we can choose columns 2 and 3 and change the matrix so that it looks like this (we can choose not to change the first row and swap 2nd and 3rd letters in other rows):

```
abcd
abcd
eana
mzoe
```

It is clear that the first and the second row are the same, thus satisfying the task condition.

A string of characters consisting of lowercase letters of the English alphabet and question marks is called a parameterized word (e.g., *a??cd*, *bcd*, *??*). Two parameterized words are considered similar if the question mark symbols in both words can be replaced by arbitrary lowercase letters of the english alphabet so that the resulting strings are the same. For example, parameterized words *a???* and *?b?a* are similar because by replacing the question marks in both words it is possible to obtain the word *abba*.

Mirko has recently bought a collection of parametrized words. Among the *N* words found in the collection, Mirko is interested in how many pairs of similar parameterized words exist. All the words in the collection have the same number of characters, *M*, and it is possible that a word occurs multiple times in the collection.

INPUT

The first line contains the integer numbers N $(1 \le N \le 50\,000)$ and M $(1 \le M \le 6)$.
Each of the following N lines contains one parameterized word from the collection with exactly M characters.

OUTPUT

Print the total number of similar pairs of parameterized words.

SCORING

In the test samples totally worth 30% of the points it will be valid $M \le 2$.
In the test samples totally worth additional 30% points it will be valid $M \le 4$.

SAMPLE TESTS

input	input	input
3 3 ??b c?? c?c	4 6 ab??c? ??kll? a?k??c ?bcd??	5 2 ?? b? c? ?g cg
output	**output**	**output**
2	3	8

Explanation of the first test sample:
Pairs of similar words are: (??b, c??) and (c??, c?c).

The traffic network in one country consists of N cities (marked by numbers from 1 to N) and N-1 roads connecting them in a way that all cities are connected. For each road, its length in kilometers is known, and in each city there is a gas station with a certain amount of fuel.

Due to the fuel shortages that hit the country a few years ago, the leading transport agency has decided to conduct a survey on the ability to transport goods between cities. Trucks transporting goods consume one unit of fuel per kilometer and the journey between two neighboring cities is considered possible if the amount of fuel in a truck's fuel tank at the time of leaving the city is greater than or equal to the distance between the cities. Each time when a truck is in a city, the truck's fuel tank can be filled up by an amount not greater than the amount of fuel in that city's gas station. The final assessment of the survey is defined as the number of ordered pairs of cities (*A, B*) such that it is possible to travel from city *A* to city *B* under the assumption that a truck starts the journey with an empty fuel tank (at the beginning of the journey the fuel tank should be filled at the gas station in city A).

For the simplicity of the research, the following assumptions have been taken into account:

- A truck's fuel tank has unlimited capacity.
- A truck leaving from the city *A* travels directly to the city *B*, i.e. it does not visit any city on its journey more than once.

INPUT

The first line contains the integer number N $(1 \le N \le 100\,000)$, the number of cities.
In the second line there are N integer numbers A_i $(1 \le A_i \le 10^9)$, the amount of fuel at the gas station in the i^{th} city.
In the following N-1 rows there are three integer numbers U, V $(1 \le U, V \le N, U \ne V)$ and W $(1 \le W \le 10^9)$ describing the road between cities U and V of length W kilometers.

OUTPUT

Print the final assessment of the survey.

SCORING

In the test samples totally worth 20% of the points it will hold $N \le 5000$.
In the test samples totally worth 40% points the network of cities will form a chain, i.e. each city x $(1 \le x < N)$ will be connected to city $x + 1$.

SAMPLE TESTS

input

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

output

```
1
```

input

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

output

```
5
```

input

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

output

```
29
```

Explanation of the first example:
The only possible way to travel is from city 1 to city 2. The journey from city 2 to city 1 is not possible because before departure from city 2 a truck cannot have more than 1 unit of fuel in the fuel tank.

Explanation of the second example:
Pairs of cities among which the journey is possible (1, 2), (3, 2), (4, 5), (5, 1) and (5, 4).