icpc international collegiate programming contest

ICPC North America Regionals 2019

ICPC Rocky Mountain Regional Contest

Official Problem Set

icpc international collegiate programming contest

icpc.foundation

icpc global programming tools sponsor

icpc north america sponsor

ICPC ROCKY MOUNTAIN REGIONALS

OCTOBER 26, 2019

Contest Problems

A: Piece of Cake!
B: Fantasy Draft
C: Folding a Cube
D: Integer Division
E: Hogwarts
F: Molecules
G: Typo
H: The Biggest Triangle
I: Tired Terry
J: Watch Later
K: Lost Lineup

This contest contains 11 problems over 26 pages. Good luck.

For problems that state "*Your answer should have an absolute error of less than* 10^{-9}", your answer, x, will be compared to the correct answer, y. If $|x - y| < 10^{-9}$, then your answer will be considered correct.

Definition 1

For problems that ask for a result modulo m:
If the correct answer to the problem is the integer b, then you should display the unique value a such that:

- $0 \leq a < m$

  and

- $(a - b)$ is a multiple of m.

Definition 2

A string $s_1 s_2 \cdots s_n$ is lexicographically smaller than $t_1 t_2 \cdots t_\ell$ if

- there exists $k \leq \min(n, \ell)$ such that $s_i = t_i$ for all $1 \leq i < k$ and $s_k < t_k$

  or

- $s_i = t_i$ for all $1 \leq i \leq \min(n, \ell)$ and $n < \ell$.

Definition 3

- Uppercase letters are the uppercase English letters $(A, B, \ldots, Z)$.
- Lowercase letters are the lowercase English letters $(a, b, \ldots, z)$.

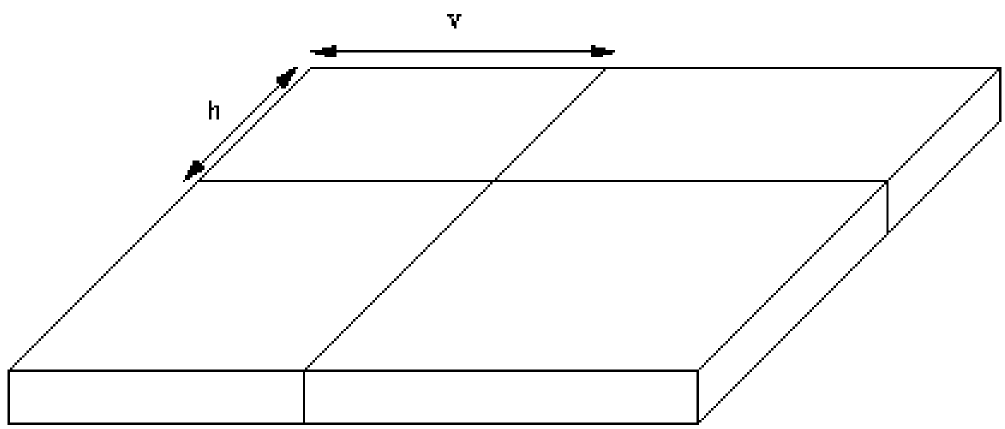
This page is intentionally left blank.

Problem A
Piece of Cake!
Time limit: 1 second

It is Greg's birthday! To celebrate, his friend Sam invites Greg and two other friends for a small party. Of course, every birthday party must have cake.

Sam ordered a square cake. She makes a single horizontal cut and a single vertical cut. In her excitement to eat cake, Sam forgot to make these cuts through the middle of the cake.

Of course, the biggest piece of cake should go to Greg since it is his birthday. Help Sam determine the volume of the biggest piece of cake that resulted from these two cuts.



Input

The input consists of a single line containing three integers n ($2 \leq n \leq 10\,000$), the length of the sides of the square cake in centimeters, h ($0 < h < n$), the distance of the horizontal cut from the top edge of the cake in centimeters, and v ($0 < v < n$), the distance of the vertical cut from the left edge of the cake in centimeters. This is illustrated in the figure above.

Each cake is 4 centimeters thick.

Output

Display the volume (in cubic centimeters) of the largest of the four pieces of cake after the horizontal and vertical cuts are made.

Sample Input 1	Sample Output 1
`10 4 7`	`168`

Sample Input 2

```
5 2 2
```

Sample Output 2

```
36
```

Sample Input 3

```
4 2 1
```

Sample Output 3

```
24
```

Problem B
Fantasy Draft
Time limit: 3 seconds

In fantasy hockey, there are n team owners that each selects k hockey players. To determine which owner gets which players, the owners hold a draft.

The draft proceeds as follows: the first owner may select any player, then the second owner can select any player except the player taken first and so on. In general, the current owner may take any player that has not been taken previously. Once all owners have selected one player, they repeat this process until all owners have selected k players. No player may be selected by multiple teams.

Initially, all players are given a ranking based on how well they played in the previous year. However, the owners may not agree with this order. For example, the first owner may believe that the player which was ranked third in the previous year is the best player and would prefer to take them.

Each owner has a preference list. On their turn, the owner selects the player that is the highest available player on their own preference list. If all players on their preference list are taken, then they resort to using the ordering from the previous year.

Given the preference list of each owner and the rankings from the previous year, which players did each owner get?

Input

The first line of the input contains two integers n ($1 \leq n \leq 60$), the number of owners, and k ($1 \leq k \leq 1\,000$), the size of each team.

The next n lines contain owners' preferences in the order of drafting. Each line starts with an integer q_i ($0 \leq q_i \leq 1\,500$), the size of the i^{th} owners' preference list. q_i names follow, separated by spaces, in order of i^{th} owner's preference. No name appears more than once in the i^{th} owners' list.

The next line contains a single integer p ($n \cdot k \leq p \leq 65\,000$), indicating the number of players in the draft.

The next p lines each contain a single name, they are ordered by their previous year's ranking. Each player name is unique and comprised of at most 12 letters of English alphabet.

The names in owners' preference lists are guaranteed to appear in the player list.

Output

Display n lines. The i^{th} of which contains the k names of the players that were selected by the i^{th} owner. The n teams should be in the original order of owners and players should be listed in the order in which they were drafted following the rules above.

Sample Input 1

```
2 2
0
0
6
Shoresy
Jonesy
Reilly
Sholtzy
Fisky
Yorkie
```

Sample Output 1

```
Shoresy Reilly
Jonesy Sholtzy
```

Sample Input 2

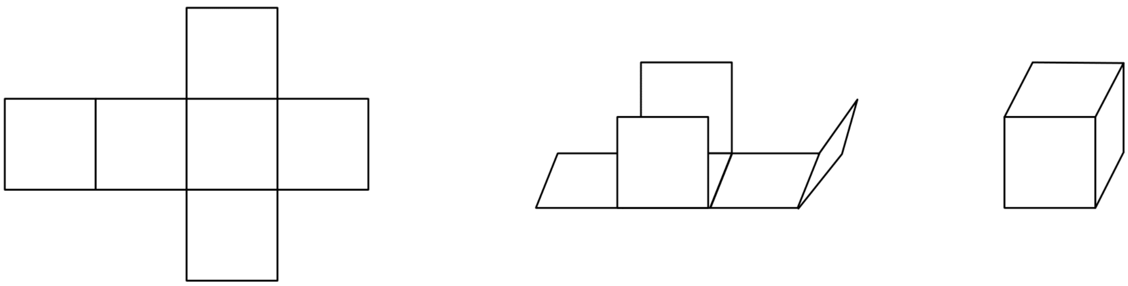
```
2 2
2 Reilly Shoresy
2 Shoresy Reilly
6
Shoresy
Jonesy
Reilly
Sholtzy
Fisky
Yorkie
```

Sample Output 2

```
Reilly Jonesy
Shoresy Sholtzy
```

Problem C
Folding a Cube
Time limit: 1 second

It is well known that a set of six unit squares that are attached together in a "cross" can be folded into a cube.



But what about other initial shapes? That is, given six unit squares that are attached together along some of their sides, can we form a unit cube by folding this arrangement?

Input

Input consists of 6 lines each containing 6 characters, describing the initial arrangement of unit squares. Each character is either a `.`, meaning it is empty, or a `#` meaning it is a unit square.

There are precisely 6 occurrences of `#` indicating the unit squares. These form a connected component, meaning it is possible to reach any `#` from any other `#` without touching a `.` by making only horizontal and vertical movements. Furthermore, there is no 2×2 subsquare consisting of only `#`. That is, the pattern

```
##
##
```

does not appear in the input.

Output

If you can fold the unit squares into a cube, display `can fold`. Otherwise display `cannot fold`.

Sample Input 1	Sample Output 1

`......`
`......`
`######`
`......`
`......`
`......`
	`cannot fold`

Sample Input 2	Sample Output 2

``......``
``#.....``
``####..``
``#.....``
``......``
``......``
	``can fold``

Sample Input 3	Sample Output 3

``..##..``
``...#..``
``..##..``
``...#..``
``......``
``......``
	``cannot fold``

Sample Input 4	Sample Output 4

``......``
``...#..``
``...#..``
``..###.``
``..#...``
``......``
	``can fold``

Problem D
Integer Division
Time limit: 2 seconds

In `C++` division with positive integers always rounds down. Because of this, sometimes when two integers are divided by the same divisor they become equal even though they were originally not equal. For example in `C++`, `5/4` and `7/4` are both equal to `1`, but $5 \neq 7$.

Given a list of nonnegative integers and a divisor, how many pairs of distinct entries in the list are there that give the same result when both are divided by the divisor in `C++`?

Input

The first line of input contains two integers n ($1 \leq n \leq 200\,000$), the number of elements in the list, and d ($1 \leq d \leq 10^9$), the divisor.

The second line of input contains n integers $a_1, \ldots, a_n$ ($0 \leq a_i \leq 10^9$), where a_i is the i^{th} element of the list.

Output

Display a single integer indicating the number of distinct pairs of indices (i, j) with $1 \leq i < j \leq n$ such that $a_i/d = a_j/d$ when using integer division in `C++`. Note that the numbers in the list are not necessarily distinct (i.e. it is possible that $a_i = a_j$ for some indices $i \neq j$).

Sample Input 1	Sample Output 1
5 4 4 5 6 7 8	6

Sample Input 2	Sample Output 2
5 1 4 5 6 7 8	0

Sample Input 3	Sample Output 3
6 1 1 2 1 2 1 2	6

This page is intentionally left blank.

Problem E
Hogwarts
Time limit: 1 second

The Hogwarts School of Witchcraft and Wizardry is the home of many students during the school year. The school has many rooms connected by corridors and stairs. Each room has four exits labelled by the integers $1, 2, 3$ or 4. Some exits lead to another room, some of them are blocked, and some even lead back to the same room you just came from.

New students often have difficulty finding their way, especially since the corridors and stairs are regularly moving, disconnecting and reconnecting different rooms! Luckily, these reconfigurations only take place when no one is walking in the school. All you want to know is how to get from the entrance to the dormitory. A senior student has given you instructions as a sequence of numbers among $1, 2, 3, 4$. The first number in the sequence is the exit to take from the starting room. The second number is the exit to take from the second room in the path, and so on. If at any point the indicated exit is blocked, you go back to the entrance and give up. To be successful you must arrive at the dormitory at the end of the entire sequence. Even if it appears you have reached the dormitory before the entire sequence is followed, you are not sure if that is an illusion. Therefore you follow the entire sequence.

You carefully followed the instructions and arrived at the dormitory. However, the way the rooms are connected to each other has changed after the senior student gave you the instructions, and you just happen to arrive at the same destination even if the rooms you encountered along the way may be completely different.

You wonder if you are just lucky, or if the reconfiguration of the corridors and stairs ensures that the instructions still lead you to the same destination. Isn't that magical?

You will be given a configuration of the school when the senior student walked from the entrance to the dormitory, as well as the configuration of the school when you start following the given instructions. You want to know if every possible sequence of instructions that led the senior student to the dormitory will also lead you to the dormitory in the configuration you walk through. Both the senior student and you start walking from the entrance of the school.

Input

The first line of input contains a single integer n ($2 \leq n \leq 1\,000$), indicating the number of rooms in the school. The rooms are numbered 1 to n, where room 1 is the entrance and room n is the dormitory.

The next n lines of input describe the configuration of the school when the senior student walked to the dormitory, followed by another n lines describing the configuration of the school when you start to walk to the dormitory.

The i^{th} line in the school configuration consists of four non-negative integers, indicating which room exits $1, 2, 3$ and 4 lead to. If the room number is 0, the corresponding exit is blocked.

Output

If it is not possible for the senior student to walk from the entrance to the dormitory, display `Impossible`.

If it is possible, display `Yes` if you can get from the entrance to the dormitory by following any sequence of instructions taking the senior student from the entrance to the dormitory. Otherwise, display `No`.

Sample Input 1

```
4
1 1 1 2
2 2 2 3
3 3 3 4
0 0 0 0
2 2 2 2
3 3 3 3
4 4 4 4
4 4 4 4
```

Sample Output 1

```
Yes
```

Sample Input 2

```
4
1 1 1 2
2 2 2 3
3 3 3 4
0 0 0 0
2 2 2 2
3 3 3 3
4 4 4 4
0 0 0 0
```

Sample Output 2

```
No
```

Problem F
Molecules
Time limit: 2 seconds

A molecule consists of atoms that are held together by chemical bonds. Each bond links two atoms together. Each atom may be linked to multiple other atoms, each with a separate chemical bond. All atoms in a molecule are connected to each other via chemical bonds, directly or indirectly.

The chemical properties of a molecule is determined by not only how pairs of atoms are connected by chemical bonds, but also the physical locations of the atoms within the molecule. Chemical bonds can pull atoms toward each other, so it is sometimes difficult to determine the location of the atoms given the complex interactions of all the chemical bonds in a molecule.

You are given the description of a molecule. Each chemical bond connects two distinct atoms, and there is at most one bond between each pair of atoms. The coordinates of some of the atoms are known and fixed, and the remaining atoms naturally move to the locations such that each atom is at the average of the locations of the connected neighboring atoms via chemical bonds. For simplicity, the atoms in the molecule are on the Cartesian xy-plane.

Input

The first line of input consists of two integers n ($2 \leq n \leq 100$), the number of atoms, and m ($n-1 \leq m \leq \frac{n(n-1)}{2}$), the number of chemical bonds.

The next n lines describe the location of the atoms. The i^{th} of which contains two integers x, y ($0 \leq x, y \leq 1\,000$ or $x = y = -1$), which are the x and y coordinates of the i^{th} atom. If both coordinates are -1, however, the location of this atom is not known.

The next m lines describe the chemical bonds. The i^{th} of which contains two integers a and b ($1 \leq a < b \leq n$) indicating that there is a chemical bond between atom a and atom b.

It is guaranteed that at least one atom has its location fixed.

Output

Display n lines that describe the final location of each atom. Specifically, on the i^{th} such line, display two numbers x and y, the final coordinates of the i^{th} atom. If there are multiple solutions, any of them is accepted. A solution is accepted if the coordinates of each unknown atom and the average coordinates of all its neighboring atoms via chemical bonds differ by at most 10^{-3}. Note that it is acceptable for multiple atoms to share the same coordinates.

Sample Input 1	Sample Output 1
3 2 0 0 -1 -1 2 0 1 2 2 3	0 0 1 0 2 0

Sample Input 2	Sample Output 2
5 4 0 0 -1 -1 -1 -1 -1 -1 4 0 1 2 2 3 3 4 4 5	0 0 1 0 2 0 3 0 4 0

Sample Input 3	Sample Output 3
4 3 0 0 2 0 1 1 -1 -1 1 4 2 4 3 4	0 0 2 0 1 1 1 0.3333333

Problem G
Typo
Time limit: 6 seconds

It is now far into the future and human civilization is ancient history. Archaeologists from a distant planet have recently discovered Earth. Among many other things, they want to decipher the English language.

They have collected many printed documents to form a dictionary, but are aware that sometimes words are not spelled correctly (typos are a universal problem). They want to classify each word in the dictionary as either correct or a typo. Naïvely, they do this using a simple rule: a typo is any word in the dictionary such that deleting a single character from that word produces another word in the dictionary.

Help these alien archaeologists out! Given a dictionary of words, determine which words are typos. That is, which words result in another word in the dictionary after deleting a single character.

For example if our dictionary is {`hoose`, `hose`, `nose`, `noises`}. Then `hoose` is a typo because we can obtain `hose` by deleting a single 'o' from `hoose`. But `noises` is not a typo because deleting any single character does not result in another word in the dictionary.

However, if our dictionary is {`hoose`, `hose`, `nose`, `noises`, `noise`} then the typos are `hoose`, `noises`, and `noise`.

Input

The first line of input contains a single integer n, indicating the number of words in the dictionary.

The next n lines describe the dictionary. The i^{th} of which contains the i^{th} word in the dictionary. Each word consists only of lowercase English letters. All words are unique.

The total length of all strings is at most $1\,000\,000$.

Output

Display the words that are typos in the dictionary. These should be output in the same order they appear in the input. If there are no typos, simply display the phrase `NO TYPOS`.

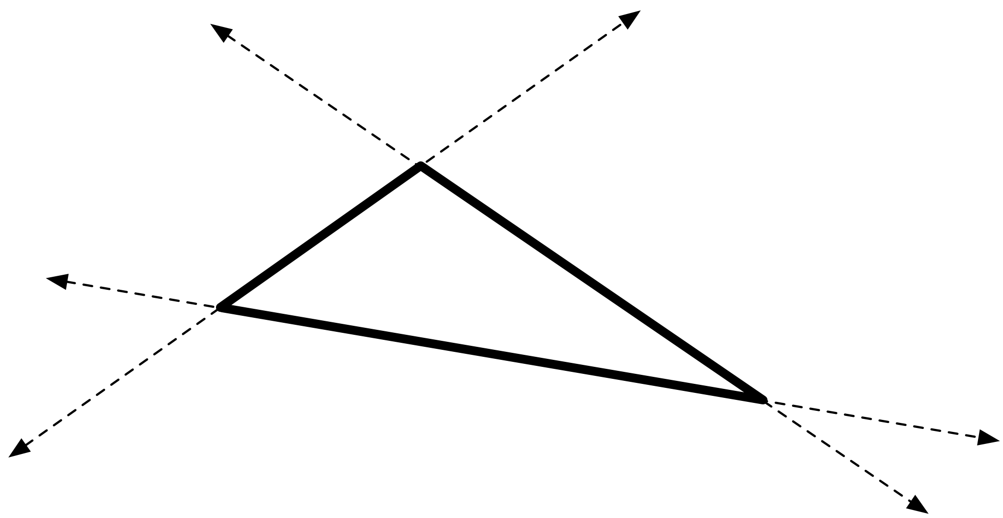
Sample Input 1	Sample Output 1
5 hoose hose nose noises noise	hoose noises noise

Sample Input 2	Sample Output 2
4 hose hoose oose moose	hoose moose

Sample Input 3	Sample Output 3
5 banana bananana bannanaa orange orangers	NO TYPOS

Problem H
The Biggest Triangle
Time limit: 3 seconds

Three infinite lines define a triangle, unless they meet at a common point or some of them are parallel.



Given a collection of infinite lines, what is the largest possible perimeter of a triangle defined by some three lines in the collection?

Input

The first line of input contains a single integer n ($3 \le n \le 100$) indicating the number of infinite lines.

The next n lines describe the collection of infinite lines. The i^{th} such line contains four integers x_1, y_1, x_2, y_2 ($-10\,000 \le x_1, y_1, x_2, y_2 \le 10\,000$) where $(x_1, y_1) \ne (x_2, y_2)$ are two points lying on the i^{th} infinite line.

Output

Display a single real value which is the perimeter of the largest triangle that can be formed from three of the infinite lines. Your output will be considered correct if it is within an absolute or relative error of 10^{-5} of the correct answer.

If no triangle can be formed using the given lines, then you should instead display the message `NO TRIANGLE`.

Sample Input 1

```
3
0 0 0 1
0 0 1 0
0 1 1 0
```

Sample Output 1

```
3.4142135624
```

Sample Input 2

```
3
0 0 0 1
0 0 1 0
0 0 1 1
```

Sample Output 2

```
no triangle
```

Sample Input 3

```
4
0 0 0 1
0 4 3 0
0 0 1 0
-1 -1 1 1
```

Sample Output 3

```
12.0000000000
```

Problem I
Tired Terry
Time limit: 1 second

Terry is feeling tired and he suspects it is because of a lack of sleep. He created a device that records his sleeping pattern over a period of time measured in seconds.

Assuming that the recorded sleeping pattern keeps repeating, help Terry by letting him know how often he is tired during each of the repeating time periods.

More precisely, for integers p and d, we say that Terry is tired at second i if from second $i - p + 1$ to second i (inclusive) he has slept for less than d seconds.

Input

The first line of input contains three integers n ($1 \leq n \leq 86\,400$), the length of Terry's sleep pattern, p ($1 \leq p \leq N$), and D ($1 \leq d \leq p$) as described above.

The second line of input contains a single string of length n which describes the period of time that is recorded. The i^{th} such character is a `W` if Terry is awake at the i^{th} second, or is a `Z` if Terry is asleep at the i^{th} second.

Output

Display a single integer which represents the number of seconds that Terry is tired during each of the repeating time periods.

Sample Input 1	Sample Output 1
2 1 1 WZ	1

Sample Input 2	Sample Output 2
5 3 2 WZWWZ	4

This page is intentionally left blank.

Problem J
Watch Later
Time limit: 6 seconds

While browsing YouTube videos, you commonly use the handy dandy tool that is *Add Video to Watch Later*. One day you finally decide that 'Later' has finally arrived and you have SO many videos in your list.

You have a list of videos with different types. For example, some might be bouldering videos, some might be cat videos, and so on. You want to watch all videos of the same type before you watch videos of a different type, but you are allowed to watch the video types in any order you wish. For example, you might want to watch all bouldering videos before watching any cat videos.

To start watching, you have to click on a video to play it. You may click on any video in your list to start watching that video. Whenever a video finishes playing, it is automatically deleted from the list. The order of the remaining videos does not change when a video finishes playing. Also, the next video in the list is automatically played if it is of the same type as the video you just watched. If it is of a different type, or if there is no video after the one you just watched, you must click on another video in the list to watch more videos (unless you have watched all videos).

Given the description of your Watch Later list, what is the minimum number of clicks needed to watch every video with the restrictions described above?

Input

The first line of the input contains two integers n ($1 \leq n \leq 400$), the number of videos in your Watch Later list, and k ($1 \leq k \leq 20$), the number of different video types in the list.

The second line of input contains a string of length n which describes the Watch Later list. The i^{th} character in the string is a lowercase English letter which describes the type of the i^{th} video. Two videos are of the same type only if they are denoted by the same letter.

Output

Output a single integer on a line, indicating the minimum number of clicks required to watch all videos currently in your Watch Later list.

Sample Input 1	Sample Output 1
4 2 abba	2

Sample Input 2	Sample Output 2
4 2 rtrt	3

This page is intentionally left blank.

Problem K
Lost Lineup
Time limit: 1 second

Jimmy and his friends were all standing in a lineup for ice cream when a huge gust blew them all around. The friends want to keep things fair and make sure everyone gets their ice cream in the order they started with. The friends do not remember the order, but each of them remember exactly how many people were between them and Jimmy. Jimmy is always the first person in line. Can you help him and his friends remember the order?

Input

The first line contains a single integer n ($1 \leq n \leq 100$), the number of people in the line.

The second line contains $n-1$ space separated integers, where d_i ($0 \leq d_i \leq n-2$) is the number of people between the $(i+1)^{\text{th}}$ person and Jimmy.

Jimmy is always first in the lineup.

Output

Print a single line with n integers, the people in the order of the original lineup. It is guaranteed that there is always a unique solution.

Sample Input 1	Sample Output 1
2 0	1 2

Sample Input 2	Sample Output 2
4 1 2 0	1 4 2 3

This page is intentionally left blank.