Problem A
Welcome Sign

Time Limit: 1 second

Amanda is opening a new bakery in town. She will put up a fancy LED sign to show a grand opening message on the storefront window. The sign has r rows and c columns of display units. Each unit can display one letter or be blank.

Amanda has written a message consisting of exactly r words. She will arrange them onto the LED sign such that each row will display one word from the message in order. Each word must be as centered as possible, which means that if the number of blanks on the left side and right side of the word are l and r, then $|l - r|$ must be as close to zero as possible. On those rows where l cannot equal r, Amanda would like to balance the number of blanks on the two sides. More specifically, for the first and every other such row (1^{st}, 3^{rd}, 5^{th}, and so forth), she would like l to be less than r. For the other rows (2^{nd}, 4^{th}, 6^{th}, and so forth), she would like l to be greater than r.

What would the sign look like after Amanda sets it up according to the above?

Input

The first line of input contains two integers r and c ($1 \leq r, c \leq 50$), the number of rows and the number of columns of display units.

The next r lines each contain a word with at least one and at most c lowercase letters (a–z), giving the words to display per row in order.

Output

Output r lines. Each line must contain exactly c characters representing the displayed characters on one row of the LED sign. Output a dot (.) for each blank display unit.

Sample Input 1	Sample Output 1
8 7 welcome we are open free coffee for icpc	welcome ..we... ..are.. ..open. .free.. .coffee ..for.. .icpc..

This page is intentionally left blank.

Problem B
Triangles of a Square
Time Limit: 1 second

Ashley has given Brandon a square of side 2024. She also has drawn a single line segment that connects two different sides of the square.

Brandon wants to draw some additional line segments such that it is possible to decompose the square into a set of disjoint triangles, where each triangle has sides that are either subsegments of the sides of the square, or subsegments of any drawn line segment.

Compute the minimum number of additional line segments Brandon needs to draw to make this possible.

Input

Imagine that the square is axis-aligned with its bottom-left corner at $(0, 0)$ and top-right corner at $(2024, 2024)$.

Input has a single line with four integers x_1, y_1, x_2, y_2 $(0 \leq x_1, y_1, x_2, y_2 \leq 2024)$ specifying the coordinates of the end points of the line segment initially drawn by Ashley. One end point is at (x_1, y_1) and the other end point is at (x_2, y_2).

It is guaranteed the two end points are distinct. Both end points are on sides of the square. If the segment intersects a side of the square, it does so at exactly one point.

Output

Output a single integer, the minimum number of additional line segments Brandon needs to draw.

Sample Input 1	Sample Output 1
0 10 10 0	2

Sample Input 2	Sample Output 2
2024 2024 0 0	0

This page is intentionally left blank.

Problem C
Traffic Lights

Time Limit: 1 second

You are staring down a street with n traffic lights. Each traffic light has been set to alternate its color between r seconds of red and g seconds of green based on the two duration parameters r and g. These two parameters may vary across different traffic lights.

You saw that all the traffic lights had been green, but had just all turned red at the same time. You are wondering how many seconds it would take before the traffic lights are all green again.

Input

The first line of input contains a single integer n ($1 \le n \le 10$), the number of traffic lights.

The next n lines each contain two integers r and g ($1 \le r, g \le 10$), giving the number of seconds that a traffic light stays red and green respectively.

Output

Output a single integer, the earliest time that the lights are all green again. If this will never happen, output -1.

Sample Input 1	Sample Output 1
3 2 4 3 1 6 1	27

This page is intentionally left blank.

Problem D
Duel of Cards
Time Limit: 1 second

Alice and Bob are playing a card game. There are $2n$ cards uniquely numbered from 1 to $2n$. The cards are shuffled and dealt to the two players so that each player gets n cards. Each player then arranges the cards they get into a deck in any order that they choose, facing down.

The game has n turns. In each turn, both players reveal the card that is on the top of their deck and compare the numbers on the two cards. The player with the larger card wins and scores one point. This is repeated until all cards in the decks are compared.

After getting her n cards, Alice wonders what is the minimum and maximum number of points she may possibly score in the game.

Input

The first line of input contains a single integer n ($1 \leq n \leq 1\,000$), the number of cards that Alice gets.

The next n lines each have a single integer between 1 and $2n$ (both inclusive) giving a card that is dealt to Alice. It is guaranteed that all those cards are unique.

Output

Output two integers, the minimum and maximum number of points Alice may score.

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

This page is intentionally left blank.

Problem E
Which One is Larger
Time Limit: 1 second

A question that often confuses a chat bot is:

> 9.9 or 9.11, which one is larger?

Some chat bot would incorrectly state that 9.11 is larger than 9.9. Though we are not exactly sure what reasoning is behind that wrong answer, one hypothesis is that the chat bot might have compared the decimal numbers as integer tuples and thought that $(9, 11) > (9, 9)$. Integer tuples (x_1, y_1) and (x_2, y_2) are compared by first comparing x_1 against x_2, and if they are equal, then comparing y_1 against y_2.

Your task is to determine given two decimal numbers whether their comparison would confuse a chat bot that may use integer tuple comparison. That is, you want to check if the comparison result would be the same under integer tuple comparison and regular decimal comparison.

Input

Input has two lines. Each line has one positive decimal numbers with 1 to 5 digits before the decimal point, and 1 to 5 digits after the decimal point. There are no leading or trailing zeros (a single zero before or after the decimal point is possible). The two decimal numbers are not equal.

You may assume that the chat bot discards any leading zeros when converting the digits after the decimal point to an integer in a tuple, e.g. 4.01 converts to $(4, 1)$. Note that it is possible that the two numbers are equal by integer tuple comparison, in which case it is considered to be a different result than the result of a regular decimal comparison.

Output

Output the decimal number that is larger if the comparison would be the same under integer tuple comparison and regular decimal comparison. Otherwise, output -1.

Sample Input 1	Sample Output 1
9.9 9.11	-1

Sample Input 2	Sample Output 2
12.34 5.6789	12.34

Sample Input 3	Sample Output 3
0.01 0.009	-1

Sample Input 4	Sample Output 4
4.1 4.01	-1

Problem F
Office Building
Time Limit: 1 second

Company Z has purchased a land lot for their new office building. The land is shaped like a rectangular grid with r rows and c columns, in which each cell has a tree. The age of each tree is known.

Because Company Z is at the innovative front of the world, their new office building will not just be rectangular. Instead, it will have some exotic shape and a very special floor plan. The shape can be represented by some connected grid cells. There is no particular facing requirement of the building, and the floor plan can be rotated by 90 degrees an arbitrary number of times. However, the floor plan cannot be flipped vertically or horizontally.

Company Z wants to choose a building location within their land lot. The trees in those cells that are occupied by the building will have to be cut down. Company Z would like to preserve the trees so that the sum of age of the remaining trees is as large as possible. Could you help them choose the best location for their new office building?

Input

The first line of input contains two integers r and c ($1 \leq r, c \leq 20$), the dimensions of the land lot.

The next r lines each contain c integers between 1 and 100 (both inclusive) describing the ages of the trees in the land lot.

The next line contains two integer s and t ($1 \leq s \leq r, 1 \leq t \leq c$), the dimensions of the floor plan.

The next s lines each contain t characters that describe the floor plan of the building. A hash (`#`) denotes a cell occupied by the building and a dot (`.`) denotes a non-occupied cell. There is at least one hash. It is guaranteed that the hashes form a connected shape. Two hashes are directly connected if their cells share a side, and a shape is connected when all its hashes are either directly or indirectly connected. There is no row or column in the floor plan that is completely empty.

Output

Output a single integer, the maximum sum of age of the trees that can be preserved.

Sample Input 1	Sample Output 1
4 5 4 3 1 1 1 5 4 3 5 1 9 6 2 1 1 8 7 1 1 2 3 4 ###. #.## #..#	58

Sample Input 2	Sample Output 2
3 3 6 6 6 3 6 1 1 1 1 2 3 #.. ###	25

Problem G
Intergalactic Team
Time Limit: 3 seconds

The Intergalactic Competitive Programming Contest (ICPC) is coming up, and it's time to choose a team that will represent our planet in this esteemed competition. The ICPC president has announced the team size for this year's competition to all planets that want to compete. The Earth ICPC committee needs to form a team that consists of exactly this number of members.

To maximize compatibility and teamwork between team members, a set of people can form a team if for any pair of members (u, v) in the set, u must specify v as someone they want to work with and vice versa. In addition, as part of the competitors' demand, if two competitors specify that they want to work with each other, then either both of them or neither of them shall be in the team.

Earth has n eligible competitors to participate in this year's competition. Earth has collected data regarding for every competitor who they want to work with as teammates. With this information available, can you help the Earth ICPC committee determine the number of ways to choose a team of the required size for the upcoming competition? Two team configurations are considered different if there is at least one member that is in one configuration but not in the other.

Input

The first line of input contains three integers n, m, and k ($1 \leq n \leq 10^5, 1 \leq m \leq min(n \cdot (n-1), 10^6), 1 \leq k \leq n$), where n is the number of prospective competitors, m is the number of entries specifying that which competitors are willing to work with which competitors, and k is the exact team size required for this year's ICPC.

The competitors are numbered 1 to n. The next m lines each contain two integers x and y ($1 \leq x, y \leq n, x \neq y$), denoting that competitor x wants to work with competitor y. It is guaranteed that all those entries are unique.

Output

Output a single integer, the number of ways for the Earth ICPC committee to choose a team for the upcoming ICPC.

Sample Input 1	Sample Output 1
7 7 2 1 2 2 3 3 1 4 5 5 4 6 7 7 6	2

Problem H
Herb Mixing
Time Limit: 1 second

Leon has collected some herbs of two possible colors: green and red. Leon's health level will be boosted after he eats those herbs. Mixing the herbs can increase their effect according to the following recipes:

- One green herb: Boosts health by 1.
- Two green herbs: Boosts health by 3.
- Three green herbs: Boosts health by 10.
- One green herb and one red herb: Boosts health by 10.

The herbs cannot be consumed in any way other than those listed above.

What is the maximum amount of Leon's health that can be boosted if he optimally mixes and eats his herbs?

Input

Input contains two integers between 0 and 100 (both inclusive), the number of green herbs and the number of red herbs that Leon has collected.

Output

Output a single integer, the maximum amount of Leon's health that can be boosted.

Sample Input 1	Sample Output 1
8 3	43

Sample Input 2	Sample Output 2
0 2	0

This page is intentionally left blank.

Problem I
Island Memories
Time Limit: 3 seconds

There is an island country that has n islands numbered 1 to n. The islands are connected by $n-1$ bidirectional drawbridges, such that it is possible travel from each island to every other island. Every drawbridge can be lifted to provide clearance for boat traffic. Sometimes a drawbridge can be lifted for a prolonged period of time, during which the country operates as two *zones* of road traffic. Each zone is a maximal set of islands connected by the un-lifted drawbridges. It is impossible to travel by road from one zone to the other zone. The country never lifts more than one drawbridge at any time in order to mitigate the inconvenience of road travel.

You are writing a tourist guide for people who would like to visit this island country. You found that the country does not yet have a map that describes how their islands are connected by drawbridges. You thus interviewed m islanders who live in this country to gain some information. Each islander told you a memory that describes which islands were in their zone sometime in the past when there was a drawbridge lifted. However, some islanders might have remembered things wrong. You would like to check if all the islanders' memories are consistent, which means that there exists a way to connect the islands by drawbridges so that all the zones described by the m islanders can actually be formed after lifting exactly one drawbridge at a time.

Input

The first line has two integers n, m ($2 \le n, m \le 1\,000$), the number of islands in the country and the number of islanders you interviewed.

This is followed by m islanders' memories. Each islander's memory starts with a single integer k ($1 \le k < n$) on the first line, the number of islands that are in this islander's zone. The next line has k integers in increasing order giving those islands in the zone. You may assume that the islanders never left out any islands in their zone when describing their memories (i.e. each set of islands in a memory is maximal as connected by the un-lifted drawbridges).

Output

Output 1 if all the islanders' memories are consistent, or 0 otherwise.

Sample Input 1	Sample Output 1
5 2 2 1 2 3 1 2 3	1

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

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

Problem J
Perfect Squares
Time Limit: 1 second

A famous theorem in number theory states that every positive integer can be written as the sum of four perfect squares. You have noticed, though, that usually fewer squares are enough. For example, 27 only requires three perfect squares: $27 = 5^2 + 1^2 + 1^2$.

You share your observations with a mathematician friend, who rattles off the following perfect squares facts:

- An odd prime p can be written as the sum of *two* squares if and only if $p \equiv 1 \pmod 4$.
- If two positive integers a and b can be written as the sum of two squares, then so can their product ab.
- Every positive integer can be written as the sum of *three* perfect squares, unless it is of the form $4^a \cdot (8b + 7)$, where a and b are some non-negative integers.

This last fact about sums of three squares intrigues you, and so you would like to write a program that verifies the claim is true by producing the actual squares.

Input

Input contains a single integer n ($1 \leq n \leq 10^{12}$).

Output

If n can be expressed as the sum of three squares, output three integers x, y, and z. Your answer will be judged correct if $0 \leq x, y, z \leq \sqrt{n}$ and $n = x^2 + y^2 + z^2$. If there are multiple valid choices for x, y, and z you may output any of them. You must output exactly three integers, even if n can be expressed as the sum of two or fewer squares.

If n cannot be expressed as the sum of three squares, output -1 and no further output.

Sample Input 1	Sample Output 1
`22`	`3 3 2`

Sample Input 2	Sample Output 2
`23`	`-1`

Sample Input 3	Sample Output 3
`999999999989`	`471545 0 881842`

Problem K
Optimized Cheating
Time Limit: 1 second

Bob's favorite game just released a limited-time item for sale that will boost his game character's power significantly. However, there is not enough time for Bob to acquire sufficient in-game currency to purchase the item. Bob thus decides to resort to a cheat tool he found online to modify his in-game currency value.

The cheat tool will take two values x and y specified by Bob and overwrites all memory slots that contain the value x with the value y. Bob does not want to use the cheat tool unsafely. He does not want to modify any memory slots other than the one that stores his currency value and cause the game to crash. Therefore, Bob needs to make sure that his currency value does not have any duplicate in the game's memory space before running the cheat tool. Bob can use a set of operations provided by the game to modify his currency value, such as doing missions, purchasing items, and so on. Those operations can add, subtract, multiply or divide his currency value by a constant. All those operations will only change Bob's currency value, and they will not affect any other memory slots. Bob cannot use an operation that would cause his currency value to become negative (e.g. buying an item that costs more than his available currency).

Bob knows that the memory space of the game can be represented as an array consisting of n integers. He also knows the location of the memory slot within this array that stores his currency value. However, Bob does not know how to modify his currency value as quickly as possible in order to use the cheat tool safely. Can you help him?

Input

The first line of input contains three integers n, m, and k ($1 \le n \le 10^4, 1 \le m \le 1\,000, 1 \le k \le n$), where n is the number of memory slots in the game's memory space, m is the number of operations that Bob can use, and k is the 1-based index of the memory slot that stores Bob's in-game currency value in the game's memory space.

The next n lines each contain a single integer between 1 and 10^9, giving the values stored in the game's memory space in order.

The next m lines each contain a single character $\texttt{p}$ ($+$, $-$, $\star$, or $/$) and an integer v ($1 \le v \le 10^9$) that describe one operation that Bob can use to change his currency value. If Bob's current currency value is u, then after applying the operation his currency value will become $u\ \texttt{p}\ v$. For instance, applying an operation $+\ 3$ will increase Bob's currency value by 3. Divisions are integer divisions (e.g., $7\ /\ 3 = 2$). An operation cannot be applied if it would result in a negative currency value. Each operation can be applied multiple times (including zero).

Output

If it is possible for Bob to make to his in-game currency value unique in the game's memory space, output a single integer t on the first line, the minimum number of operations that Bob must apply. Then output t lines that each have an integer denoting the 1-based index of the operation that Bob should apply in order. If there are multiple ways to apply the operations, you may output any of them.

If it is impossible for Bob to make his currency value unique in the game's memory space, output -1.

Sample Input 1	Sample Output 1
5 3 4 2 1 3 2 3 - 1 * 2 + 3	1 2

Problem L
Pianissimo
Time Limit: 1 second

Composers use dynamics in sheet music to indicate how loud or soft the notes shall be played. Consider an 8-scale dynamic system:

- *ppp*: pianississimo (the softest)
- *pp*: pianissimo (very soft)
- *p*: piano (soft)
- *mp*: mezzopiano (moderately soft)
- *mf*: mezzoforte (moderately loud)
- *f*: forte (loud)
- *ff*: fortissimo (very loud)
- *fff*: fortississimo (the loudest)

When a musician performs, it practically does not matter how absolutely loud or soft a note is played. The audience only hear their relative difference. Suppose we use numbers to describe the absolute power of a note, where larger numbers indicate larger absolute power. Consider a musician who plays all her notes with power up to 100. When she goes from power 40 to 70, some audience may think that she goes from *p* to *f*, while some others may think that she goes from *pp* to *mf*.

The musician's interpretation must be consistent throughout the entire piece. That is, between two notes of different dynamics, the note with a louder dynamic must always have a *strictly* larger absolute power. Between two notes of the same dynamic, their power can vary (usually slightly, but there is no strict requirement).

You just heard a musician perform a piece with n notes. You know the piece very well and you remember all the dynamics that the composer wrote. How consistent is the musician's performance according to the written dynamics? You would like to count the unordered pairs of notes that violate the dynamics. Such a pair of notes is not necessarily consecutive and could be far from each other in the sheet music.

Input

The first line of input contains two integers n and m ($1 \leq m \leq n \leq 2 \cdot 10^5$), the number of notes and the number of dynamic changes.

The next n lines each have an integer between 1 and 10^9, giving the absolute power of a note played by the musician. Larger numbers indicate larger absolute power.

The next m lines each have a 1-based index i and a dynamic d, which mean the composer wrote that starting from note i the dynamic should change to d. Each dynamic is a string from the 8 dynamics: `ppp, pp, p, mp, mf, f, ff, fff`. The dynamics have distinct indices and are given in increasing order of their index. The first dynamic always starts from note 1.

Output

Output a single integer, the number of unordered pairs of notes that violate the dynamics.

Sample 1 Explanation

The 1^{st} pair of notes that violates the dynamics is note 3 (power 20, dynamic `f`) and note 6 (power 20, dynamic `p`). The 2^{nd} pair of notes that violates the dynamics is note 4 (power 30, dynamic `ff`) and note 7 (power 35, dynamic `f`).

Sample Input 1

```
8 6
10
15
20
30
19
20
35
40
1 p
3 f
4 ff
5 p
7 f
8 ff
```

Sample Output 1

```
2
```

Sample Input 2	Sample Output 2
12 5 5 10 9 20 30 40 90 90 11 10 4 3 1 p 4 f 7 ff 9 p 11 ppp	0

This page is intentionally left blank.

Problem M
Memories of Passport Stamps
Time Limit: 1 second

You just got your new passport, fresh with pages ready to be stamped by immigration officers. Sadly, because your passport has so many pages, immigration officers are too lazy to try to use your pages efficiently, so you may need to get a new passport sooner than you think.

You have some trips prepared. For each trip, when you go through passport control, the immigration officer will look for some contiguous pages, none of which are stamped, and then stamp all of them. Because the officer is lazy, there is no guarantee which contiguous pages get stamped.

Now, your passport no longer has enough contiguous empty pages to satisfy your next trip, so you're in the process of applying for a new passport. Before you do that, you decide to scan through your passport and reminisce about all the fun trips you had. Your least favorite part of these trips was waiting for immigration officers to stamp your passport.

Leafing through your passport, you remember that you took k trips. There are n contiguous sections of stamped pages. What is the minimum value s such that it is possible for each officer to stamp somewhere between 0 and s pages (both inclusive), so that you can get exactly the sections of stamped pages that you have in your passport now? Different officers may stamp different numbers of pages, and an officer is allowed to stamp zero pages.

Input

The first line of input contains two integers n ($1 \le n \le 10^5$) and k ($n \le k \le 10^{18}$), where n is the number of contiguous sections of stamped pages, and k is the number of trips you took.

The next n lines each contain a single integer p ($1 \le p \le 10^{18}$), the number of contiguous stamped pages in a section of your passport. It is guaranteed your passport will have at most 10^{18} stamped pages in total.

Output

Output a single integer, the minimum value s such that it is possible for each officer to stamp somewhere between 0 and s pages so that you can get exactly the sections of stamped pages that you have in your passport now.

Sample Input 1	Sample Output 1
3 5 9 12 5	6