A A Prize No One Can Win

Time limit: 1.5s

After the festive opening of your new store, the Boutique store for Alternative Paramedicine and Cwakhsahlvereigh, to your disappointment you find out that you are not making as many sales as you had hoped. To remedy this, you decide to run a special offer: you will mark some subset of the n items for sale in your store as participating in the offer, and if people buy exactly two of these items, and the cost of these items is *strictly* more than X euros, you will give them a free complimentary unicorn horn!

Since you recently found out all your unicorn horns are really narwahl tusks, you decide to rig the offer by picking the participating items in such a way that no one can earn a horn anyway.

To make sure no one becomes suspicious, you want to mark as many items as possible as participating in the offer.

Input

- On the first line two integers, $1 \leq n \leq 10^5$, the number of items for sale in your store, and $1 \leq X \leq 10^9$, the minimum cost specified in the statement.
- On the second line n positive integers, each at most 10^9. These are the prices of the items in the store.

Output

Print the maximum number of items you can mark as part of your special offer, without anyone actually being able to receive a horn.

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

Sample Input 2	Sample Output 2
5 10 4 8 1 9 7	2

Sample Input 3	Sample Output 3
4 10 1 3 1 7	4

Sample Input 4	Sample Output 4
1 5 6	1

B Birthday Boy

Time limit: 1s

Bobby has just joined a new company, and human resources has asked him to note his birthday on the office calendar. Bobby the Birthday Boy wants to feel special! Also, Bobby the Birthday Boy does not mind lying for attention.

He notices that the longer people have not celebrated a birthday or eaten cake, the more they like it when a new one comes around. So he wants to pick his birthday in such a way that the longest period of time without a birthday possible has just passed. Of course he does not want to share his birthday with any colleague, either.

Can you help him make up a fake birthday to make him feel as special as possible? Bobby does not care about leap years: you can assume every year is not a leap year, and that no one has a birthday on the 29th of February. In case of a tie, Bobby decides to fill in the date that is soonest (*strictly*) after the current date, the 27th of October, because that means he will get to celebrate his birthday as soon as possible.

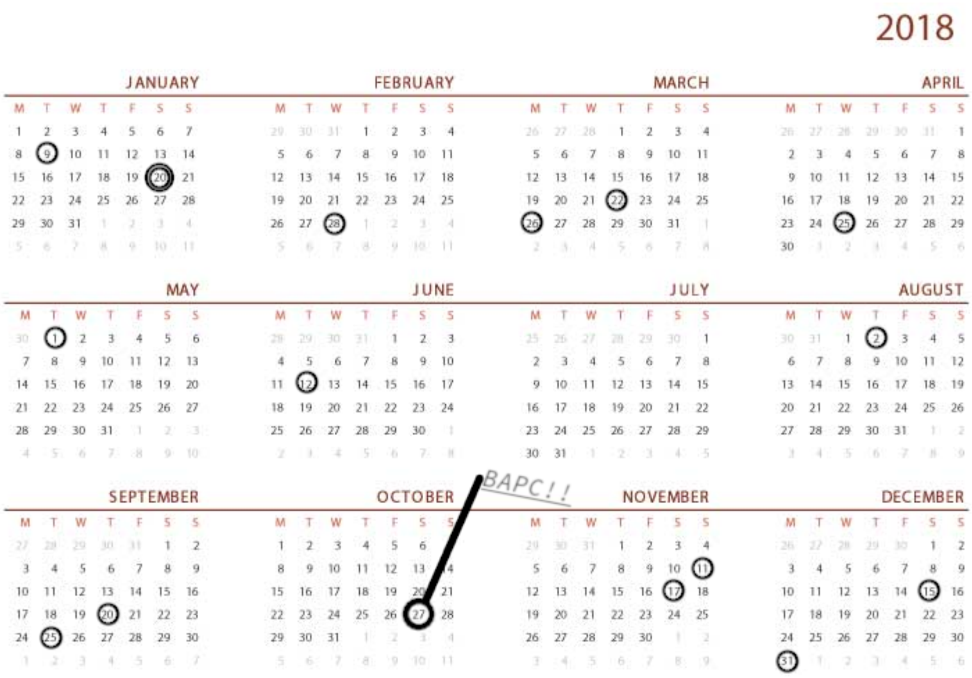


Figure 1: Sample case 2. Calendar is from `http://printablecalendarholidays.com`.

Input

- One line with a number $1 \leq n \leq 100$, the number of colleagues Bobby has in his new office.
- Then follow n lines, each line corresponding to one coworker. Each line gives the name of the colleague (using at most 20 upper or lower case letters) separated from their birthday date by a space. The date is in format `mm-dd`.

Output

Print the fake birthday date (format: `mm-dd`) chosen by Bobby.

Sample Input 1	Sample Output 1
3 Henk 01-09 Roos 09-20 Pietje 11-11	09-19

Sample Input 2	Sample Output 2
16 Henk 01-09 Luc 12-31 Jan 03-22 Roos 09-20 Pietje 11-11 Anne 02-28 Pierre 09-25 Dan 12-15 Lieze 11-17 Charlotte 05-01 Lenny 08-02 Marc 04-25 Martha 06-12 John 03-26 Matthew 01-20 John 01-20	08-01

Sample Input 3	Sample Output 3
3 JohnIII 04-29 JohnVI 10-28 JohnIIX 04-28	04-27

Sample Input 4	Sample Output 4
3 CharlesII 04-30 CharlesV 10-29 CharlesVII 04-29	10-28

C Cardboard Container

Time limit: 1s

Fidget spinners are *so* 2017; this years' rage are fidget cubes. A fidget cube is a cube with unit side lengths, which you hold in your hand and fidget with. Kids these days, right?

You work in the planning department for a company that creates and ships fidget cubes. Having done some market analysis, you found that your customers want to receive shipments of *exactly* V fidget cubes.

This means you have to design a container that will hold exactly V fidget cubes. Since fidget cubes are very fragile, you cannot have any empty space in your container. If there is empty space, they might move around, bump into each other and get damaged. Because of this, you decide to ship the fidget cubes in a rectangular cardboard box.

The cost of a cardboard box is proportional to its surface area, costing exactly one unit of money per square unit of surface area. Of course you want to spend as little money as possible. Subject to the above constraints, how much money do you have to spend on a box for V fidget cubes?

Input

The input contains a single integer, $1 \leq V \leq 10^6$, the number of fidget cubes for which you need to build a box.

Output

Print the cost of the cheapest rectangular box as specified in the statement.

Sample Input 1	Sample Output 1
1	6

Sample Input 2	Sample Output 2
4	16

Sample Input 3	Sample Output 3
3	14

Sample Input 4	Sample Output 4
5913	2790

D Driver Disagreement

Time limit: 2s

Alice and Bob are travelling in Italy. They are travelling by car and unfortunately they took a wrong turn. Now they are stuck in the city centre of Pisa. (You may know that you need an allowance to drive in the city centre, so they are at risk of getting a fine.) As they were not fully prepared for this, they have a map, but no GPS. The map lists all intersections. At each intersection you can go either left or right (you cannot go straight or take a U-turn, as many streets are one-way).

Of course, they paid attention when entering Pisa and tried to follow on the map. Unfortunately, Alice thinks they are at intersection A, while Bob believes they are now at intersection B. You can imagine this is quite a stressful situation. Instead of figuring out how to get out of Pisa, they want to know who is right first. On the map it is indicated from which intersections you can see the leaning tower of Pisa. So they believe they can conduct an experiment: drive a bit and take the same actions on the map starting from A and B. They can trace the route they drive on the map for both of their starting points. As soon as the tower of Pisa should be visible for one of them but not for the other, they can look out of the window to see who is right. You may assume exactly one of them is right.

Input

- The first line of the input has three space-separated integers. The first integer, $2 \leq n \leq 10^5$ is the number of intersections. The next two integers are $0 \leq A, B < n$, the intersections that Alice and Bob respectively think they are currently at. In particular $A \neq B$.
- Then follow n lines. The i'th of these lines ($0 \leq i < n$) has three space-separated integers: l_i r_i t_i. If you are at intersection i and take a left turn, you arrive at l_i, while a right turn brings you to r_i. The number $t_i = 1$ if you can see the leaning tower of Pisa from intersection i. Otherwise $t_i = 0$.

Output

Print the minimal number of turns it takes to show either person correct. If no experiment can tell whether Alice or Bob is correct, print "`indistinguishable`".

Sample Input 1	Sample Output 1
3 1 2 1 2 1 0 2 0 0 1 0	indistinguishable

Sample Input 2	Sample Output 2
2 0 1 1 1 1 0 0 0	0

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

E Entirely Unsorted Sequences

Time limit: 4s

You have recently been promoted to lead scientist at NASA, the National Association for Sorting Algorithms. Congratulations! Your primary responsibility is testing the sorting algorithms that your team produces. Fortunately, NASA has a large budget this year, and you were able to buy some state of the art integers you can use to test the sorting algorithms.

Figure 1: Image via Flickr by Heather Paul, 'warriorwoman531', CC BY-ND 2.0.

As the lead scientist, you are well aware that algorithms are tested by their behaviour on worst case inputs. So, to test sorting algorithms, you need sequences that are as unsorted as possible.

Given a sequence of numbers $(a_1, \ldots, a_n)$ we say that an element a_k is sorted if for all indices j such that $j > k$, $a_j \geq a_k$ and for all indices j such that $j < k$, $a_j \leq a_k$. For example, in

$$(1, 3, 2, 3, 4, 6, 5, 5)$$

the sorted elements are the 1, the second occurrence of 3, and the 4. Note that a sequence is sorted if and only if all its elements are sorted. A sequence is called *entirely unsorted* if none of its elements are sorted.

Given a sequence of integers, what is the number of entirely unsorted sequences you can make by permuting its elements? Two sequences $(b_1, \ldots, b_n)$ and $(c_1, \ldots, c_n)$ are considered to be different if there is some index $i \in \{1, \ldots, n\}$ for which $b_i \neq c_i$. Because the number of permutations may be very large, please give it modulo $10^9 + 9$.

Input

The input starts with an integer $1 \leq n \leq 5000$. Then follows a single line with n integers $a_1, \ldots, a_n$, with $0 \leq a_i \leq 10^9$ for all i.

Output

Print a single integer: the number of entirely unsorted sequences you can make by permuting the a_i, modulo $10^9 + 9$.

Sample Input 1	Sample Output 1
4 0 1 2 3	14

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

Sample Input 3	Sample Output 3
`13` `1 2 3 4 5 6 7 8 9 10 11 12 13`	`298600727`

F Financial Planning

Time limit: 3s

Being a responsible young adult, you have decided to start planning for retirement. Doing some back-of-the-envelope calculations, you figured out you need at least M euros to retire comfortably.

You are currently broke, but fortunately a generous gazillionaire friend has offered to lend you an arbitrary amount of money (as much as you need), without interest, to invest in the stock market. After making some profits you will then return the original sum to your friend, leaving you with the remainder.

Available to you are n investment opportunities, the i-th of which costs c_i euros. You also used your computer science skills to predict that the i-th investment will earn you p_i euros per day. What is the minimum number of days you need before you can pay back your friend and retire?

For example, consider the first sample. If you buy only the second investment (which costs 15 euros) you will earn $p_2 = 10$ euros per day. After two days you will have earned 20 euros, exactly enough to pay off your friend (from whom you borrowed 15 euros) and retire with the remaining profits (5 euros). There is no way to make a net amount of 5 euros in a single day, so two days is the fastest possible.

Input

- The first line contains the number of investment options $1 \leq n \leq 10^5$ and the minimum amount of money you need to retire $1 \leq M \leq 10^9$.
- Then, n lines follow. Each line i has two integers: the daily profits of this investment $1 \leq p_i \leq 10^9$ and its initial cost $1 \leq c_i \leq 10^9$.

Output

Print the minimum number of days needed to recoup your investments and retire with at least M euros, if you follow an optimal investment strategy.

Sample Input 1	Sample Output 1
2 5 4 10 10 15	2

Sample Input 2	Sample Output 2

`4 10`
`1 8`
`3 12`
`4 17`
`10 100`
	`6`

Sample Input 3	Sample Output 3

`3 5`
`4 1`
`9 10`
`6 3`
	`1`

G Game Night

Time limit: 1s

It is finally Bobby's birthday, and all of his Acquaintances, Buddies and Colleagues have gathered for a board game night. They are going to play a board game which is played in up to three big teams. Bobby decided to split his guests into how well he knows them: the Acquaintances on team A, the Buddies on team B, and the Colleagues on team C.

While Bobby was busy explaining the rules to everyone, all his guests already took seats around his large, circular living room table. However, for the game it is crucial that all people sitting on a team are sitting next to each other. Otherwise, members of other teams could easily eavesdrop on their planning, ruining the game. So some people may need to change seats to avoid this from happening.

Bobby wants to start playing the game as soon as possible, so he wants people to switch seats as efficiently as possible. Given the current arrangement around the circular table, can you figure out the minimal number of people that must switch seats so that the teams are lined up correctly?

Input

- The first line of the input contains the integer n, where $1 \leq n \leq 10^5$ is the number of players (as well as seats).
- The second line contains a string of length n, consisting only of the characters in `ABC`. This indicates the teams of the people sitting around the table in order.

Output

Print a single integer: the minimal number of people you have to ask to move seats to make sure the teams sit together.

Sample Input 1	Sample Output 1
5 ABABC	2

Sample Input 2	Sample Output 2
12 ABCABCABCABC	6

Sample Input 3	Sample Output 3
4 ACBA	0

Sample Input 4	Sample Output 4
6 BABABA	2

Sample Input 5	Sample Output 5
9 ABABCBCAC	3

H Harry the Hamster

Time limit: 3s

Harry the Hamster lives in a giant hamster cage. Inside the cage there is a set of n plastic balls connected by unidirectional hamster tubes of varying lengths. Harry is currently in ball s and his bed is in ball t.

Figure 1: Image via Flickr by Bill McChesney, 'bsabarnowl', CC BY 2.0.

Being a simple hamster, Harry's brain halves are not so great at communicating with each other and have a mind of their own. Harry's left brain half, usually being active when Harry is in the hamster wheel, likes running for as long as possible. Harry's right brain half, rarely active at all, would like to go to sleep as soon as possible. Together, Harry's brain halves will be navigating Harry through the maze of tubes, in each ball deciding which of the outgoing tubes to follow.

Harry's brain halves get really tired after making a decision and then need to rest a bit, so they cannot make two decisions in a row. Thus, they make decisions on which tube to take in alternating turns, with the left brain half going first. So starting in ball s, Harry's left brain half will decide on a tube to follow, ending in some ball u, where Harry's left brain half will rest and Harry's right brain half will pick an outgoing tube, et cetera.

Counterintuitively, the brain halves are familiar with the entire hamster cage and can plan arbitrarily far ahead. Assuming both brain halves make optimal decisions, how long will it take for Harry to reach his bed? It is guaranteed that each ball has at least one outgoing tube, except the ball containing Harry's bed which has none (there Harry will rest easily). There are no tubes connecting a ball to itself, but there may be multiple tubes going from one ball to another.

Input

- On the first line are four space-separated integers: the number of plastic balls $1 \leq n \leq 10^5$, the number of tubes $0 \leq m \leq 2 \cdot 10^5$, and the locations of Harry and his bed $0 \leq s, t < n$.
- Then m lines follow, each containing three space-separated integers describing a single tube: the ball in which the tube starts $0 \leq a_i < n$, in which it ends $0 \leq b_i < n$ and the time it takes to traverse $1 \leq w_i \leq 10^4$. Note that each tube can only be traversed in one direction.

Output

Print the time it takes for Harry to reach his bed, or the string `infinity` if Harry is doomed to roam the tubes forever.

Sample Input 1	Sample Output 1
4 5 0 3 0 1 1 1 2 2 2 0 4 2 3 1 2 3 3	11

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

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

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

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

I In Case of an Invasion, Please...

Time limit: 3.5s

After Curiosity discovered not just water on Mars, but also an aggressive, bloodthirsty bunch of aliens, the Louvain-la-Neuve municipal government decided to take precautionary measures; they built shelters in order to shelter everyone in the city in the event of an extraterrestial attack.

Several alien-proof shelters have been erected throughout the city, where citizens can weather an alien invasion. However, due to municipal regulations and local building codes the shelters are limited in size. This makes it necessary for the government to assign every citizen a shelter to calmly direct themselves towards in the rare event of a fleet of UFOs blotting out the sun. Conditional on no shelter being assigned more people than it can fit, it is of the utmost importance that the time it takes until everyone has arrived at a shelter is minimized.

We model Louvain-la-Neuve as a network of n locations at which people live, connected by m bidirectional roads. Located at s points throughout the city are the shelters, each with a given maximum capacity. What is the minimum amount of time it takes for everyone to arrive at a shelter, when we assign people to shelters optimally?

The Louvain-la-Neuve municipal government has made sure that there is enough shelter capacity for its citizens and all shelters can be reached from any location, i.e. it is always possible to shelter everyone in some way.

Input

- On the first line are three integers, the number of locations $1 \leq n \leq 10^5$, roads $0 \leq m \leq 2 \cdot 10^5$, and shelters $1 \leq s \leq 10$.
- Then follows a line with n integers $0 \leq p_i \leq 10^9$, indicating the the number of people living at location $1 \leq i \leq n$.
- Then follow m lines containing three integers $1 \leq u, v \leq n$ and $1 \leq w \leq 10^9$ indicating that there is a bidirectional road connecting u and v that takes w time to traverse. For any two locations there is at most one road connecting them directly, and no road connects a location to itself.
- Finally follow s lines with two integers $1 \leq s_i \leq n$ and $1 \leq c_i \leq 10^9$, indicating that there is a shelter with capacity c_i at location s_i.

Output

Print the minimum amount of time it takes to shelter everyone.

Sample Input 1	Sample Output 1

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

Sample Input 2	Sample Output 2

`4 5 2`
`2 0 0 2`
`1 2 6`
`1 3 2`
`2 3 3`
`3 4 4`
`4 2 6`
`3 2`
`2 2`
	`5`

Sample Input 3	Sample Output 3

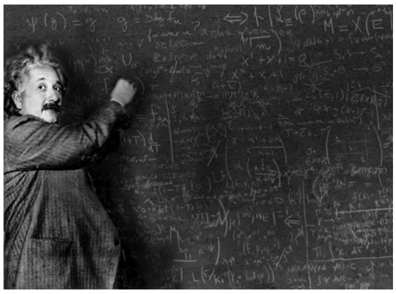
`7 8 3`
`0 1 1 1 1 0 2`
`1 2 1`
`2 3 1`
`3 1 1`
`4 6 5`
`4 3 1`
`6 7 10`
`7 5 3`
`5 6 3`
`6 5`
`1 1`
`2 1`
	`6`

Sample Input 4	Sample Output 4

`2 1 1`
`0 2`
`1 2 1000000000`
`2 2`
	`0`

J Janitor Troubles

Time limit: 1s



While working a night shift at the university as a janitor, you absent-mindedly erase a blackboard covered with equations, only to realize afterwards that these were no ordinary equations! They were the notes of the venerable *Professor E. I. N. Stein* who earlier in the day solved the elusive *maximum quadrilateral problem*! Quick, you have to redo his work so no one noticed what happened.

The *maximum quadrilateral problem* is quite easy to state: given four side lengths s_1, s_2, s_3 and s_4, find the maxiumum area of any quadrilateral that can be constructed using these lengths. A quadrilateral is a polygon with four vertices.

Input

The input consists of a single line with four positive integers, the four side lengths s_1, s_2, s_3, and s_4.

It is guaranteed that $2s_i < \sum_{j=1}^{4} s_j$, for all i, and that $1 \le s_i \le 1000$.

Output

Output a single floating point number, the maximal area as described above. Your answer must be accurate to an absolute or relative error of at most 10^{-6}.

Sample Input 1	Sample Output 1
`3 3 3 3`	`9`

Sample Input 2	Sample Output 2
`1 2 1 1`	`1.299038105676658`

Sample Input 3	Sample Output 3
`2 2 1 4`	`3.307189138830738`

K Kingpin Escape

Time limit: 2s

You are the kingpin of a large network of criminal hackers. Legend has it there has never been a richer criminal than you. Not just because you are the smartest, but also because you are the stingiest.

The police have been after you for years, but they have never been able to catch you thanks to your great set of escape routes. Whenever they want to catch you in one of your many hideouts, you quickly get away through your network of tunnels, back-alleys and speakeasies. Your routes are set up so that from every hideout you have in the city you can get to any other hideout by following only your secret passageways. Furthermore, because you are such a penny-pincher, your network is the smallest possible: from every hideout to every other hideout there is precisely *one* route through the network, no more and no fewer.

Yesterday, your mole in the police force has informed you of an unfortunate fact: the police are on to you! They have gotten wind of your secret network, and will attempt to catch you. They are planning to block some of your escape routes, and catch you in the act. They will start blocking your secret passageways one by one, until you have nowhere left to go.

Fortunately, your headquarters are absolutely safe. If you can just get there, you are always fine. Furthermore, your mole in the police force can inform you immediately as soon as the police start blocking passageways, so that they only have time to block one of them before you get notified. If you get back to your headquarters before they block any more routes, you're safe.

You want to add some passageways to the network so that whenever at most one of them is blocked, you can still get to your headquarters from any other hideout. Since the news has not changed your frugality, you want to extend your network as cheaply as possible. Can you figure out the least number of passageways you need to add, and which ones you need?

Input

- The input starts with two integers $2 \leq n \leq 10^5$, the number of hideouts in the network, and $0 \leq h < n$, the location of your headquarters.
- Then follow $n - 1$ lines, each with two integers $0 \leq a, b < n$, signifying that there is an escape route between location a and location b.

Output

The output consists of:

- An integer m, the least number of escape routes you need to add to make the network safe again.

- Then, m lines with two integers $0 \leq a, b < n$ each, the hideouts between which one of the escape routes has to be added.

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

Sample Input 2	Sample Output 2
6 0 0 1 0 2 0 3 1 4 1 5	2 3 5 2 4