A - Abnormal 94ers

According to a study conducted by Stanford University, these days a lot of people spend more time on social networks like Facebook, Telegram and Viber than they sleep. If they sleep eight hours a day then they spend at least eight hours on these social networks, so they have less than eight hours to eat, study and work!

Amirkabir University of Technology has about 1000 new students each year, and this year newcomers are called 94ers. Suppose 500 of them live like the study mentioned before, so they spend 500*8=4000 hours per day, 4000*30=120,000 hours per month and 120,000*12=1,440,000 hours per year on these social networks altogether!

Samandoon is a 94er and uses Telegram and Viber a lot. He has N friends, T of them use Telegram (these T persons may use Viber too), V of them use Viber (these V persons may use Telegram too), M of them use nor Telegram neither Viber and K of them use both Telegram and Viber. Given N, T, V and M, your job is to calculate K.

Input

Input contains four space separated integers, N, T, V and M in a single line.

$0 \leq N, T, V, M \leq 2015$

Output

Print K in a single line.

Sample 1

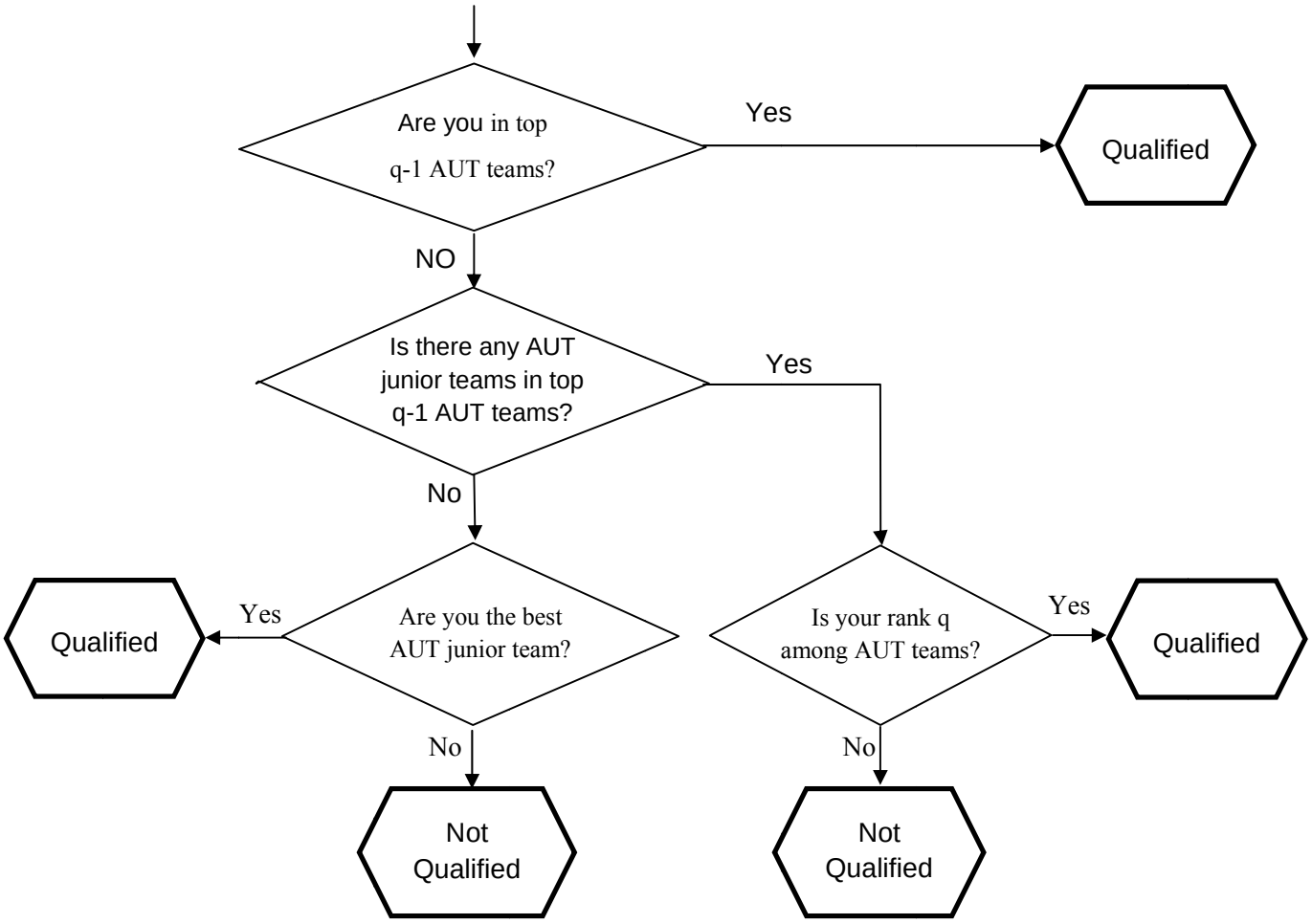
Input	Output
1394 959 733 217	515

Sample 2

Input	Output
4920 2389 3835 295	1599

B - Qualified Teams

After this contest something great happens for AUT teams, which is the best teams from AUT will be selected for Tehran Regional ICPC 2015.
In previous years there was a problem that senior teams performed very well in the selection contest, and junior teams would lose their motivation to practice. To deal with this problem AUT ACM ICPC Committee used a new approach and invented the following flowchart:



We call a team **junior team** if all of its members have entered university in 1393 or 1394.
Because we don't have enough time between the end of the contest and final ceremony, this task is given to you. You should write a program that reads the final ranking and prints the names of the teams that will be qualified for the Tehran Regional ICPC 2015.

Input

First line of input contains two space separated integers n and q ($1 \le q \le n \le 100$), the number of teams in the contest and the number of AUT teams that will be qualified for the Regional contest, respectively. Next n lines contain the ranking. Each line contains one record in the following format (EY = Entrance Year):

Rank<space>TeamName<space>Institute<space>EY1<space>EY2<space>EY3

- $1 \le Rank \le n$
- *TeamName* is a non-empty string of length at most 20, consisting English letters.
- *Institute* is a non-empty string of length at most 8, consisting uppercase English letters.
- $1385 \le EY1, EY2, EY3 \le 1394$

Teams are presented in strictly increasing order of their *Rank*s. It is guaranteed that the number of teams qualifying for the Regional is not greater than the number of AUT teams.

Output

Print the name of AUT teams which will be qualified for the Regional, each name in a separate line in increasing order of their *Rank*s.

Sample 1

Input	Output
10 5 1 PMP AUT 1385 1385 1385 2 DeadlyArmyOfAUT AUT 1386 1387 1388 3 ShootingCrunchers MIT 1391 1390 1390 4 Nanehvis AUT 1387 1387 1387 5 ZuluPiledrivers MANU 1394 1394 1393 6 SafeMen AUT 1387 1388 1388 7 Ooona AUT 1393 1392 1390 8 EccentricSpiders BUT 1391 1390 1392 9 SpiritBusters AUT 1389 1392 1392 10 RedSlayers AUT 1394 1394 1392	PMP DeadlyArmyOfAUT Nanehvis SafeMen Ooona

Sample 2

Input	Output
10 5 1 PMP AUT 1385 1385 1385 2 DeadlyArmyOfAUT AUT 1386 1387 1388 3 ShootingCrunchers MIT 1391 1390 1390 4 Nanehvis AUT 1387 1387 1387 5 ZuluPiledrivers MANU 1394 1394 1393 6 SafeMen AUT 1387 1388 1388 7 Ooona AUT 1393 1392 1390 8 EccentricSpiders BUT 1391 1390 1392 9 SpiritBusters AUT 1389 1392 1392 10 Abnormals AUT 1394 1393 1393	PMP DeadlyArmyOfAUT Nanehvis SafeMen Abnormals

C - Tuxic Queries

The game SuperTux uses lists to store tokens.

A list starts with "(" and ends with ")" (without quotes) and contains non-zero elements delimited by whitespaces. An element can be a *Token* or another list. A *Token* is a non-empty string consisting alphanumeric characters. *Whitespace* is a non-empty sequence of space, tab, or new-line. There can also be *whitespace* right after "(" or right before ")".

For example, the following is a valid list which contains two *Token*s and a sub-list as its members:
(a (b c) d)

The label of a list is its first element, if that element is a *Token*, otherwise its label is an empty string.

We can use the following query language to find information in a list:

- List.T[k], where k is a non-negative integer, returns the k-th *Token* in the list.
  For example, if List is "(a (d e) c)" then, List.T[0] = "a" and List.T[1] = "c".
- List.L[label, k], where label is a non-empty string and k is a non-negative integer, returns the k-th sub-list of the list whose label matches the given label.
  For example, if List is "(a (a b) (c d) (a c) b)" then List.L[a, 0] = "(a b)" and List.L[a, 1] = "(a c)".

Our input is a list and some queries. The queries are a chain of T and L operators. Your task is to write a program which finds the result of the queries.

Input

Input starts with a valid list, which can span over multiple lines. Next line contains the number of queries Q. Each of the next Q lines contains a query.
The list contains at most 1000 characters, and Q is at most 20.

Output

For each query, print its result if the query can resolve to a *Token*. Otherwise, print "FAILED".

Sample 1

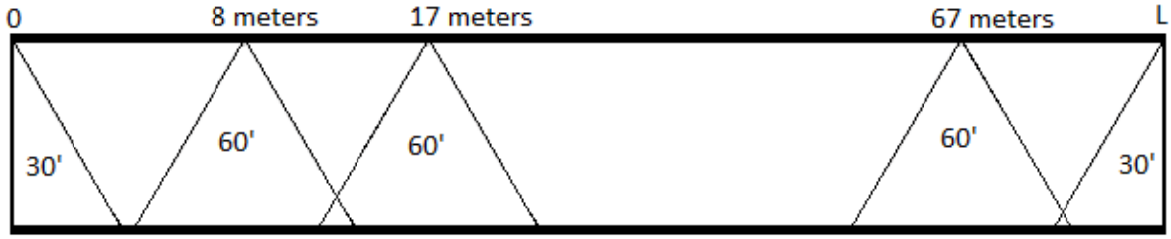
Input	Output
(a (a b) (c d) (a c) b) 3 List.T[0] List.L[a, 1] List.L[a, 1].T[1]	a FAILED c

Sample 2

Input	Output
(sprite (rect 4 5 6 7) (type a)) 5 List.T[0] List.T[1] List.L[type, 0].T[0] List.L[type, 0].T[1] List.L[type, 1].T[0]	sprite FAILED type a FAILED

D - Isn't P equal to NP?!

A few years ago a computer scientist claimed that P is equal to NP. Other scientists who believed P is not equal to NP thought the scientist was mad and with the help of The Supreme Court they sentenced him to life imprisonment!
Right now he is the manager of the electrical equipments in prison (e.g. wires, lamps, ...) and he is also working on his proof.
Last week a prisoner named Mesut escaped this prison and in his way out of the prison passed a corridor. A soldier is standing in a tower and is watching this corridor from distant. Look at the below picture, it's the soldier's view of the corridor.



The corridor has length of **L** and height of **H** and there are **N** lamps installed in the corridor. Now they have asked the scientist to add **K** new lamps in the corridor to maximize the area which is enlightened and the soldier in the tower can watch. All lamps (old and new) are of the same kind, enlightening 60 degrees and should be installed vertically symmetric.
Since the scientist is too busy proving P=NP, he needs your help to do the task.

Input

The first line contains integers *N, K, L and H* ($2 \le N \le 50{,}000$, $0 \le K \le 50{,}000$, $1 \le L, H \le 10^8$). Each lamp has a position which is its distance from the left side of the corridor. The following *N* lines each containing one integer which is the position of a lamp. One of the N lamps is installed at the beginning and one of them is installed at the end of the corridor. No two lamps are installed at the same point. You don't have to choose integer distances for new lamps.

Output

Print **K** lines each containing the position of a new installed lamp in format **p/q** where **p** and **q** are relatively prime integers (follow the format of the output from the test samples). It is guaranteed that p and q will fit in 64 bit signed integer.

Sample 1

Input	Output
3 5 20 10 7 0 20	7/2 48/5 61/5 74/5 87/5

Sample 2

Input	Output
4 7 9 1 4 0 9 2	22/3 3/1 29/6 17/3 49/6 1/1 13/2

E - Lucky names

In ancient Greek myths, it was believed that people whose names were palindrome were lucky and had a longer lifetime than others. This belief was so common in king Rechcer's time. One of his ministers, Renner, invented an evaluation system in order to calculate how much a name is lucky. He named it Palindromic Number (PN) and the greater PN of a name the luckier that name is.
PN is defined as the number of pairs of non-overlapping substrings of the name in a way that concatenation of these strings is palindrome.

(The substring s[i... j] ($1 \le i \le j \le |s|$) of string $s = s_1 s_2 ... s_{|s|}$ is a string $s_i s_{i+1} ... s_j$. For example, substring s[2...4] of string s = "abacaba" equals "bac".)

In a more formal way, PN is the quantity of tuples (a, b, x, y) such that $1 \le a \le b < x \le y \le |s|$ and s[a... b]+s[x... y] is palindrome ("+" is a sign for concatenation of two strings).

A palindrome is a string that can be read the same way from left to right and from right to left. For example, "renner", "z" and "rechcer" are palindromes but "azadeh" and "fatemeh" are not.

The king ordered Renner that PN should be calculated for all names, by doing so, not only his family but also his next generations will be able to choose best names for their children.

Input

Input contains a non-empty string ***S*** which consists of lowercase letters ('a'...'z'), ***S*** contains at most 2000 characters.

Output

Output a single number — the Palindromic Number of the string.

Sample 1

Input	Output
thebattlewillbelegendary	68

Sample 2

Input	Output
aaaabbbb	30

F – Teleport

One day in our weekly meetings for preparing the problem set of this contest, Poopi came up with an interesting problem, but we couldn't generate a good story for it, so here is the problem with no story behind it:
We have a tree, not in our garden, but on a paper. We want to put a teleport between two vertices. Now your job is to find whether the diameter of the tree would decrease or not?
When we put a teleport between two vertices we can move between them immediately (i.e. the distance between them becomes zero).

*"In mathematics, and more specifically in graph theory, a **tree** is an undirected graph in which any two vertices are connected by exactly one path. In other words, any connected graph without simple cycles is a tree."* Wiki
*"The length of the "longest shortest path" (i.e., the longest graph geodesic) between any two graph vertices of a graph is called graph **diameter**."* Wiki

Input

The first line contains an integer n ($1 \le n \le 100{,}000$) — the number of vertices.
Next $n-1$ lines, each contains an edge of the tree in the format "x_i y_i" (without the quotes) ($1 \le x_i, y_i \le n$, $x_i \ne y_i$).
Next line contains an integer q ($1 \le q \le 100{,}000$) — the number of queries.
Next q lines contains the queries in the format "a_i b_i" (without the quotes) ($1 \le a_i, b_i \le n$) where a_i and b_i are the vertices which we want to put a teleport between them (Each query is applied to the original tree separately).

Output

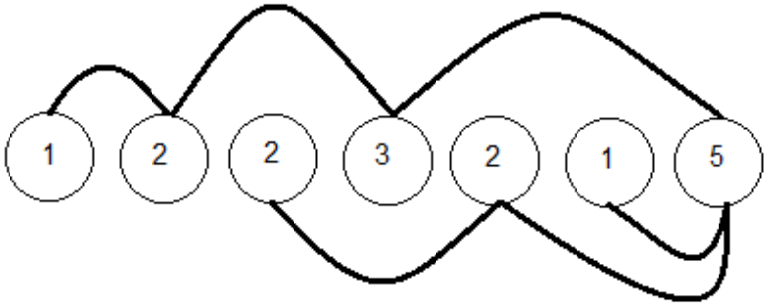
For each query print a single line with "YES" if the diameter decreases, and print "NO" otherwise.

Sample 1

Input	Output
14 1 2 2 3 1 11 13 1 4 6 6 5 6 10 9 12 3 8 9 6 7 13 14 3 2 5 5 7 12 10 8 4 5 11 4 14 12	YES NO NO YES YES

G - Jibali Numbers

Arian's birthday is next Wednesday. He likes a special kind of numbers called Jibali Numbers. You may ask what Jibali Numbers are. If you look at a number as a graph and each digit as a vertex then edges are like this: each digit **d** has an undirected edge to it's right d-th neighbor (if one exists). A number is a Jibali Number if the corresponding graph is **connected**.
For example 1223215 is a 7-digit Jibali Number.



Since MeHdi likes him a lot, he wants to buy all k-digit Jibali Numbers for Arian's birthday. Would you help MeHdi count the number of k-digit Jibali Numbers?

Input

The first line contains integer k ($1 \le k \le 10^6$).

Output

Print a single number --- the number of k-digit Jibali Numbers. Since the answer may get very big, print the answer modulo 1,000,000,007.

Sample 1

Input	Output
1	10

Sample 2

Input	Output
4	60

Sample 3

Input	Output
1000000	274322295

H - The loadstar

Once upon a time, AUT freshmen (1294's) went to a camp. It was night and exactly after their midterms. The students were sad since their math midterm was so hard and most of them had ruined it. Their math professor was in the bus too. She wanted to change their mood and make them happier. So she said:

"Hey guys, I have good news. I want to add x to your grades, but x is different for each student and it will be calculated by this formula:

$$x = \frac{k}{A}$$

But what is K and A? K is a constant which I will tell you later! And A is a number which depends on your answers to my next question!

Look at the sky, can you find the loadstar?"

Ghazal pointed to the star, and said: "It's there!"

The professor: "Yes, and I want you to find some other stars so that if you connect them by line, they will make a simple polygon which surrounds the loadstar. Since your viewpoints are different, I will give you a picture of the sky. So what is A? It is the area of the polygon."

*"In geometry a **simple polygon** is defined as a flat shape consisting of straight, non-intersecting line segments or "sides" that are joined pair-wise to form a closed path." wiki*

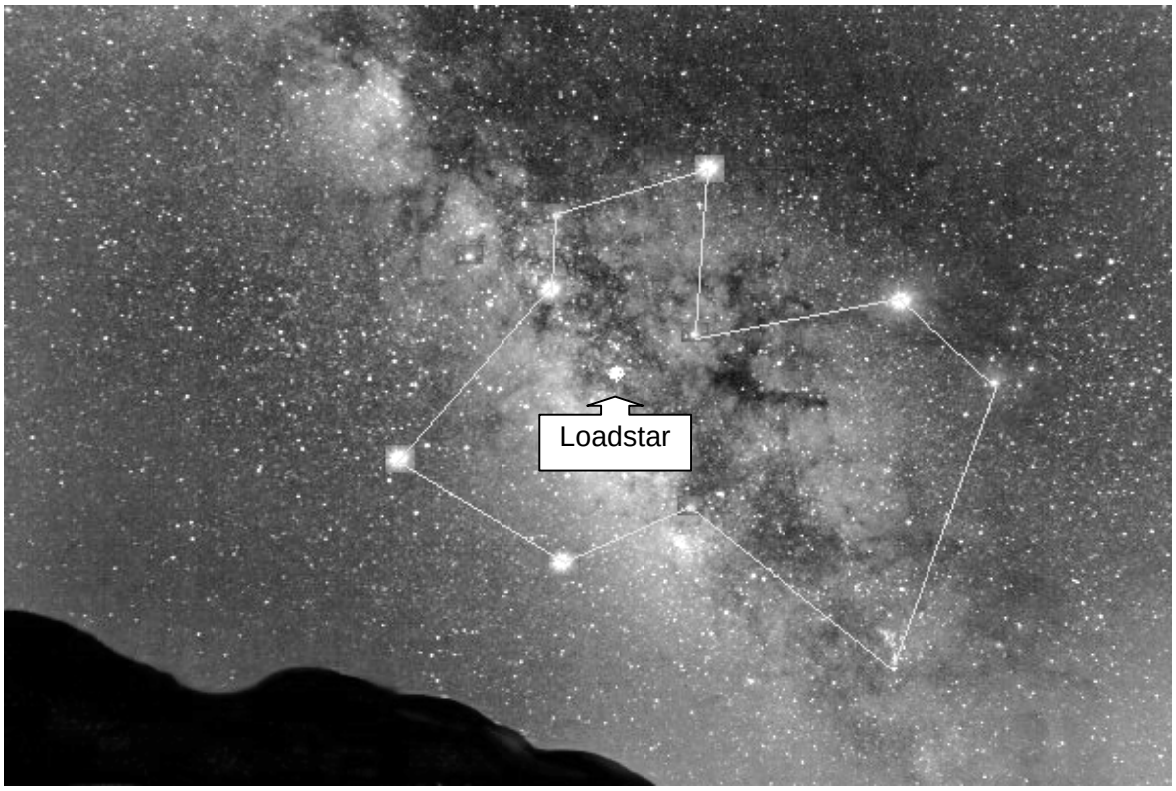


Figure 1- a simple polygon drown by a student

Help Ghazal to find some stars in order to maximize his grade.

Input

Each test begins with an integer n ($1 \leq n \leq 300$) which is the number of stars in the sky. The next line contains two integers x_L and y_L which show the position of loadstar. Each of the next n-1 lines contains two integers x_i and y_i which are the positions of other stars. No two stars have the same coordinates. All coordinates are less than 10^9 by absolute value.

Output

Print a number which is the minimum area of the polygon. Notice that the loadstar's position should be exactly **inside** the polygon. If there is not any polygon surrounding the loadstar, print "Impossible". The answer will be considered correct if the absolute or relative error doesn't exceed 10^{-5}.

Sample 1

Input	Output
5 0 0 0 -1 0 1 -1 2 1 2	2.00000

Sample 2

Input	Output
5 -1 -1 1 1 2 2 3 3 0 10	Impossible

I - The scientist and the card game

Do you remember **the scientist** from problem D? The warden has introduced a new two-player game with a prize of "one week no labor" for the prisoner who can win him in this game.
In this game there are 2N cards each with a certain color and one person distributes them among the players (N cards for each player) and they cannot change the order of the cards (distribution consists of two phases, first the distributor splits the cards in two sets of size N, then reorders each set and gives cards to the players).
There is an empty deck in front of the players. They take turns, in each turn the current player puts the top card from his hands on the deck. After putting his card if the top two cards of the deck have the same color he will take all the cards of the deck and appends them under the cards in his hand. The player who takes all 2N cards wins the game (obviously it's possible that the game never ends).
Knowing that the warden plays first, Khashi, one of the scientist's friends has volunteered to play against the warden. He thinks he can win against the warden in exactly 2N turns (i.e. after 2N turns Khashi has all the cards and the game ends).

The scientist distributes the cards. He wants to help his friend. In how many ways can the scientist split the 2N cards so that there exists a proper reordering for that split? With a proper reordering Khashi would win the game in exactly 2N turns.

When counting splits, two distributions $\boldsymbol{d}_1$ and $\boldsymbol{d}_2$ are the same if for each color $\boldsymbol{c}$, the number of cards of color $\boldsymbol{c}$ in warden's hands are equal in $\boldsymbol{d}_1$ and $\boldsymbol{d}_2$.
If we denote colors with positive integers, these two distributions are the same:

$\boldsymbol{d}_1$={Warden:(1, 1, 2), Khashi:(1, 2, 2)} $\boldsymbol{d}_2$={Warden:(2,1,1), Khashi:(2,1,2)}

but these two are not:

$\boldsymbol{d}_1$={Warden:(1, 1, 2), Khashi:(1, 2, 2)} $\boldsymbol{d}_2$={Warden:(1,2,2), Khashi:(1,1,2)}

If there is any proper distribution for the scientist, find the lexicographically least for him.

As you know in a distribution each player's cards has a fixed ordering. If we denote colors with positive integers, a player's cards are mapped to an array. An *identifier* of a distribution is achieved by appending Khashi's cards at the end of Warden's cards (e.g. if Warden:(1,2,3,4,4) and Khashi:(8,2,4,9,9) then *identifier* is (1,2,3,4,4,8,2,4,9,9)). We sort distributions lexicographically using their identifiers.

Input

The first input line contains a single integer N ($1 \leq N \leq 2000$). The next line contains 2N integers, denoting the color of the 2N cards. There will be at most K ($1 \leq K \leq 2000$) colors which are numbered in range 1 to 2000.

Output

In the first line, print the number of ways the distributor can split the 2N cards so that there exists a proper reordering for that split. Since the answer may get very large print the answer modulo 1,000,000,007. If there is any, in the second line, print space separated identifier of the lexicographically least distribution. Follow sample tests for more details.

Sample 1

Input	Output
2 1 1 2 2	1 1 2 1 2

Sample 2

Input	Output
3 1 3 2 2 2 1	4 1 1 2 2 3 2

Sample 3

Input	Output
4 1 2 3 4 4 3 2 1	13 1 1 2 3 2 4 4 3

J – LoVerS

Elizabeth and Michael live in two different cities. They miss each other a lot and want to meet as soon as possible. There are bidirectional roads between some cities. Traveling each road takes 94 minutes and costs a specific amount of money as the tax.

You should write a program to calculate ***t***, the earliest time they can meet and the minimum amount of money ***c*** that they have to spend totally to meet in ***t*** minutes.

Input

First line of input contains four space separated integers *n*, *m*, *c1* and *c2* ($1 \le n \le 1000$, $1 \le m \le 10{,}000$, $1 \le c1, c2 \le$ n), number of cities, number of roads, Elizabeth's city and Michael's city, respectively. The following *m* lines, each contains three integers x_i, y_i and tax_i, representing a road between city x_i and city y_i with tax_i ($1 \le x_i, y_i \le n$; $x_i \ne y_i$; $1 \le tax_i \le 10{,}000$). You can simply assume there is no more than one road between any two cities.

Output

In a single line, print two space separated integers ***t*** and ***c,*** if it's possible for them to meet. Otherwise, print "NEVER" (without quotes).

Sample 1

Input	Output
6 6 2 6 1 2 1 1 3 5 3 6 6 5 6 3 4 1 2 4 5 1	188 7