

Problem A. Adaptation Stories

Input file: `standard input`
Output file: `standard output`
Time limit: 1 second
Memory limit: 256 megabytes

During the military adaptation period, the candidate DH forgot to bring his water bottle to drill. So that he would not run out of water, the military training leaders lent him a water gallon, affectionately called by `minalbinha`. However it should also be used to resupply the water bottle of the other candidates.

During the training, the military training leaders will bring some amount of water to refill `minalbinha` and the other candidates will take some amount of water from `minalbinha` to refill their bottles.

DH, being the responsible for `minalbinha`, knows the order and the amount of water of every request. He asked your help to determine the minimum amount of water needed to bring on `minalbinha` so that every request can be satisfied.

Consider that `minalbinha` can carry any amount of water.

Input

The first line of input contains one integer, n ($0 \leq n \leq 10^5$) — the number of requests.

The next line contains n integers, v_i ($-1000 \leq v_i \leq 1000$, $v_i \neq 0$) — the amount of water of i -th request. If v_i is negative, the water is being taken from `minalbinha`, otherwise the water is being given to `minalbinha`.

Output

Print the minimum amount of water that is needed to bring on `minalbinha` to fulfill all requests.

Examples

standard input	standard output
7 2 1 -1 -4 3 -5 3	4
3 10 3 -10	0
3 -10 -10 -10	30

Problem B. Before the Great Day

Input file: `standard input`
Output file: `standard output`
Time limit: 1 second
Memory limit: 256 megabytes

Everyone in the boy's dormitory is desperate. No one remembered to choose the items they should carry in their backpacks for training camp and it's already late, so every minute they waste is a minute less of sleep.

On the other hand, all girls are already ready to sleep since they started preparing earlier. Arnon told Diene what was happening and she offered herself to help everyone in the other dorm.

Arnon will tell her the weight of the basic training kit, the number of items available to pick and the name and weight of each item. After that, he is going to tell her the number of boys in the dorm room and their weights.

Everyone was instructed to carry the basic training kit and a subset of the items, but everyone must carry at least half of his weight. Since they are *bizurados* (as the brazilian military say) they know better to carry the minimum weight as possible.

Help her write a program that says which items each boy should carry such that the total weight is minimum and at least half of each boy's weight.

Diene knows that each student has exactly one of each item.

Input

The first line contains two integers, w_{kit} and m ($0 \leq w_{kit} \leq 100$, $1 \leq m \leq 1000$) — the weight of the basic training kit, and the number of different items available to pick.

Each of the next m lines contains a string s_i and one integer w_i ($1 \leq |s_i| \leq 10$, $1 \leq w_i \leq 1000$) — the name and the weight of the i -th item.

The next line contains one integer, n ($1 \leq n \leq 100$) — the number of students that Diene needs to help.

Each of the next n lines contains one integer, ws_i ($2 \times w_{kit} \leq ws_i \leq 2000$) — the weight of the i -th student.

(All weights are in kilogram and yeah, obesity is a real problem).

Output

For each boy, print one line of output containing the number of items the boy will carry followed by the name of each item he will carry, separated with spaces, in any order.

If there are multiple answers, you can print any one of them. If there is no possible answer, print "-1" (without quotes).

Example

standard input	standard output
100 5	1 TeddyBear
TeddyBear 3	1 Apostol
Chocolate 5	0
Apostol 13	2 Chocolate Apostol
Water 10	-1
Pajamas 4	
5	
205	
226	
200	
235	
271	

Problem C. China Adventures

Input file: `standard input`
Output file: `standard output`
Time limit: 1 second
Memory limit: 512 megabytes

The ACM ICPC World Finals is a huge event! On 2018 the World Finals was at Beijing, China.

The chinese culture is way different than brazilian culture. They eat

糟糕鱼片

(as they translated: "Bad fish chips")

and

左中堂香煎鸡

(as they translated: "Sauteed chicken on the left")

Their traffic is crazy, with cars driving on the cycle path, motorcycles driving the wrong way or on the sidewalk. But, the weirdest of all, they use QR-codes for everything.

A QR-code is a square image, in this case a 21×21 pixel matrix, that each pixel is either black or white. The code carries some information, usually an URL or a text, and can be read by almost all modern smartphones.



They use QR-codes so that people don't have to copy text all the time. They just have to open their smartphone's camera and the *magic happens*.

During a tour on Beijing, Lucas Santana, the Handsome, noticed that to visit some tourist places he needed to know the QR-code to buy the tickets for that place. Since he's a very smart guy, he decided to create a mobile application to draw the QR-codes, instead of searching for them.

The mobile application works as follow: it starts with a 21×21 white pixel matrix. You can click any pixel to swap it's color from white to black or from black to white in 1 decisecond. Also, you can restart the application at any time to have an all white pixel matrix again, but it take a deciseconds to do so. When you complete the QR-code, you can assume you can enter the tourist place.

Lucas knows the QR-code of n places he wants to visit. He doesn't care about the order he visits the places, but, since he only have a few days left on China, he wants to spend the least time as possible drawing the QR-codes. Can you help him saying the minimum amount of deciseconds to draw all n QR-codes, the order he should visit the places and when he should restart the application?

He will not visit the same place more than once. (Don't even try this. He will give you a *wrong answer!*).

Input

The first line of input contains two integers, n and a ($1 \leq n \leq 18$, $1 \leq a \leq 50$) — the number of places he wants to visit and the total time to restart the application.

The next lines contains the QR-codes of the places separated by empty lines. There will be n QR-codes, each having 21 lines of 21 digits being 0 or 1. Digit 0 means a white pixel and 1 a black pixel.

Output

Print the minimum amount of deciseconds to draw all QR-codes followed by at least n lines and at most $2 \times n - 1$ lines. Each line should have one number between 1 and n , meaning that he must visit the place i , or one character “*” (without quotes), meaning that he must restart the application.

All numbers should be distinct.

If there are multiple answers, print any of them.

Example

standard input	standard output
1 10 111111100101101111111 100000101101001000001 101110101100101011101 101110100101001011101 101110101000101011101 100000101001101000001 111111101010101111111 000000001111100000000 110100110110001110110 111101000100000110110 111110111110101101101 001110001011001101001 000111110010110100100 000000001001000100000 111111101110011110110 100000100011110001000 101110100101011100010 101110101111000010011 101110100100100110101 100000101010011100000 111111101101100010010	228 1

Note

In sample case, he should just draw the only QR-code available and visit the place. The QR-code is:



Problem D. Dire Grades

Input file: **standard input**
Output file: **standard output**
Time limit: 1 second
Memory limit: 256 megabytes

Students from Instituto Militar de Engenharia (IME) have **a lot** of exams during graduation. Naum, a former student, not being good at programming, requested you to create a program that he could use to verify his means more easily and also be able to change some of his grades.

He will give you n numbers, indicating his grades, and will request q queries. There are two types of queries:

- 1 e v : change the grade of exam e to v
- 2 l r : print the mean of all exams between l and r .

The mean from l to r is defined as $\frac{\sum_{i=l}^r a_i}{r-l+1}$, with a_i being the grade of exam i .

All real numbers will have 2 decimals places.

Input

The first line contains one integer n ($1 \leq n \leq 10^5$) — the number of grades.

The second line contains n space separated **real** numbers $a_1, a_2, \dots, a_n$ ($0.00 \leq a_i \leq 10.00$), where a_i is the grade on exam i .

The third line contains one integer q ($1 \leq q \leq 10^5$).

The next q lines contains, each, one integer t_i ($1 \leq t_i \leq 2$), indicating the type of the query i , and two other values, described below, depending on the query type.

If t_i is 1, the values are one integer e_i and one **real** number v_i ($1 \leq e_i \leq n$, $0.00 \leq v_i \leq 10.00$) — the exam that will change de grade and the new grade of the exam.

If t_i is 2, the values are two integers l_i and r_i ($1 \leq l_i \leq r_i \leq n$) — the interval limits to take the median.

It's guaranteed that at least one query is of type 2.

Output

For every query of type 2, print one line with the mean of all exams in the given interval.

The error between the answer and the printed value must not exceed 10^{-6} in absolute value.

Examples

standard input	standard output
1 10.00 3 2 1 1 1 1 8.50 2 1 1	10.000000 8.500000
3 4.00 6.00 8.00 3 2 1 3 1 2 10.00 2 1 3	6.000000 7.333333

Problem E. Efficient Tracking

Input file: **standard input**
Output file: **standard output**
Time limit: 1 second
Memory limit: 256 megabytes

The student Navarro was called to be the company commander. As company commander, he is responsible for tracking where each member of the company is.

As there are many people, he decide to split the company in several layers and each person in the layer i has his/her location tracked by someone in the layer $i - 1$. Navarro is the only person in the layer 0.

When he was splitting the people, he notice that each pair of cadets has a friendship level and the job is done better when the sum of the friendship levels between each person and the responsible for tracking his/her location is bigger.

Since he is very busy doing upsolvings of the contests and homeworks, he asked you to code a program for him that calculates the maximum friendship level possible for the whole company and tell, for each person, which other person will track down his/her location in this configuration.

The friendship level of a company configuration is defined as the sum of all friendship levels between each person and the responsible for tracking his/her location.

Input

The first line of input contains one number n ($2 \leq n \leq 10^3$) - the number of people in the company.

The next $n - 1$ lines contains the friendship levels. The i -th line contains i integers, f_{ij} ($0 \leq f_{ij} \leq 10^5$) - the j -th integer is the friendship level between $(i + 1)$ -th person and j -th person.

You can assume that the friendship level is reciprocal. In other words, the friendship level between i -th person and j -th person is the same as the friendship level between j -th person and i -th person.

Navarro is always the person number 1 and is the only person that is not tracked down by the others.

Output

Print the maximum friendship level of the company followed by $n - 1$ lines. On i -th line you should print the responsible for tracking the location of the $(i + 1)$ -th person.

If there are multiple answers, print any of them.

Examples

standard input	standard output
3 1 2 3	5 3 1
3 2 3 1	5 1 1
4 3 4 5 3 2 1	12 3 1 1

Problem F. First Day

Input file: **standard input**
Output file: **standard output**
Time limit: 1 second
Memory limit: 256 megabytes

IME students are going to their first military field training. They will be placed in certain areas, individually, but a student in a certain area may be able to see other areas, so, in order to avoid distractions, no student should be able to see any other.

Earlier, for field reconnaissance, soldiers were placed in all areas. During that, two **IMPORTANT** things were noticed:

- The sum of the number soldiers that each soldier could see is $2 \times n - 2$, where n is the number of soldiers;
- If an officer says to any soldier to raise his hand and to all others to only raise theirs if they see another soldier with their hand already raised, eventually all soldiers will have raised their hand.

IME instructors are asking you to help them to find out how many students at most can be placed at the same time. To do so, they will give you which areas can be seen from each area.

Input

The first line contains two integers, n and m ($1 \leq n \leq 10^5$, $0 \leq m \leq \min(10^5, n \times (n - 1)/2)$) — the number of areas and the number of relations.

The next m lines contains the relations between the areas. Each line has two integers, a_i and a_j ($1 \leq a_i, a_j \leq n$), meaning that students in the areas a_i and a_j can see each other.

Output

Output the number of students that can do the activity at the same time without seeing others.

Example

standard input	standard output
5 4 1 2 1 3 2 4 2 5	3

Problem G. Greatest IME

Input file: `standard input`
Output file: `standard output`
Time limit: 1 second
Memory limit: 256 megabytes

Many programming competitions are done in teams, usually having 3 contestants per team, and the contestants from the same group have to sit on adjacent chairs to be able to discuss problems and pair-program.

(*The second greatest*) IME (Instituto de Matemática e Estatística), has a lot of teams. For this year's tryouts, they decided that each team will have 3 contestants, the chairs will be positioned in a row and each contestant will be able to sit anywhere they want (but the contestants of the same team must sit adjacent and no two contestants can sit on the same chair).

There are n teams competing and c chairs positioned in a row.

Since the number of ways that contestants may sit is too large and they don't know how to calculate it, they asked students from (*the greatest*) IME (Instituto Militar de Engenharia), to calculate it for them.

Two ways are considered distinct if at least one contestant is sitting on a different chair on both ways.

As the answer can be very large, so you must print it modulo $10^9 + 7$.

Input

The first line and only line contains two integers, n and c ($1 \leq n \leq 10^5$, $3 \times n \leq c \leq 3 \times 10^5$) — the number of teams and the number of chairs, respectively.

Output

Print the number of ways the contestants can sit modulo $10^9 + 7$.

Examples

standard input	standard output
1 4	12
2 8	432
3 11	12960

Problem H. Holidays

Input file: `standard input`
Output file: `standard output`
Time limit: 1 second
Memory limit: 256 megabytes

Another extended holiday and Lieutenant Calazans is sad because she can't travel again.

There are a lot of places that she lived at and would like to visit again, but all these places are very hard to go to and usually a lot of flight connections are needed to get there. Because of this, she decided to plan ahead the trip for the next holidays.

As the holidays are very short, she doesn't want to make too many flight connections. Besides that, because of the delays on the salary payment of the Lieutenants on the 5th year, she also wants to spend the least amount of money as possible.

As the Lieutenant is very busy organizing the classes to IME++, she asked you to create a program that, given all m possible flights available, their costs and the list of q cities she wants to visit on future holidays with the maximum number of flight connections to get there, should say the minimum cost to reach each destination departing from Rio de Janeiro.

Consider that the flights are always available and it's guaranteed that no two flights have the same route, in other words, for any two distinct flights available (a_i, b_i) and (a_j, b_j) , then $(a_i, b_i) \neq (a_j, b_j)$.

She will only visit one destination on each holiday then come back to Rio de Janeiro after the holiday. The costs are only for going to the destination so you can let her plan her travel back.

Input

The first line of input contains three integers, n and m ($2 \leq n \leq 10^3$, $0 \leq m \leq 10^3$) — the number of cities and number of flights.

The next m lines contains three integers, a_i , b_i and c_i ($1 \leq a_i, b_i \leq n$, $a_i \neq b_i$, $0 \leq c_i \leq 10^3$) — the departure city of flight i , the destination city of flight i and the cost of flight i .

The next line contains one integers, q ($1 \leq q \leq 10^3$) — the number destination cities.

The next q lines contains two integers, d_i and k_i ($2 \leq d \leq n$, $0 \leq k \leq n - 2$) — the destination city she wants to go on the i -th holiday and the maximum number of flight connections to get there.

Rio de Janeiro is always city number 1.

Output

For each destination city in the list, print “=]” (without quotes) followed by the minimum cost to reach the destination from Rio de Janeiro, if it's possible to reach it. Otherwise, print “=[” (without quotes).

Examples

standard input	standard output
3 2 1 2 100 2 3 100 2 2 0 3 0	=] 100 =[
5 7 1 2 400 1 3 100 3 2 200 1 4 400 3 4 100 3 5 250 4 5 100 6 5 0 5 1 5 2 2 0 2 1 3 3	=] =] 350 =] 300 =] 400 =] 300 =] 100

Problem I. Infantrymen's Math

Input file: **standard input**
Output file: **standard output**
Time limit: **1 second**
Memory limit: **256 megabytes**

In 2048, IME still is a military institute, but since the students started to look down on the instructors because of their questionable math knowledge, lots of Math classes are given to the instructors to make them able to command the students.

The students' duty schedule remains the same as it's always been. At night, some students are given some kind of job to make sure the others have a good night of sleep. There is a student designated to write this schedule, responsible of making a fair division between students so they won't be overworking. In 2048, this student is Xavi, a really nice, sweet, athletical, smart, good-looking guy.

Colonel Hector, commander of IME students in 2048, is trying to innovate the students' duty schedule. Since every student has a different ID number from 0 to $n - 1$ (being n the number of students), he gave Xavi a number k , and Xavi has to schedule the duties calling the students by their ID in a sequence s such that:

$$s_0 = 0$$
$$s_i = (s_{i-1} + k) \% n$$

(% is the modulo operator).

For example, if $n = 7$ and $k = 2$, the order of students on duty is the following sequence: 0, 2, 4, 6, 1, 3, 5, ...

Although Colonel Hector just wants to apply some math stuff he learned, Xavi realized a problem: It could happen that some students are never called to work, and some others are called lots of times! Lucky as Xavi is, he realized the number of students is prime. Because of that, he discarded this possibility, but Colonel isn't sure about how fair is this distribution and decided to use another (a really bad and unfair one!) method to schedule the work. Xavi tried to convince Colonel to maintain this scheduling method, but the explanation got very messy and Colonel Hector didn't believe Xavi. He decided to show the distribution of work in a program to the Colonel.

You are the guy who will help Xavi (seems that you are stuck on IME forever...). He will tell you some queries, each having the number of students called to duty and the ID of one of the students, and you have to say how many times the student with ID given will be scheduled.

Input

The first line of input gives three integers n , k , q ($2 \leq n, k \leq 10^9$, $1 \leq q \leq 10^5$) — the amount of students (IME has grown a lot!), the value of k that Colonel Hector gave and the number of queries. It is guaranteed that n is prime.

Then, q lines follow, each containing two integer c_i , d_i ($1 \leq c_i \leq 10^9$, $0 \leq d_i < n$) — the number of students called to duty and the ID of the student.

Output

For each query output how many times the student with ID d_i has been called, after c_i calls for service.

Examples

standard input	standard output
7 1 3 1 0 12 4 14 0	1 2 2
13 3 7 3 2 4 1 12 0 12 5 50 4 200 6 9 7	0 0 1 1 4 16 0

Problem J. JHADCBEIGF

Input file: **standard input**
Output file: **standard output**
Time limit: **1 second**
Memory limit: **256 megabytes**

The members from Incomprehensive Message Encryptors (IME) are learning a new cipher. This technique, although kinda weird, consists of using a key k that is a permutation of the alphabet. In other words, if the alphabet Σ is a sequence of n symbols $a_1, a_2, \dots, a_n$, the key k is a sequence of n symbols $a_{p_1}, a_{p_2}, \dots, a_{p_n}$, for some permutation $p_1, p_2, \dots, p_n$ of $1, 2, \dots, n$.

They learned to use the key to encrypt a string S by the following technique: for each i in $1, 2, \dots, n$, they switch each occurrence of the character a_i in the string S by the corresponding character a_{p_i} in the key k . This substitution occurs for all characters at once, so every substitution doesn't influence the other ones.

For example, if $\Sigma = \{A, B, C, D, E\}$ and $k = \{E, D, C, A, B\}$, the encryption will change every symbol A to E , B to D , C to C , D to A and E to B . So the string $ABCBAD E$ after encryption turns into $EDCDEAB$.

After learning the encryption, the members received m tasks to execute. There are three types of tasks:

- 1 s : add the string s to the dictionary;
- 2: for each string currently in the dictionary, change it by its encrypted version;
- 3 s : check if the string s is prefix of some string in the dictionary.

Because of how complex it is to encrypt it manually, and they are not really good at programming (by their incomprehensiveness), they need someone to solve these tasks for them. That's why they gave you this responsibility.

The alphabet is always the first n uppercase English letters in alphabetical order.

Input

The first line of input contains two integers n, m ($1 \leq n \leq 26, 0 \leq m \leq 10^5$) — the size of the alphabet and the number of tasks.

The second line contains one string k ($|k| = n$) — the initial key. It's guaranteed that k is a permutation of the alphabet.

The next m lines contains, each, one integer t_i ($1 \leq t_i \leq 3$) — the type of the task i .

If t_i is 1, the line also contains one string s_i ($1 \leq |s_i| \leq 5 \times 10^5$) — the string to be added to the dictionary.

If t_i is 3, the line also contains one string s_i ($1 \leq |s_i| \leq 5 \times 10^5$) — the string to be checked against the dictionary. It's guaranteed that s_i only contains symbols from the alphabet.

It's guaranteed that there's at least one task of type 3 and that $\sum |s_i| \leq 5 \times 10^5$.

Output

For each task of type 3, print "YES" if the string is a prefix of some string in the dictionary, and "NO" otherwise.

standard input	standard output
5 10	YES
ECABD	NO
1 ABACABA	NO
1 EAAE	YES
2	NO
1 AADE	
3 ECEAECE	
3 DED	
2	
3 EAAE	
3 BDDB	
3 AB	
5 12	YES
BCDEA	NO
1 AAAAA	NO
3 AAAAA	NO
2	NO
3 AAAAA	YES
2	
3 AAAAA	
2	
3 AAAAA	
2	
3 AAAAA	
2	
3 AAAAA	