Problem A. Innovative Experiment

Input file:	`input.txt`
Output file:	`output.txt`
Time limit:	1 second
Memory limit:	64 megabytes

A shuffle machine is the special device used in a casino to shuffle card decks. Just one key pressed, and cards will be automatically shuffled so that it will be completely impossible to say where each card is.

A world famous scientist Innokentiy solved the mystery of the shuffle machine. He realised that if a deck of n cards numbered in the order $1, 2, \ldots, n$ is put into the machine, it will be shuffled in such a way that deck $p_1, p_2, \ldots, p_n$ will form after the key is pressed. Here, numbers $p_1, p_2, \ldots, p_n$ are the same numbers of cards but rearranged in some order by a shuffle machine. Each shuffle machine has its own numbers $p_1, p_2, \ldots, p_n$.

Innokentiy noticed that if one presses the shuffle key several times, the cards in the deck sooner or later return to their start order. Now the scientist wants to design such a machine that if its key is pressed exactly k ($k \geq 1$) times, the order of cards is not changed, but if it is pressed less than k times, that is, l times ($1 \leq l < k$), the cards never return to the initial order. Innokentiy is free to choose any number of cards n in the deck, but it should not exceed 10^5 because he has a limited amount of money for his experiment.

Input

The input contains the only integer k ($1 \leq k \leq 10^{18}$). It is the minimal number of presses so that the order of cards in the deck will not change.

Output

In the first line output integer n ($1 \leq n \leq 10^5$) — the number of cards in the deck that the world famous scientist Innokentiy should use in his experiment.

In the second line output n different integers $p_1, \ldots, p_n$ ($1 \leq p_i \leq n$) characterizing Innokentiy's shuffle machine.

If there are many possible solutions, output any of them. If the scientist is unable to design such a machine, output «`No solution`» (without quotes).

Examples

input.txt	output.txt
3	3 2 3 1
6	8 3 5 1 4 7 8 2 6
1000000007	No solution

Note

Let's see the first sample. Here $k = 3$, and one of the possible solutions is $p_1 = 2$, $p_2 = 3$, $p_3 = 1$.

At the beginning deck 1, 2, 3 is put into the shuffle machine.

After the first shuffling the order of cards will be 2, 3, 1.

After the second shuffling — 3, 1, 2.

And finally, after the third shuffling it will be 1, 2, 3, that is equal to the start order.

Problem B. Epic Battle

Input file:	`input.txt`
Output file:	`output.txt`
Time limit:	1 second
Memory limit:	64 megabytes

Constantine and Mike, two users of a little-known website Codehorses, cannot decide who is more cool. That's why they play one interesting game. The winner takes it all!

At the start of the game n stone piles, where the i-th pile contains a_i stones, lie on the table. During his move a player can split any pile that has no less that 2 stones, into two non-empty piles. If someone cannot move, he loses.

Both players play optimally. Constantine starts the game. Determine who will win this epic battle!

Input

The first line contains one integer n ($1 \le n \le 20$) — the number of piles on the table at the start of the game.

The second line contains n integers a_i ($1 \le a_i \le 40$) separated by space — the numbers of stones in the piles.

Output

Print «`Constantine`» if Constantine wins the game and «`Mike`» if it is Mike.

Examples

input.txt	output.txt
3 2 2 2	Constantine
2 5 1	Mike

Problem C. Pink Elephants

Input file:	`input.txt`
Output file:	`output.txt`
Time limit:	1 second
Memory limit:	64 megabytes

Alexey loves to sleep very much. And when he wants to fall asleep quicker he counts pink elephants.

Alexey usually counts n pink elephants before he falls asleep, and the heights of all these elephants are natural numbers not exceeding m. But this time elephants were specific: any of them was strictly taller than the previous one! Help Alexey to calculate the number of such sequences of pink elephants where any pink elephant is taller than the previous one.

Input

The only one line of the input contains 2 integers n and m ($1 \le n, m \le 200$) — the number of pink elephants and their maximum possible height.

Output

Print the answer to the problem modulo $1\,000\,000\,007$ ($10^9 + 7$).

Examples

input.txt	output.txt
1 1	1
2 1	0
2 4	6
3 4	4

Note

In the first sample only one such sequence exists: $\{1\}$.

In the second sample $m < n$: there are no such sequences, even of pink elephants.

In the third sample the possible sequences are $\{1, 2\}$, $\{1, 3\}$, $\{1, 4\}$, $\{2, 3\}$, $\{2, 4\}$, $\{3, 4\}$.

And here are the possible sequences for the fourth sample: $\{1, 2, 3\}$, $\{1, 2, 4\}$, $\{1, 3, 4\}$, $\{2, 3, 4\}$.

Problem D. Broadcasting

Input file:	`input.txt`
Output file:	`output.txt`
Time limit:	1 second
Memory limit:	64 megabytes

Professor Dmitry likes parallel programming. However, parallel algorithms are usually complicated and require much attentiveness. Situation when necessary data located on only one computer but it should be located on all computers to continue the program execution arises very often.

Professor Dmitry has n computers, each of them is connected to each other. One computer in one second can send the information to no more than k other computers, and after that these k computers will be able to send this information like the first one. At the beginning, professor has necessary data on only one computer. Help him to calculate how much time he needs to send this data to all his computers.

Input

The only line of the input contains 2 integers n and k ($1 \le n, k \le 10^9$) separated by space. They are the number of professor Dmitry's computers and the maximum number of computers to which the specific computer can send information in one second, correspondingly.

Output

Print one integer — minimum time in seconds that is needed to send the information from one computer to all others.

Examples

input.txt	output.txt
4 1	2
10 2	3
1000000000 999999999	1

Problem E. Tests Preparation

Input file:	`input.txt`
Output file:	`output.txt`
Time limit:	1 second
Memory limit:	64 megabytes

While preparing one of this contest's problems, the jury found out that there exist a lot of wrong solutions that are difficult to challenge. The jury wrote n tests and m wrong solutions each of them fails at least at one of these tests.

Now the jury wants to minimize the number of tests in such a way that all m solutions will fail at least at one of these tests. Of course, the participants will help the jury to do it!

Input

The first line of the input contains 2 integers n and m ($1 \le n \le 20$, $1 \le m \le 60$) — the number of tests and wrong solutions correspondingly.

Each of next n lines describes the i-th test and contains integer k ($1 \le k \le m$) — the number of solutions that fail the i-th test, followed by k different integers — the numbers of these solutions. The solutions are numbered from 1 to m.

It is guaranteed that each of the m solutions fails at least at one of the n tests.

Output

In the first line print one integer t — the minimum number of tests that is needed to challenge all m wrong solutions.

In the second line write t different integers — the numbers of these tests.

If there are many minimal testsets, you can output any of them.

Examples

input.txt	output.txt
3 5 2 2 1 1 4 2 3 5	3 1 2 3
3 3 3 1 2 3 3 2 3 1 3 3 1 2	1 1
3 3 2 1 2 2 2 3 2 3 1	2 1 2
3 3 2 2 3 1 1 2 1 3	2 1 2

Problem F. Magic Chains

Input file:	`input.txt`
Output file:	`output.txt`
Time limit:	2 seconds
Memory limit:	128 megabytes

Have you ever played «Magic Chains» game? No? Well, you have to do it now.

Magic Chain is a finite sequence of words, in which neighboring words differ in exactly one letter. For example, you have a word «HARE». You can change any letter in this word for some other letter, so as to get an existing word. In the new word you also change a letter and get the next word, ans so on. Your task is to get the specified word, for example, «WOLF».

In this problem you have a dictionary containing all words of some language. For simplicity sake all words consist of small Latin letters. You should find the shortest magic chain starting from the first word of the given dictionary and ending with the last one.

Input

The first line contains an integer n ($2 \le n \le 60\,000$) — the number of words in the dictionary.

The next n lines contain words one at a line. All words are different, consist of small Latin letters and have the same length, which isn't exceeding 10 characters.

Output

In the first line write an integer m — the number of words in the shortest magic chain.

In the next m lines write the magic chain one word at a line, including the first one and the last one.

If there are many answers, write any of them. If there is no such magic chain, in the single line write the word «`FAIL`» (without quotes).

Examples

input.txt	output.txt
7 voda yoda vina boda vona beda vino	4 voda vona vina vino
24 muha mura tura tara kara kare kafe kafr kaur kauk klon kruk klan uruk koan urok korn srok kora stok fora ston fura slon	10 muha mura fura fora kora korn koan klan klon slon
2 mother father	FAIL

Problem G. Procrastination

Input file:	`input.txt`
Output file:	`output.txt`
Time limit:	1 second
Memory limit:	64 megabytes

In psychology, *procrastination* refers to the act of replacing high-priority actions with tasks of lower priority, or doing something from which one derives enjoyment, and thus putting off important tasks to a later time.

Student Nikita suffers from procrastination. Tomorrow he has a test, but he hasn't learned any question yet. Suddenly, he realized that he has no time to learn all questions — he can learn only k questions in the remaining time. A test variant contains n questions, the i-th question belongs to the i-th subject. The i-th subject contains c_i questions, each of them can be in Nikita's test variant as the i-th question with equal probability.

Nikita believes that if he learns a question, he can answer it right. He decided to choose questions for learning in such way, that the expected value of the number of the answered questions is maximal. Find, how many questions from each subject he must learn to reach his goal.

Input

The first line contains two space-separated integers k and n ($0 \le k < 200000$, $1 \le n \le 2000$) — the number of questions Nikita can learn and the number of questions in a test variant.

The second line contains n space-separated integers c_i ($1 \le c_i \le 100$), c_i is the number of questions in the i-th subject. It is guaranteed that Nikita has no time to learn all questions, i.e. $k < \sum\limits_{i=1}^{n} c_i$.

Output

Print n integers, i-th integer is the number of questions chosen for learning in the i-th subject. If there are multiple ways to get the maximal expected value, output any of them.

Examples

input.txt	output.txt
0 1 3	0
2 1 3	2
2 2 3 2	0 2

Note

In the third sample test it is most advantageous to learn two questions of the second subject. So, the expected value of the number of the right answers is $\frac{0}{3} + \frac{2}{2} = 1$.

If Nikita chooses both two questions from the first subject, the expected value is $\frac{2}{3} + \frac{0}{2} = \frac{2}{3}$.

At last, if he chooses one question from each subject, the expected value of the number of the right answers is $\frac{1}{3} + \frac{1}{2} = \frac{5}{6}$, which is also less than 1.

Problem H. The Longest Good Substring

Input file:	`input.txt`
Output file:	`output.txt`
Time limit:	1 second
Memory limit:	64 megabytes

Sometimes, in string problems you are to find a string, satisfying some conditions. This time authors were too lazy to think of a tricky term for such string, so they called it *good*.

A string is called good, if it contains no more than k different symbols. Find the longest good substring of the given string.

Input

The first line of input contains integer k ($1 \le k \le 26$).

The second line contains non-empty string s, consisting of lowercase Latin letters. The length of s does not exceed 10^5.

Output

Output two integers l and r ($l \le r$), such that substring $s_l \ldots s_r$ (indexing is 1-based) is good, and its length (which equals $r - l + 1$) is maximal of all good substrings of s.

If there is more than one solution, output any of them.

Examples

input.txt	output.txt
`1` `a`	`1 1`
`3` `abrakadabra`	`4 8`
`5` `allyourbasearebelongtous`	`7 16`

Problem I. Prohibition

Input file:	`input.txt`
Output file:	`output.txt`
Time limit:	1 second
Memory limit:	64 megabytes

Many people know Berland, the country that has been described at Saratov's contests for ten years. But this problem tells us about another country — Beerland. Beerland has n cities and $n-1$ bidirectional roads such that any city is reachable from the others, and the distance between two cities that are directly connected by a road can be passed in one day.

Despite the country's name, its new president introduced a prohibition. It is clear that atmosphere in the cities was heated up after that. To prevent the rebellion, the president decided to place military squads at some cities in such a way that any city, if the rebellion fires up there, could be protected by some squad no later than in one day. Of course, the president cares about the budget of the country, so the number of military squads should be minimal.

Input

The first line of the input contains one integer n ($1 \le n \le 100$) — the number of cities in Beerland.

Each of next $n-1$ lines contains two integers a and b ($1 \le a, b \le n$, $a \ne b$) — the numbers of directly connected cities. Each road is described exactly once.

Output

Output n numbers separated by space. The i-th number should be equal to 1 if the president should place a military squad in the i-th city, and 0 otherwise.

If there are many possible solutions, you can output any of them.

Examples

input.txt	output.txt
3 1 2 2 3	0 1 0
4 4 2 2 1 1 3	1 1 0 0
7 1 2 2 3 3 5 4 2 5 7 6 5	0 1 0 0 1 0 0

Problem J. Secret Laboratory

Input file:	`input.txt`
Output file:	`output.txt`
Time limit:	1 second
Memory limit:	64 megabytes

Hillys is in danger! Aliens called the DomZ raid the settlements and steal the citizens. However, most people are waiting for the quick end of the war because they are absolutely sure that elite military forces, Alpha Sections, successfully protect the people of the planet.

Jade, a twenty-year old girl, who belongs to a rebel organization called IRIS Network, could get to the secret Alpha Sections' plant. The rebels think that some important evidence that will reveal the truth and end the war could be photographed in the room numbered n.

Alpha Sections found out that somebody has penetrated into their laboratory, so the evidence will be destroyed in T seconds. Therefore, Jade must do her work in that time. The problem is that the passages between rooms are protected: any passage has its own security level s. The passage that has security level s can be gone through only if one has the security code of level more or equal to s.

Rebels can get any security code that Jade requests, but the higher-level security code costs more money. So Jade should determine what is the minimum level of security code she needs to get the evidence no later that T.

Input

First line contains two space-separated integers n and m ($2 \le n \le 1000$, $1 \le m \le 2000$) — the number of rooms and the number of passages between them. At the beginning Jade stands in the room numbered 1, and the room where she must get has number n.

Next m lines describe the passages. Each of these lines contains 4 integers separated by space: x_i, y_i, d_i, s_i ($1 \le x_i, y_i \le n$, $x_i \ne y_i$, $1 \le d_i \le 10^6$, $1 \le s_i \le 10^6$) — the numbers of rooms that are connected by the i-th passage, time needed to go through the passage, and its security level. Each passage is described exactly once. The passages are bidirectional, and it is possible to reach any room from any other room if one has the security code of maximum level.

The last line contains one integer T ($0 \le T \le 10^9$) — time at which Jade should get into room n.

Output

If Jade is not able to reach room n in time, even if she has the maximal security level, output «`NO`» (without quotes).

Otherwise in the first line output «`YES`», and in the second — two numbers separated by space: the minimal level of security code that makes it possible to get into room n in time no later than T, and the minimal time needed for it.

Examples

input.txt	output.txt
4 4 1 2 5 2 2 3 3 1 2 4 4 5 3 4 2 3 10	YES 3 10
4 4 1 2 5 2 2 3 3 1 2 4 4 5 3 4 3 3 10	YES 5 9
4 4 1 2 5 2 2 3 3 1 2 4 6 5 3 4 3 3 10	NO

Problem K. Triskaidekaphobia

Input file:	`input.txt`
Output file:	`output.txt`
Time limit:	1 second
Memory limit:	64 megabytes

Kostya is sick of *triskaidekaphobia*, i.e. he is afraid of this page's number. On Tuesday he was reading problem K again and again, and every time he disliked it more and more because this number appeared in its text too many times. Furthermore, it occured even in the MD5 hash of its title! It seemed that the problem was necessary to replace.

Finally, the stress took over him and he went to bed. But the fear didn't leave him: he dreamed that text of the problem consists only of the digits «`1`» and «`3`». Kostya was about to wake up, but his wish of sleep defeated his fear, so he decided to remove some digits from the text right in dream so that the damned number would not occur in it. Of course, he wants to do it as quickly as possible, and now he is wondering what minimal number of symbols can be deleted.

Input

The only line of the input contains the problem's text that came to Kostya in his dream. This text is not empty, its length does not exceed 1000 symbols, and its symbols are only digits «`1`» and «`3`».

Output

Print one integer — minimal number of symbols that should be removed from the text such that Kostya's unpleasant number will not occur in it as a substring.

Examples

input.txt	output.txt
313	1
1133	2
113133	3

Note

When Kostya woke up, he still decided to replace problem K. The result is before your eyes.

Problem L. Make Your Donation Now

Input file:	`input.txt`
Output file:	`output.txt`
Time limit:	1 second
Memory limit:	64 megabytes

Wimbo Jales decided to take donations from users of his little-known site. But the pitiful photo on the main page of the site is just not enough. For each of n site users he found out two values: a_i — the minimal amount of money the i-th user wants to donate and b_i — the maximal amount of money the i-th user can donate.

Wimbo Jales should set the minimal possible value of donation p. If $p < a_i$, the i-th user donates a_i, because he appreciates the project. If $p > b_i$ — he does not donate anything, because this sum of money is too great for him. At last, if $a_i \le p \le b_i$, he donates exactly p.

What value of p should Wimbo Jales set to get the maximal total sum of donations?

Input

The first line contains a single integer n $(1 \le n \le 10^5)$ — the number of site users.

Each of the following n lines contains two numbers separated by space: a_i and b_i $(1 \le a_i < b_i \le 10^9)$ — the minimal sum of money the i-th user wants to donate and the maximal sum of money the i-th user can donate.

Output

Output two integers separated by space — the minimal possible value of a donation which gives the maximal total sum of donations, and the value of this sum. If the maximal sum can be reached with more than one value of a donation, output the smallest of them.

Examples

input.txt	output.txt
1 5 10	10 10
2 5 10 20 25	10 30