Problem A. The Carpet

Input file:	`standard input`
Output file:	`standard output`
Time limit:	10 seconds
Memory limit:	128 mebibytes

Byteasar is visiting a carpet store looking at a certain carpet. Unfortunately, some fragments of the carpet are looking unsightly, due to factory defects. Because Byteasar would like to buy a substantial amount of the floor covering, he decided to allow himself to buy carpet with no more than one faulty spot. He will cover such a defect with a large flower-pot and the problem would be gone.

For simplicity, the carpet available in the store is represented as a rectangle with a height of h and width w, divided into $h \times w$ squares of size 1×1. We know whether each carpet square has a defect, or not. Byteasar would like to buy a rectangular carpet piece which is as large as possible, consisting of square units, wherein at most one square is faulty. What is the area of such a piece?

Input

The first line of input contains two integers h and w ($1 \le h, w \le 2000$), denoting respectively the height and width of the carpet available in the store. Subsequent h rows describe the carpet. Each of these lines contains a string of w characters `.` (square without faults), and `#` (faulty square), describing the individual carpet squares.

Output

Output the maximum area of a rectangular piece of the carpet, which consists of unit squares and has one defective square at most.

Examples

standard input	standard output

`4 5`
`#.#..`
`....#`
`..#..`
`....#`
	`12`

Problem B. The Construction

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	128 mebibytes

Byteasar would like to construct a racetrack in Byteotia. He competes with Bytemon, who would prefer building a ski jump, in gathering the funds necessary for the construction. Both projects are expensive, so Byteasar and Bytemon try to persuade The King of Byteotia for a donation.

The king must now choose one of the two options: either to provide funding for the racetrack or for the ski jump. For this purpose he would seek advice of the Chief Adviser, who is the head of the hierarchy of king's counsellors. Each counsellor is either an expert and provides his own recommendation, or leads a team of counsellors. Such a team leader recommends a decision, which is taken in accordance with the opinion of the majority of the members of his team. Fortunately, the number of counsellors in each team is odd. Therefore, the final decision depends only on the experts (i.e. counsellors not leading any of the teams). Each counsellor, with an exception of the Chief Adviser, has exactly one superior above him.

Byteasar and Bytemon are not waiting idly. They try to convince the experts to their case. It is not a simple task—persuading one expert takes exactly one day. When the expert is convinced there is no possibility he will change his opinion. It may happen that some of the experts at the very beginning already have a definite opinion that will not be changed.

Every day at dawn Byteasar selects one undecided expert and visits him in order to convince him. Bytemon does not like to get up so early, and so he selects an expert whom he wants to convince a bit later (and thus he loses a chance to convince the expert who already is busy, being persuaded by Byteasar). Rivals operate until all the experts would have a definite opinion. Byteasar and Bytemon know the hierarchy of the king's counsellors. Can Byteasar plan his lobbying activities in such a way that the Chief Advisor recommends the construction of the race track, regardless of how Bytemon would proceed?

Input

The first line of input contains one integer n ($2 \le n \le 1000$), denoting the number of counsellors. The counsellors are numbered from 1 do n. The counsellor numbered 1 is the Chief Advisor. In the i-th of n following lines there is a description of the i-th counsellor. It starts with an integer c_i ($-2 \le c_i \le n$). If $c_i \le 0$, the described counsellor is an expert (and his description consists only of number c_i). Values of -2, -1 and 0 indicate that he is, respectively, in favour of building the racetrack, in favour of building the ski jump, or undecided. When $c_i \ge 1$, the number c_i is odd, indicating that the advisor i leads a team of counsellors consisting of c_i members. Team member numbers are sequentially given in the following part of the line. Each counsellor with a number greater than 1 belongs to exactly one team.

Output

If Byteasar cannot convince the experts that the Chief Advisor recommends the construction of the racetrack, your program should print one output line containing the word “`NIE`” (Polish for *no*). Otherwise, your program should produce two lines. The first one should contain the word “`TAK`” (Polish for *yes*), followed by the integer d, which states in how many ways Byteasar can choose an expert to be convinced on the first day, in order to be certain that when making optimal decisions in the following days, a favourable recommendation of the Chief Adviser would be obtained. The second line should contain a sequence of d numbers of these very experts, in the *increasing* order. If, at the beginning, all the experts have definite opinions (and the recommendation of the Chief Adviser is beneficial to Byteasar), the program should print $d = 0$.

Examples

standard input	standard output

4
3 2 3 4
-2
0
-1
	TAK 1
	3

Problem C. Game of Doubling

Input file:	`standard input`
Output file:	`standard output`
Time limit:	20 seconds
Memory limit:	128 mebibytes

Game of doubling is actually more a puzzle than a game. The game board is a rectangle divided into fields that are unit squares. At the beginning, some fields contain a token, and some do not.

The player's objective is to gather a highest number of tokens in a single field. The only possible move is to locate two fields adjoining sides that contain the same (positive) number of tokens and transfer all tokens from one of these fields onto the other.

Write a program that for a given initial board configuration, would determine for each field the maximum number of tokens a player could accumulate in this field.

Input

The first line contains two integers n and m ($1 \leq n, m \leq 200$) indicating the number of rows and number of columns of the board. Each of the following n lines contains a sequence of m digits: 0 or 1. The digit 1 indicates a field containing a token, and the digit 0 indicates an empty field.

Output

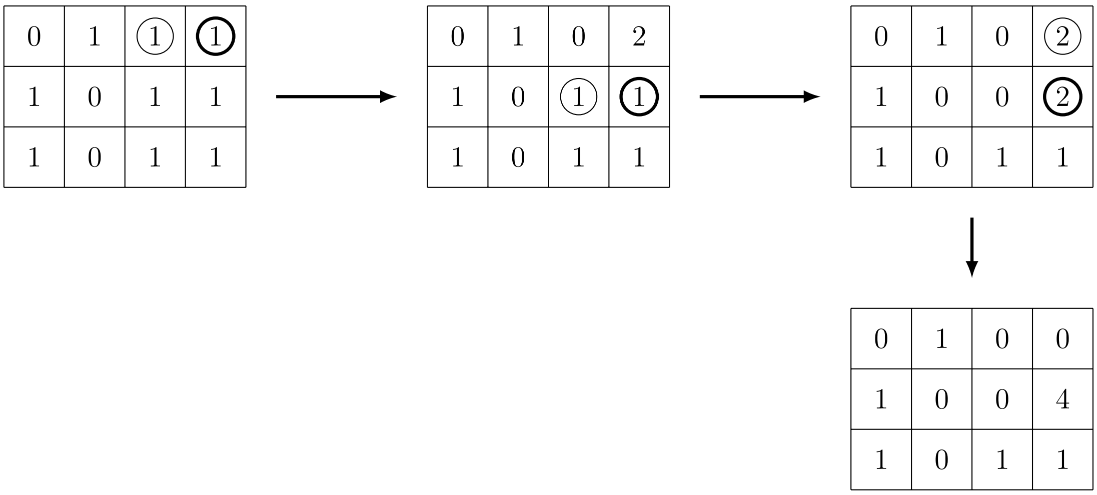
Your program must output n lines each containing m integers. The j-th number in the i-th row should indicate the maximum number of tokens that a player could accumulate on the field located at the intersection i-th row and j-th column, starting from the given initial configuration.

Examples

standard input	standard output

3 4
0111
1011
1011
	0 2 4 4
	2 0 4 4
	2 0 4 4

Note



The above example explains how to collect 4 tokens at a field positioned at the intersection of the middle row and the last column.

Problem D. The Kingdom

Input file:	`standard input`
Output file:	`standard output`
Time limit:	3 seconds
Memory limit:	128 mebibytes

In Bytetoria there are n cities and even number of bidirectional roads connecting the cities. The road network allows commuting between any two cities of the kingdom.

King Bythor, the ruler of Bytetoria, is known for his passion for even numbers. When he realized that there are cities in his kingdom, which are a starting point of an odd number of roads leading out, he immediately demanded the expansion of the road network.

King's Adviser knows the Bytetoria finances well, and knows that the implementation of such a serious investment would make organization of the Winter Olympic Games impossible, very much expected by the Bytetorian people. He plans to convince the King that Bytetoria has enough "even" values in order to ask for a postponement of investment decision until the next year.

Firstly, the advisor would surprise the king by the fact that there is an even number of cities in Bytetoria with an odd number of outgoing roads. Then he plans to divide these cities into pairs, and for each of such created pairs (u, v) assign the route from u to v, consisting of an even number of roads. To further impress the king, no road would appear more than once on the way from u to v. Furthermore, no Bytetorian road will be included in more than one of the designated routes.

The advisor is certain that such argumentation will convince the king. However, he is unable to cope with the actual routes design, so he asked you for help.

Input

The first line of input contains two integers n and m ($2 \leq n, m \leq 250\,000$). They denote the number of cities and the number of roads in Byteoria, respectively. m is an even number.

Each of the subsequent m lines contains two integers a, b ($1 \leq a, b \leq n$, $a \neq b$) indicating that the cities a and b are connected by a bidirectional road. Between any two cities there exists not more than one road.

It can be assumed that there exists a city, which is a staring point to an odd number of roads.

Output

By k we denote the number of cities which are a staring point to the odd number of roads (the advisor is certain that k is an even number).

If it is not possible to determine routes according to adviser's plan, the only line of output should contain the word `NIE` (Polish for *no*).

Otherwise, the program should output $\frac{k}{2}$ descriptions of designated routes. Each of the route's descriptions should consist of two lines.

The first line of i-th description should contain three integers u_i, v_i, l_i ($2 \mid l_i$) indicating that the route begins at city u_i, ends at city v_i and consists of l_i roads.

The second line of i-th description should contain l_i integers denoting the numbers of the consecutive roads on the route.

The roads are numbered from 1 to m, according to the order in which they were given at input.

In case there are multiple solutions, your program can output any one of them.

Examples

standard input	standard output

6 8
1 2
2 3
3 4
4 5
5 6
6 1
1 4
2 5
	1 5 6
	1 2 3 7 6 5
	2 4 2
	8 4

Problem E. Light Bulbs

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	64 mebibytes

Finishing jobs at Byteasar's new home are ending shortly. All that remains is to install new light bulbs in each of the n rooms. The minimum bulb power rating has been determined for each of the rooms, to provide sufficient illumination.

Byteasar bought n light bulbs, but now he noticed that they do not meet fully his expectations. It may not be possible to provide all rooms with an adequate illumination, and it is also possible that maybe some bulbs unnecessarily have too high power rating. Byteasar then decided to go to the store and replace some of the bulbs, so to be able to sufficiently illuminate all the rooms, and at the same time minimize the total power of the light bulbs. The shop has light bulbs available in all power ratings expressed by a positive number. Byteasar's backpack capacity allows him to to take no more than k bulbs to be replaced, which is therefore the maximum number of light bulbs that he is ready to replace.

Help Byteasar to choose which bulbs should be replaced, so that all the rooms are adequately lit, and at the same time to minimize the total power of light bulbs.

Input

The first line of the input contains two integers n and k ($1 \le k \le n \le 500\,000$), indicating the number of rooms (this is also the number of light bulbs) and the number of light bulbs that can be fitted in Byteasar's backpack. The rooms are numbered from 1 to n. The second line of the input contains n integers $p_1, p_2, \ldots, p_n$ ($1 \le p_i \le 10^9$) denoting power ratings of the light bulbs which currently are at Byteasar's disposal. The third line of input contains n integers $w_1, w_2, \ldots, w_n$ ($1 \le w_i \le 10^9$) denoting the illumination requirements concerning the subsequent rooms: the room i should be fitted with a bulb with bearing a minimal rating of w_i.

Output

In case it is not possible to replace at most k bulbs in order to have all the rooms sufficiently illuminated, your program should output `NIE` (Polish for *no*). Otherwise, an integer should be written indicating the minimum total power of all light bulbs used to illuminate the home after exchanging at most k bulbs.

Examples

standard input	standard output

6 2
12 1 7 5 2 10
1 4 11 4 7 5
	33

Note

It is enough to replace the bulb with a power rating of 2 with a bulb bearing a power rating of 4, and the bulb with a power rating of 10 with one rated 4. Then almost all the rooms will have a light bulb installed of exactly such power rating as being a minimum requirement. The exception would be a room that would have been illuminated adequately with the bulb with the power rating of 11, which will be illuminated by a bulb rated 12.

Problem F. Little Marketplace

Input file:	`standard input`
Output file:	`standard output`
Time limit:	20 seconds
Memory limit:	128 mebibytes

Little Bytec spends his holidays at Grandma Bytegranny. Every morning Grandma goes to the little marketplace to buy various products. The boy quickly noticed an interesting regularity: every day his grandmother spends an amount expressed by an odd integer on her shopping. Bytec soon found out that this regularity is characteristic to all Byteonian grandmothers.

Every day Bytegranny buys not more than one item of each of the n products available in the market. Thrifty grandma does not want to do her shopping carrying too much money with her. One day she asked Bytec for a hint of how much money she needs to take, in case she wants to purchase exactly k products. Unfortunately, Bytec does not know which products grandmother wants to buy, so the amount taken must be sufficient for any k items (and that their total cost would be an odd number). The same situation repeated itself several times. Bytec then decided to approach the problem methodically and write a program that, provided all the prices of products available on the market, will answer grandmother's questions.

Input

The first line of input contains one integer n ($1 \leq n \leq 1\,000\,000$) denoting the number of products available at the market. The second line contains n integers from the range $[1, 10^9]$, denoting the prices of individual products. The third line contains one integer m ($1 \leq m \leq 1\,000\,000$) denoting the remaining number of days that Bytec will stay at Grandma's place. Each of the following m lines contains one integer k_i ($1 \leq k_i \leq n$), denoting the number of products that grandma is going to buy on a given day.

Output

Your program should output m lines. In the i-th line (for $i = 1, \ldots, m$) one integer should be written, indicating the maximum odd price total for k_i products. In case it is not possible to determine k_i products, such that their total price would be represented by an odd number, the i-th output line should contain the number -1.

Examples

standard input	standard output

4
4 2 1 3
3
2
3
4
	7
	9
	-1

Problem G. The Matrix

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	128 mebibytes

Byteotian Printing Factory (BPF) has received a large production order for striped wallpaper. Striped wallpaper is the hit of the season in interior design. Each wallpaper has n equal-width coloured vertical stripes. BPF is to take care of the design and printing of wallpapers. The customer described the colour of certain wallpaper stripes. In the case of other stripe colours, the customer allowed BPF a complete freedom.

In the production of wallpapers BPF use matrices to print a certain number of consecutive stripes on the wallpaper. The matrix has a specific colour of each of the printed stripes. The matrix may be shorter than the entire wallpaper. If the matrix consists of k stripes it is applied in *all* $n - k + 1$ possible positions where its stripes align with the wallpaper stripes, each time printing all the matrix stripes. In this way, a single wallpaper stripe can be printed over more than once. In case a given stripe is printed over using various colours, the final colour is a blend of these colours.

BPF employees, irrespective of their sense of aesthetics, would primarily like to design the shortest possible matrix that will allow printing the entire wallpaper. They must bear in mind that in the case of stripes defined by the customer they must use pure colour, without any addition of any other colour. In other words, each matrix application printing over such a single colour stripe, the matrix stripe colour must be exactly as defined by the client.

Input

The only input line contains a string composed of Latin alphabet upper case letters and stars (‘`*`)’, specifying the desired appearance of the wallpaper. The letters represent different colour stripes, while the stars indicate stripes of the colour not specified by the client. The length of the character string n satisfies $1 \leq n \leq 1\,000\,000$.

Output

Your program should output a single line containing a single integer k: the minimum matrix length, which would enable printing the desired wallpaper.

Examples

standard input	standard output
`A*B*B*A`	`6`

Note

The matrix of length 6 which enables printing the wallpaper presented at the input (composed of seven stripes) is “`ABBBBA`”.

Problem H. The Museum

Input file:	`standard input`
Output file:	`standard output`
Time limit:	4 seconds
Memory limit:	128 mebibytes

Bytemon, a well known burglar, wants to rob the National Museum of Byteotia. He particularly is interested in the Royal Family Jewels, which are displayed in the most magnificent hall of the museum. There are n exhibits watched over by m guards in this hall. The museum's custodian wanted to ensure that the visitors, admiring exhibits, are not being disturbed by the guards more than necessary. Therefore, he ordered them to keep their set positions all the time, and look in one direction only.

Bytemon managed to get plan of this hall, where all the exhibits have been marked, as well as the arrangement of the guards. He obtained a quotation on all displayed jewels from a jeweller he knew. He also learned how much it would cost to discretely persuade each guard to close his eyes to Bytemon's activities at the time of the burglary.

Bytemon is wondering now how rich he can get. Therefore, he wants to chose the guards to be bribed, in such a way that the total value of the gems that are not in sight of any of guards that have not been bribed, less the cost of bribing selected guards, is as large as possible.

Input

The first line of input contains two integers n and m ($1 \leq n, m \leq 200\,000$), describing the number of exhibits and the number of guards. To describe their positioning assume that the museum plan has a rectangular coordinate system imposed. The second line of the input contains two integers w and h ($1 \leq w, h \leq 10^9$), describing the field of vision of the guards. Each of the guards is looking in the direction of decreasing y-coordinates, and the tangent of half of the angle of his view amounts to w/h. For simplicity, we assume that the guards and the exhibits are of a negligible size. The guard is observing all the exhibits, which are located in his field of vision (including the edge), even in case they are occluded by other exhibits or guards.

Subsequent n lines describe the position of the exhibits; i-th of these lines contains three integers x_i, y_i, v_i ($-10^9 \leq x_i, y_i \leq 10^9$, $1 \leq v_i \leq 10^9$) indicating that the i-th exhibit has a value of v_i bytecoins and is located at the point (x_i, y_i). Subsequent m lines describe the guard positions, analogically. (However, v_i denotes the amount, in bytecoins, to be paid by Bytemon to bribe i-th guard.) At each point there can be at most one guard or exhibit.

Output

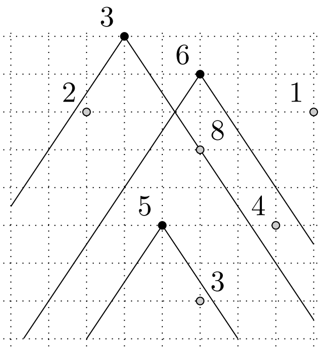
Your program should output a single line containing a single integer indicating the maximum profit, in bytecoins, that could be achieved by Bytemon.

Examples

standard input	standard output

5 3
2 3
2 6 2
5 1 3
5 5 8
7 3 4
8 6 1
3 8 3
4 3 5
5 7 6
	6

Note



Angle of vision of each of the guards slightly exceeds 67°. Bytemon should bribe two guards, paying $3+6$ bytecoins, and take exhibits of a value of $2+8+4+1$ bytecoins.

Problem I. Tape

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	64 mebibytes

Bytie's dad works as a tailor. One day, when dad left Bytie alone in his workshop, Bytie found a very nice tape. It actually was a tape measure containing consecutive numbers from 1 to n. Somehow Bytie also managed to find his dad's scissors...

Unfortunately, when dad came, it was already too late: Bytie already cut the tape a few times. What is fortunate, in each of the cuts Bytie cut out a fragment of the tape, reversed it and put it back exactly at the same position. Bytie claims that he performed exactly k such cuts. For example, if the initial tape was [1, 2, 3, 4, 5, 6] and Bytie cut out the fragment [3, 4, 5] and reversed it, the tape would be [1, 2, 5, 4, 3, 6].

Help Bytie's dad and try to restore the initial tape using k operations of cutting and reversing fragments.

Input

The first line of input contains two integers n and k ($1 \le n \le 100\,000$, $0 \le k \le 3$) that represent the number of integers written on the tape and the number of operations that Bytie claims to have performed. The next line contains the numbers written on the tape when Bytie's dad entered the room: a sequence of n integers a_i ($1 \le a_i \le n$).

Output

The first line of output should contain one word "TAK" (Polish for *yes*) or "NIE" (Polish for *no*), depending on whether one can restore the initial state of the tape using k cuts. If the answer is positive, the following k lines of output should contain two integers l_i, r_i each ($1 \le l_i \le r_i \le n$): the indices of the first and last element of consecutive fragments that are to be reversed. If there is more than one correct answer, your program should output any none of them.

Examples

standard input	standard output
4 2 3 4 1 2	TAK 1 3 2 4
4 1 3 4 1 2	NIE

Note

After Bytie played with the tape, the sequence of numbers is [3, 4, 1, 2]. In the first example we can restore the initial state of the tape in two operations:

1. Cut and reverse the fragment from the first to the third tape element, the result is: [1, 4, 3, 2].
2. Cut and reverse the fragment from the second to the fourth tape element, the result is: [1, 2, 3, 4].

One cannot restore the initial state of the tape using just a single operation.

Problem J. The Task

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	128 mebibytes

Byteasar is busy preparing a task for a programming contest. Has has already written a sketch of the content:

> *There are n cities in Byteotia, connected by $n-1$ bidirectional roads in such a way that using the road network it is possible to drive between any two of the cities. It takes one hour to cover the distance between two directly connected cities. Cities are numbered from 1 to n. The number of inhabitants living in the city i is equal to a_i.*
>
> *Next year elections are to be held in Byteotia. In order to have a full control over the conduct of the voting process, the King of Byteotia decided that the vote will be held in one city only. All Byteotian residents are to go to the city, where the ballot will be held, using the the shortest way, and they would vote there. Now all that remains is to choose the city where the voting will take place. This choice depends on many factors. In particular, for each city i, we would like to calculate the total time required by all Byteotian residents to reach the city i (we indicate this value by b_i).*

Byteasar had already prepared an extremely difficult set of tests to the task, but accidentally lost half of the data. Now, all that remained out of each test were only the descriptions of road connections and output files containing the b_i values. On this basis, he would like to restore the data concerning the population of each Byteotian city.

Input

The first line of input contains an integer n ($2 \le n \le 300\,000$) denoting the number of Byteotian cities. Each of the following $n-1$ lines contains a description of one road connection in the form of a pair of integers x_i, y_i ($1 \le x_i, y_i \le n$). They indicate that the cities x_i and y_i have a road connection. The subsequent line contains a sequence of n integers b_i ($0 \le b_i \le 10^9$).

Output

Produce a single line containing a sequence of n integers a_i. The number a_i should indicate the population of Byteotian city number i. After solving Byteasar's exam task, for a given string a_i we should receive string b_i provided at the Input.

The input data are selected in such a way that a solution always exists and is unique.

Examples

standard input	standard output

2
1 2
17 31
	31 17

Problem K. Teams

Input file:	*standard input*
Output file:	*standard output*
Time limit:	2 seconds
Memory limit:	128 mebibytes

Mr Byteotish is the most popular P.E. teacher at No. 64 Primary School named after Byteasar, the Travelling Salesman of Byteotia. During each lesson, after a brief warm-up, he ask students to choose the team game they would like to play, and then helps them to get divided into teams.

During the line-up the students number themselves up using consecutive numbers from 1 to n. Mr Byteotish creates the teams in such a way that each of them is formed from students with consecutive numbers. Each student must belong to one of the teams.

The teacher knows his students well and knows that the student number i will be satisfied with his assignation only in case the number of the players in his team will not be less than c_i and not exceed d_i.

Mr Byteotish wonders whether it is possible to divide students into teams in such a way, that every student would be satisfied. In case it is possible, he would like to know the maximum possible number of teams that could be formed, as well as the number of divisions that are pursuing this maximum.

Input

The first line of input contains one integer n $1 \le n \le 1\,000\,000$), denoting the number of students. The following n lines describe the preferences of students: i-th of these lines contains two integers c_i, d_i ($1 \le c_i \le d_i \le n$), indicating that the student bearing number i is satisfied, providing the number of players in his team will belong to the interval $[c_i, d_i]$.

Output

In case it is possible to divide students according to Mr Byteotish's procedure in such a way, that each of them would be satisfied, the output should contain two integers separated by a single space—the maximum number of teams and number of divisions pursuing this maximum. The second of these numbers should be produced modulo $10^9 + 7$.

If students cannot be divided in accordance with the above requirements, output should contain only the word NIE (Polish for *no*).

Examples

standard input	standard output

9
1 4
2 5
3 4
1 5
1 1
2 5
3 5
1 3
1 1
	5 2

2
1 1
2 2
	NIE