Information

Memory limit

The limit is 512 MiB for each problem.

Source code limit

The size of each solution source code can't exceed 256 KiB.

Submissions limit

You can submit at most 50 solutions for each problem.

You can submit a solution to each task at most once per 30 seconds. This restriction does not apply in the last 15 minutes of the contest round.

Scoring

Each problem consists of several subtasks. The subtask score is awarded if all tests in the subtask are passed.

The number of points scored for the problem is the total number of points scored on each of its subtasks. The score for the subtask is the maximum number of points earned for this subtask among all the solutions submitted.

Feedback

To get feedback for your solution, go to "Runs" tab in PCMS2 Web Client and use "View Feedback" link. In each problem of the contest you will see the score for each subtask, or the verdict for the first failed test.

Scoreboard

The contestants' scoreboard is available during the contest. Use "Monitor" link in PCMS2 Web Client to access the scoreboard. The standings provided in PCMS2 Web Client are not final.

Problem A. Big Money

Time limit: 1 second

Peter won a lot of money in the lottery — a huge amount of m rubles! Peter is smart, so instead of wasting his winnings he decided to make a deposit in the bank. But making one deposit can be risky, so he decided to make two instead.

The bank asks Peter to choose one of n offers. Each offer consists of two deposits, each of which is characterized by three numbers l, r and p — making a deposit of not less than l and not more than r rubles, Peter will receive back the invested money plus an additional p percent of them upon the expiration of the deposit term. In each offer you can use either of the two deposits or even use none of them, but if you make two deposits at once you need to invest money to both of them at the same time and profit from the deposits will be received independently of each other. Different offers cannot be combined.

Peter is already dreaming of making his wishes come true, so he turned to you for help. For each offer calculate how much money Peter may have if he uses the deposits offered.

Input

The first two lines contain integers m and n ($1 \le m \le 2 \cdot 10^9$, $1 \le n \le 10^5$) — Peter's money and the number of bank offers. The following n lines contain six integers l_1, r_1, p_1, l_2, r_2, p_2 ($1 \le l_i \le r_i \le 2 \cdot 10^9$, $1 \le p_i \le 200$) — description of the offered deposits.

Output

For each offer print in a separate line one number — the maximum amount of money that Peter may have. The answer will be considered correct if its absolute or relative error does not exceed 10^{-9}. Namely: let's assume that your answer is a, and the answer of the jury is b. The checker program will consider your answer correct, if $\frac{|a-b|}{\max(1,a)} \le 10^{-9}$.

Scoring

Subtask	Score	Constraints	
		$\sum r$	Additional
1	10	$\sum r \le 4000$	—
2	15	$\sum r \le 4 \cdot 10^6$	—
3	20	$\sum r \le 2 \cdot 10^9$	$l = 1$
4	20	$\sum r \le 2 \cdot 10^9$	$l = r$
5	35	$\sum r \le 2 \cdot 10^9$	—

Example

standard input	standard output
100 3 1 50 5 1 51 10 70 200 100 30 95 110 179 239 40 109 140 31	107.550000000 204.500000000 100.000000000

Note

For the first offer, it is optimal to put 49 rubles in the first deposit and 51 in the second deposit, then Peter will receive $0.05 \cdot 49 + 0.1 \cdot 51 = 7.55$ rubles of profit. For the second offer, it is optimal to put 95 rubles in the second deposit, and not to use the first deposit at all, then Peter will receive $1.1 \cdot 95 = 104.5$ rubles of profit. For the third offer, Peter cannot use any of the deposits.

Problem B. Cake Tasting

Time limit: 1 second

Karina really loves cakes. This month she visited n different pastry shops, where she participated in a cake tasting. For each pastry shop, Karina recorded all types of cakes she tasted using a special smartphone app. However, after all tasting events, she noted that the app can only show a number of different types of cakes she tasted. And this information was counted for all $2^n - 1$ subsets of pastry shops. Karina is upset and thinks that even this information was counted incorrectly.

Help Karina understand whether this information is correct and if so find any possible Karina's record.

Input

The first line contains single integer n ($1 \le n \le 19$) — the number of pastry shops. The next line contains $2^n - 1$ integers a_i ($1 \le i \le 2^n - 1$; $1 \le a_i \le 1000$). The integer a_i equals to the number of different types of cakes Karina tasted in shops j, where j-th digit of binary representation of i is equal to one. For example, if S_k is a set of cake types in k-th pastry shop, then $a_1 = |S_1|$, $a_2 = |S_2|$, $a_3 = |S_1 \cup S_2|$, $a_4 = |S_3|$, and so on.

Output

In the first line print "Yes", if the information is correct, or "No", otherwise. If the information is correct, print a possible Karina's record in each of the following n lines. For each pastry shop, the record should begin with integer k_i ($1 \le k_i \le 1000$) equals the number of cake types in i-th pastry shop. Then, in the same line print k_i different integers $s_{i,j}$, that denote the types of cakes, $s_{i,j}$ must not exceed 10^9 by absolute value.

If there are several possible options of Karina's records, print any of them. It is guaranteed that if the solution exists, there is a solution with $k_i \le 1000$ for all $1 \le i \le n$.

Scoring

Subtask	Score	Constraints	
		n	a_i
1	10	$n \le 2$	$a_i \le 10$
2	15	$n \le 4$	$a_i \le 10$
3	30	$n \le 8$	$a_i \le 1000$
4	26	$n \le 15$	$a_i \le 1000$
5	19	$n \le 19$	$a_i \le 1000$

Examples

standard input	standard output
2 2 3 4	Yes 2 1 4 3 1 2 3
2 2 2 5	No
3 3 2 4 3 4 4 5	Yes 3 1 2 5 2 1 4 3 1 2 3

Note

In the first example there are two pastry shops. In the first pastry shop Karina tasted 2 types of cakes, in the second — 3 types, in the first and second together — 4 types. One of the possible ways corresponding to these numbers is the following. In the first pastry shop Karina tasted cakes of 1 and 4 types, in the second first pastry shop she tasted cakes of 1, 2 and 3 types, then in both pastry shops together she tried four types of cakes: 1, 2, 3 and 4.

In the second example, Karina tried 2 types of cakes in the first and second pastry shops, and 5 types in total. This obviously could not be true.

In the third example, there are three pastry shops. One of possible answers is:

$a_1 = |S_1| = |\{1, 2, 5\}|$,

$a_2 = |S_2| = |\{1, 4\}|$,

$a_3 = |S_1 \cup S_2| = |\{1, 2, 4, 5\}|$,

$a_4 = |S_3| = |\{1, 2, 3\}|$,

$a_5 = |S_1 \cup S_3| = |\{1, 2, 3, 5\}|$,

$a_6 = |S_2 \cup S_3| = |\{1, 2, 3, 4\}|$,

$a_7 = |S_1 \cup S_2 \cup S_3| = |\{1, 2, 3, 4, 5\}|$.

Problem C. Well, Just You Wait!

Time limit: 2 seconds

Vasya, the actor who played the Wolf in russian animated series «Well, Just You Wait!», took a part in scenes of chasing Petya who played the Hare in this series. Nobody ever knew that Vasya actually wants to eat Petya!

The film set is a convex polygon of n vertices on a plane. Petya is a great nature lover. Each of the next m days he will plant a tree right in a film set. Vasya wants to hide somewhere on the film set. In the morning Petya will come to plant a tree, Vasya will jump out of his cover and eat Petya.

Film set is full of surprises. At any moment workers can build a new wall. Wall is a segment that connects two vertices of the polygon. Vasya doesn't want to fail, so he decided to choose a place for a hiding such that it is impossible to install a wall that separates Vasya and Petya (If a wall comes right through Vasya's hiding place, then he can move to the part of the polygon where Petya is located).

Vasya doesn't want Petya to detect him before the attack. Therefore Vasya want to choose a place to hide with maximum possible distance from the place where Petya will plant a tree.

Your task is to help Vasya: for each of the next m days calculate the maximum possible distance from a tree to a possible place to hide.

Input

The first input line contains the only integer n — number of vertices of polygon that represents the film set ($3 \le n \le 2\,000$).

The next n lines contain pairs of integers x_i, y_i — coordinates of points that represent vertices of the polygon ($-200\,000 \le x_i, y_i \le 200\,000$), given in counterclockwise order. It is guaranteed that polygon is convex and there is no three vertices on the same line.

The next line contains single integer m — number of days ($1 \le m \le 1\,000$).

Each of the next m lines contains pairs of integers u_i, v_i — coordinates of points where Petya wants to plant a tree on the i-th day ($-200\,000 \le u_i, v_i \le 200\,000$). It is guaranteed that all of these points lie strictly inside of the polygon and do not lie on a line that connects two vertices of the polygon.

Output

Print m lines.

In the i-th line print single number — the maximum distance that Vasya can hide from location of the tree that Petya will plant on day i. Your answer will be considered correct if it's absolute or relative error related to correct answer is lower than 10^{-6}. Namely: let's assume that your answer is a, and the answer of the jury is b. The checker program will consider your answer correct, if $\frac{|a-b|}{\max(1,a)} \le 10^{-6}$.

Scoring

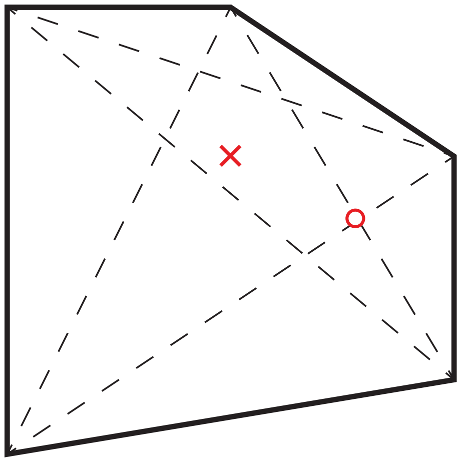
Subtasks	Scores	Constraints		
		n	m	x_i, y_i, u_i, v_i
1	6	$3 \le n \le 4$	$1 \le m \le 5$	$\|x_i\|, \|y_i\|, \|u_i\|, \|v_i\| \le 50$
2	13	$3 \le n \le 15$	$1 \le m \le 5$	$\|x_i\|, \|y_i\|, \|u_i\|, \|v_i\| \le 100$
3	12	$3 \le n \le 70$	$1 \le m \le 70$	$\|x_i\|, \|y_i\|, \|u_i\|, \|v_i\| \le 1000$
4	30	$3 \le n \le 200$	$1 \le m \le 200$	$\|x_i\|, \|y_i\|, \|u_i\|, \|v_i\| \le 10\,000$
5	39	$3 \le n \le 2\,000$	$1 \le m \le 1\,000$	$\|x_i\|, \|y_i\|, \|u_i\|, \|v_i\| \le 200\,000$

Examples

standard input	standard output
5 5 -2 5 1 2 3 -1 3 -1 -3 1 2 1	1.9166296949998198
3 3 1 10 3 5 7 3 5 2 9 3 6 6	5.0990195135927845 6.324555320336759 5.830951894845301

Note

There is an illustration of the first example below. Dashed lines represent possible locations of wall. Cross represents location of a tree and circle represents an optimal Vasya's position.



Problem D. Super Non-massive Black Hole

Time limit: 5 seconds

You are working on the study of atomic black holes. The study area is a three-dimensional space. In this space there are n integer points in which atomic black holes can arise, we call them *observation points*. Also, the installation contains m sensors. Sensors are fixed and cannot be moved relative to each other. However, all sensors together can be moved along the vector $(1, 1, 1)$. That is, if initially the i-th sensor has coordinates (u_i, v_i, w_i), after moving i-th sensor will have coordinates $(u_i + d, v_i + d, w_i + d)$, where d is an arbitrary integer that is the same for all sensors.

A sensor located at the point (u, v, w) reacts to a black hole at the point (x, y, z) if $x \le u$, $y \le v$ and $z \le w$, we say that in this case this point is in the *visibility range* of this sensor. It is necessary that each observation point is in the visibility range of at least one sensor. At the same time, in order to increase the sensitivity of the sensors, they need to be located as close as possible to the observation points. That is, it is required to choose the minimum value of d such that if all sensors are moved by the vector (d, d, d), each observation point will be in the visibility range of at least one sensor.

An arbitrary number of observation points and sensors can be located at the same point.

Input

The first line contains two integers n and m, the number of observation points and the number of sensors, respectively $(1 \le n, m \le 500\,000)$.

The next n lines contain the coordinates of the observation points. Each line contains three integers x_i, y_i, and z_i, the coordinates of the i-th observation point $(-10^{18} \le x_i, y_i, z_i \le 10^{18})$.

The next m lines contain the initial coordinates of the sensors. Each line contains three integers u_i, v_i, and w_i, the initial coordinates of the i-th sensor $(-10^{18} \le u_i, v_i, w_i \le 10^{18})$.

Output

Output one integer, the minimum value of d such that if all sensors are moved by the vector (d, d, d), each observation point will be in the visibility range of at least one sensor.

Scoring

Subtask	Score	Constraints	
		n, m	x_i, y_i, z_i, u_i, v_i, w_i
1	11	$1 \le n, m \le 5\,000$	$\lvert x_i\rvert, \lvert y_i\rvert, \lvert z_i\rvert, \lvert u_i\rvert, \lvert v_i\rvert, \lvert w_i\rvert \le 5\,000$
2	21	$1 \le n, m \le 100\,000$	$\lvert x_i\rvert, \lvert y_i\rvert, \lvert z_i\rvert, \lvert u_i\rvert, \lvert v_i\rvert, \lvert w_i\rvert \le 100\,000$
3	20	$1 \le n, m \le 200\,000$	$\lvert x_i\rvert, \lvert y_i\rvert, \lvert z_i\rvert, \lvert u_i\rvert, \lvert v_i\rvert, \lvert w_i\rvert \le 10^9$
4	20	$1 \le n, m \le 300\,000$	$\lvert x_i\rvert, \lvert y_i\rvert, \lvert z_i\rvert, \lvert u_i\rvert, \lvert v_i\rvert, \lvert w_i\rvert \le 10^9$
5	28	$1 \le n, m \le 500\,000$	$\lvert x_i\rvert, \lvert y_i\rvert, \lvert z_i\rvert, \lvert u_i\rvert, \lvert v_i\rvert, \lvert w_i\rvert \le 10^{18}$

Examples

standard input	standard output
3 2 4 5 6 5 6 4 6 4 5 1 2 3 3 2 1	4
1 1 3 2 1 4 5 6	-1

Note

In the first example there are three observation points with coordinates $(4, 5, 6)$, $(5, 6, 4)$, and $(6, 4, 5)$, and two sensors with coordinates $(1, 2, 3)$ and $(3, 2, 1)$. If all the sensors are shifted by the vector $(4, 4, 4)$, then they move to the points $(5, 6, 7)$ and $(7, 6, 5)$, respectively. The first and second observation points are in the visibility range of the first sensor, and the second and third observation points are in the visibility range of the second sensor.

Problem E. Yet Another Tree Problem

Time limit: 5 seconds

Given a tree with n vertices (a tree is an undirected connected simple graph without cycles). Each edge has an integer weight. Some vertices of the tree are marked. You need to implement a program that performs the following operations on this tree:

1. Mark the vertex. It is guaranteed that before this operation it was not marked.
2. Unmark the vertex. It is guaranteed that before this operation it was marked.
3. Change the weight of the edge.

Before performing all operations and after each operation, your program should output the maximum number x such that there is a simple path with **at least two** marked vertices on it, with total weight equal to x. If there are no two marked vertices in the tree, print «BAD».

Input

The first line contains two integers n and q ($1 \le n, q \le 150\,000$), the number of vertices and the number of operations, respectively.

The next line contains n numbers, describing whether the vertex is marked in the beginning: 1 means that the vertex is marked, 0 means that that the vertex is not marked.

The next $n-1$ lines describe the edges of the tree. For all i from 2 to n, there are two numbers p_i ($1 \le p_i < i$) and w_i ($-10^9 \le w_i \le 10^9$), which means that there is an edge in the tree between the vertices i and p_i of weight w_i.

The next q lines describe the operations. Each operation has one of the following forms:

- 1 x ($1 \le x \le n$): mark vertex x,
- 2 x ($1 \le x \le n$): unmark vertex x,
- 3 x w ($2 \le x \le n$, $-10^9 \le w \le 10^9$): make the weight of the edge between the vertices of p_x and x equal to w.

Output

Output $q+1$ lines. In the first line print the answer before performing the operations. On each next line print the answer after performing the next operation.

Scoring

By D we denote the maximum number of edges connected to a single vertex, and by H the maximum number of vertices on a simple path from vertex 1 to some other vertex.

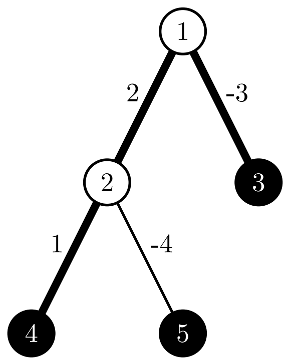
Subtask	Score	Constraints		
		n	q	Additional
1	3	$n \le 20$	$q \le 20$	—
2	3	$n \le 500$	$q \le 500$	—
3	3	$n \le 100$	$q \le 5000$	—
4	5	$n \le 5000$	$q \le 5000$	—
5	7	$n \le 10^5$	$q \le 10^5$	$D \le 30$, $H \le 20$
6	20	$n \le 10^5$	$q \le 10^5$	$H \le 20$
7	10	$n \le 10^5$	$q \le 10^5$	$p_v = v - 1$ for all v
8	17	$n \le 10^5$	$q \le 10^5$	$w_i \le 0$, $w \le 0$ for all operations
9	22	$n \le 50\,000$	$q \le 50\,000$	—
10	5	$n \le 10^5$	$q \le 10^5$	—
11	5	$n \le 150\,000$	$q \le 150\,000$	—

Example

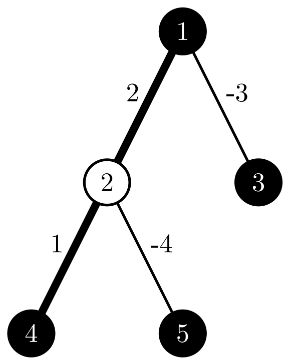
standard input	standard output
5 7 0 0 1 1 1 1 2 1 -3 2 1 2 -4 1 1 2 4 3 3 5 2 3 2 1 3 5 -1 1 1	0 3 0 8 3 BAD BAD 6

Note

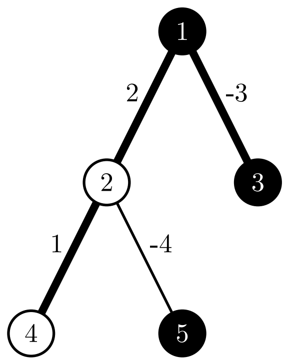
Pictures for the example. The bold path is the path of weight x with at least two marked vertices.



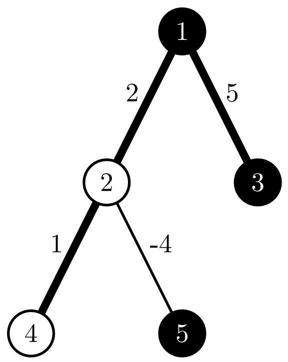
before operations



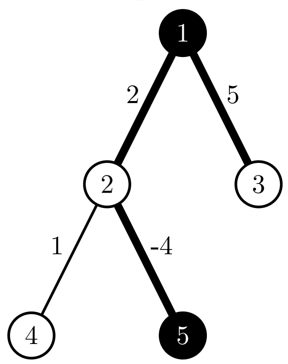
after operation 1



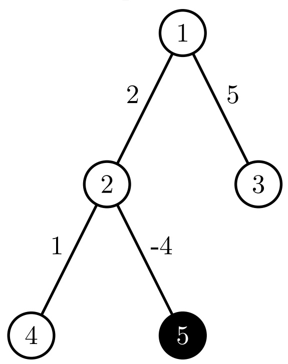
after operation 2



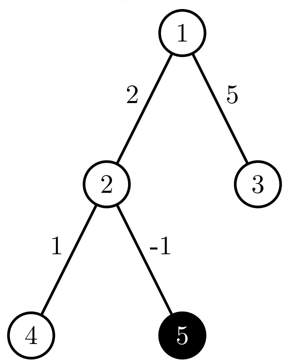
after operation 3



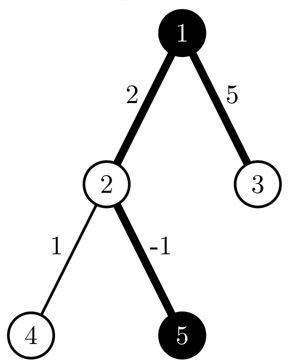
after operation 4



after operation 5



after operation 6



after operation 7