The 45th International Collegiate Programming Contest
Asia Macau Regional Contest

May 29

澳門大學
UNIVERSIDADE DE MACAU
UNIVERSITY OF MACAU

科技學院
Faculdade de Ciências e Tecnologia
Faculty of Science and Technology

香港城市大學
City University of Hong Kong

Problems

- A Accelerator
- B Boring Problem
- C Club Assignment
- D Artifacts
- E Mountain
- F Fixing Networks
- G Game on Sequence
- H Fly Me To The Moon
- I Nim Cheater
- J Jewel Grab
- K Candy Ads
- L Random Permutation

Do not open before the contest has started.

Prepared by Zhejiang University

Problem A. Accelerator

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

DreamGrid is driving a spaceship from Mars to Earth.

There are n accelerators on the trajectory to accelerate the spaceship. The i-th accelerator has an accelerating factor of a_i. The spaceship will pass the accelerators one by one. Initially, the velocity of the spaceship is 0. When the spaceship passes through an accelerator, it gains energy from the accelerator and the velocity changes. Formally, if the accelerating factor is A and the velocity before accelerating is v, the velocity after accelerating becomes $v' = (v + 1) \times A$.

However, the n accelerators are uniformly randomly shuffled. DreamGrid doesn't know the order of the accelerators passed through now. Can you tell him the expected velocity after passing through all the n accelerators?

It can be proved that the expected velocity is rational. Suppose that the answer can be denoted by $\frac{u}{d}$ where $\gcd(u, d) = 1$, you need to output an integer r such that $rd \equiv u \pmod{998\,244\,353}$ and $0 \le r < 998\,244\,353$. It can be proved that such r exists and is unique.

Input

There are multiple test cases. The first line of the input contains an integer T ($1 \le T \le 100\,000$), indicating the number of test cases. For each test case:

The first line contains an integer n ($1 \le n \le 100\,000$), indicating the number of accelerators.

The next line contains n integers $a_1, a_2, \cdots, a_n$ ($1 \le a_i \le 10^9$), indicating the accelerating factors.

It's guaranteed that the sum of n of all test cases will not exceed $100\,000$.

Output

For each test case output one line containing the integer r.

Example

standard input	standard output
3 3 1 2 3 1 10 4 5 5 5 5	665496247 10 780

Note

For the first example, there are 6 ways to order the accelerators:

- [1,2,3]: $v = ((((0 + 1) \times 1 + 1) \times 2) + 1) \times 3 = 15$
- [1,3,2]: $v = ((((0 + 1) \times 1 + 1) \times 3) + 1) \times 2 = 14$
- [2,1,3]: $v = ((((0 + 1) \times 2 + 1) \times 1) + 1) \times 3 = 12$
- [2,3,1]: $v = ((((0 + 1) \times 2 + 1) \times 3) + 1) \times 1 = 10$
- [3,1,2]: $v = ((((0 + 1) \times 3 + 1) \times 1) + 1) \times 2 = 10$

- [3,2,1]: $v = ((((0 + 1) \times 3 + 1) \times 2) + 1) \times 1 = 9$

So the expected velocity is $\frac{15+14+12+10+10+9}{3!} = \frac{70}{6}$.

Problem B. Boring Problem

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

Given a string S, n strings $T_1, T_2, \ldots, T_n$ of length m and a positive rational number sequence p of length k whose sum is 1. Each string consists of only the first k lowercase letters. Let's perform the following procedure:

1. If there exists j $(1 \le j \le n)$ such that T_j is a substring of S, stop the procedure.

2. Append the i-th lowercase letter with probability p_i to the end of S, then return to step 1.

Let's define $f(S; T, p)$ as the expected length of S when the procedure stops.

It's boring to calculate $f(S; T, p)$ for only one string S. To make the problem much harder, a string R is given. Let's denote the prefix of R of length i as $R[1 \ldots i]$. Your task is to calculate $f(R[1 \ldots i]; T, p)$ for $i = 1, 2, \cdots, |R|$.

It can be proved that $f(S; T, p)$ is a positive rational number and it can be represented as $\frac{P}{Q}$ with $\gcd(P, Q) = 1$. It is guaranteed that $Q \not\equiv 0 \pmod{10^9 + 7}$ for all strings S under the given T and p in the input. You should print the value of $PQ^{-1} \mod (10^9 + 7)$.

Input

The first line contains three positive integers n, m and k $(1 \le n \le 100,\ n \times m \le 10\,000,\ 1 \le k \le 26)$.

The second line contains k positive integers $p'_1, p'_2 \cdots, p'_k$. It is guaranteed that $p'_1 + p'_2 + \cdots + p'_k = 100$ and the probability p_i equals to $\frac{p'_i}{100}$.

The i-th line of the following n lines contains a string T_i of length m.

The last line contains a string R $(1 \le |R| \le 10\,000)$.

It is guaranteed each string consists of only the first k lowercase letters and $Q \not\equiv 0 \pmod{10^9 + 7}$ when representing $f(S; T, p)$ as $\frac{P}{Q}$ with $\gcd(P, Q) = 1$ for all strings S under the given T and p in the input.

Output

Ouput $|R|$ lines. The i-th line contains an integer representing the value of $f(R[1 \ldots i]; T, p)$.

Examples

standard input	standard output
2 2 2 50 50 aa bb ababaa	3 4 5 6 7 6
3 3 3 25 25 50 abc bac cab ababbabbcaaa	13 333333343 333333344 333333345 17 333333347 333333348 20 333333358 666666692 23 24
4 4 4 10 20 30 40 abcb cabc abbb cccc abababacabaabcca	146386692 32395942 146386694 32395944 146386696 851050282 242422295 512573933 146386700 146386701 32395951 66073407 572924730 242422302

Problem C. Club Assignment

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	512 megabytes

There are n freshmen who failed to join any club, they decided to set up two new clubs by themselves. It is encouraged to make more new friends in the club, so they want an extreme "random" partition result.

Formally, the personality of the i-th freshman can be represented as a positive integer w_i, the similarity between two freshmen A and B can be measured as $w_A \oplus w_B$, where "$\oplus$" denotes the bitwise xor operation. Your task is to assign each freshman to either the new club 1 or the new club 2, such that the smallest value of similarity between two freshmen in the same club is maximized.

Input

The input contains multiple cases. The first line of the input contains a single integer T $(1 \leq T \leq 10\,000)$, the number of cases.

For each case, the first line of the input contains an integer n $(3 \leq n \leq 100\,000)$, denoting the number of freshmen.

The second line contains n integers $w_1, w_2, \ldots, w_n$ $(1 \leq i \leq n,\ 1 \leq w_i \leq 10^9)$, denoting the personality of each freshman.

It is guaranteed that the sum of n over all cases does not exceed $200\,000$.

Output

For each case, print two lines. Print a single integer in the first line, denoting the smallest value of similarity between two freshmen in the same club in your solution. Then print n digits in the second line, denoting the solution you find. If the i-th freshman is assigned to the first club, the i-th digit should be '`1`', and if the i-th freshman is assigned to the second club, the i-th digit should be '`2`'.

If there is more than one solution, any one of them will be accepted.

Example

standard input	standard output
2 3 1 2 3 3 5 5 5	3 112 0 122

Problem D. Artifacts

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

Genshin Impact is an open-world game by Mihoyo. As the main character, you were a pair of traveling twins, passing through countless worlds during your journey. Descending upon a continent named Teyvat, you hoped that you would be able to enjoy your time here. But as you awoke among the falling stars, you saw the world in turmoil, a cataclysm raging across the land...

You sought to leave this place and move on to the next world, but then an Unknown God stood before you, barring the way. This deity was spotless, floating over a world of chaos. Looking down on you. The god took your only kin away, and you were sealed and cast into a deep slumber filled with nightmares...

When you reawakened, the world was changed. The flames of war raged no longer, and nothing was left that looked familiar. How long had you been asleep? You had no answers.

Thus you began a lonely journey, seeking the deity that you had once laid eyes upon...

In this world, you can arm yourself with artifacts to enhance your strength. There are five kinds of artifacts in total: Flower of Life, Plume of Death, Sands of Eon, Goblet of Eonothem, Circlet of Logos. You can arm at most one artifact of each kind while you can choose to not wear some kinds of artifacts.

For each piece of artifacts, it has 5 stat values. It is guaranteed that these stat values are of different types. There are 12 kinds of stat value, including: HP, ATK, HP Rate, ATK Rate, DEF Rate, Physical DMG Rate, Elemental DMG Rate, Elemental Mastery, Energy Recharge Rate, Crit Rate, Crit DMG Rate, Healing Bonus Rate.

Here, we only care about ATK, ATK Rate, Crit DMG Rate, Crit Rate (the critical rate). You have 1500 ATK initially, 5% Crit Rate, 50% Crit DMG Rate as the basis. With a total ATK Rate of $x\%$, ATK of y while arming artifacts, the new ATK will become $1500 \times (1 + x) + y$. Then We can define the expected damage E as

$$E = \texttt{ATK} \times (1 - (\texttt{Crit Rate})) + \texttt{ATK} \times (1 + (\texttt{Crit DMG Rate})) \times (\texttt{Crit Rate})$$

Specially, if Crit Rate exceeds 100%, it will be counted as 100%.

Now you are given 5 artifacts, one piece per kind. You need to compute expected damage E when you are armed with all these 5 artifacts.

Input

In the following 5×5 lines, each 5 lines describe an artifact. In each of 5 lines, each line contains a string describing one of the five stat values.

For each stat value string, it will be given under the format of "`typename+x`" or "`typename+x%`". x is a non-negative real number of no more than 2 decimals. We guarantee that the typename will be one of the 12 types described before. When the typename equals to some Rate, there will be a "`%`" in the end and x satisfies $0 \le x \le 100$. Otherwise, x will be a real number in the range of $[0, 1\,000]$.

Output

Output a real number indicating the answer. Your answer is acceptable if its absolute or relative error does not exceed 10^{-6}.

Formally speaking, suppose that your output is x and the jury's answer is y. Your output is accepted if and only if $\frac{|x-y|}{\max(1,|y|)} \le 10^{-6}$.

Example

standard input	standard output
ATK+10.00 ATK Rate+10% Crit Rate+10.00% HP+10.00 DEF Rate+10.00% Energy Recharge Rate+10.00% ATK+10.00 Crit Rate+10.00% Crit DMG Rate+10.00% DEF Rate+10% ATK+10.00 ATK Rate+10.00% Elemental DMG Rate+10% Crit DMG Rate+10.00% Crit Rate+10.00% Crit DMG Rate+10.00% ATK Rate+10% Healing Bonus Rate+10.00% HP+10 DEF Rate+10.00% ATK+10 ATK Rate+10% HP Rate+10.00% HP+10 DEF Rate+10.00%	2739.2000000000

Problem E. Mountain

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

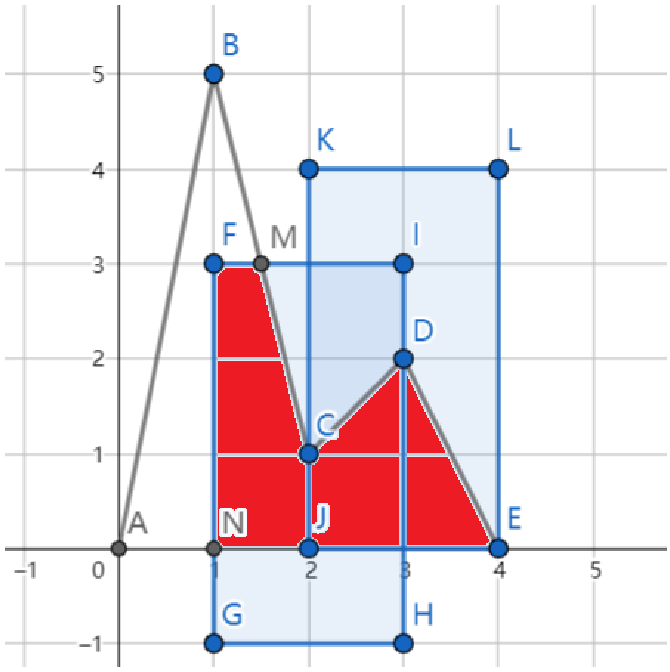
DreamGrid is climbing a mountain. The mountain is described by a polyline on a 2D plane:

$$(0,0) - (1,h_1) - (2,h_2) - \cdots - (n,h_n) - (n+1,0)$$

The region surrounded by the polyline and the x-axis denotes the mountain.

DreamGrid takes n pictures at the points (i, h_i) for each integer i where $1 \le i \le n$. A picture covers a rectangle on the plane. Formally, a picture taken at (i, h_i) covers all the points (x, y) where $i - W \le x \le i + W$ and $h_i - H \le y \le h_i + H$.

However, his hard disk has limited space. When he saves the pictures into his hard disk, he can keep only K pictures. He wants to maximize the total area of the mountain which is covered by at least one picture. You are asked to find the maximum area for $K = 1, 2, \cdots, n$.



The graph above is a sample where $n = 3, W = 1, H = 2$. The polyline describing the mountain is A-B-C-D-E. DreamGrid keeps 2 pictures taken at C and D. The red area (polygon F-M-C-D-E-N-F) is the part of the mountain covered by the kept pictures.

Input

The first line of the input contains three integers n, W and H ($1 \le n \le 200$, $1 \le W \le 5$, $1 \le H \le 10\,000$), indicating the number of points on the polyline and the size of the pictures.

The second line contains n integers $h_1, h_2, \cdots, h_n$ ($1 \le h_i \le 10\,000$), indicating the y coordinates of the n points on the polyline.

Output

Output n lines, the i-th line contains a float number indicating the maximum area when $K = i$.

Your answer is acceptable if its absolute or relative error does not exceed 10^{-6}.

Formally speaking, suppose that your output is x and the jury's answer is y. Your output is accepted if and only if $\frac{|x-y|}{\max(1,|y|)} \le 10^{-6}$.

Example

standard input	standard output
3 1 2 2 1 3	3.5000000000 4.5000000000 5.1666666667

Problem F. Fixing Networks

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

After the last network breakdown, you've been assigned to redesign the signal transmission network of ICPC (Internet Clogging Prevention Corporation)!

There are n signal stations in total, they are all newly built and can establish bidirectional signal connections with at most d other stations. ICPC demands you to reach the full potential of the network with these fancy stations. That is, all stations should have exactly d connections, no connection with oneself, and no multiple connections between the same pair of stations.

These stations will later be assigned to c departments of ICPC. Each department wants at least one station, and all stations will be assigned to these departments. To prevent another network breakdown, all stations assigned to the same department should be able to communicate with each other, while stations assigned to different departments should not be able to do so. Two stations can communicate with each other when there is a sequence of stations that begins and ends at these two stations, and every adjacent station in the sequence has a direct signal connection.

Assigning stations to departments is not your job; However, since you are probably the only responsible person left in ICPC, you want to make sure it is at least possible. Give a network proposal that satisfies all the restrictions above, or determine it is impossible.

Input

The only line of the input contains three integers n, d and c ($1 \le c \le n \le 100\,000$, $0 \le d < n$, $n \times d \le 200\,000$), denoting the number of signal stations, the number of connections each station can establish, and the number of departments.

Output

If it is impossible to connect the stations in such a way, print a single line containing "`No`". Otherwise, print "`Yes`" in the first line, then print n lines, with the i-th line containing d numbers, denoting the d stations that have connections with the i-th station. **The d numbers should be sorted in ascending order.**

The label of all stations should be integers in the range $[1, n]$. If there is more than one solution, any one of them will be accepted.

Examples

standard input	standard output
12 3 2	Yes 2 5 8 1 3 6 2 4 7 3 5 8 1 4 6 2 5 7 3 6 8 1 4 7 10 11 12 9 11 12 9 10 12 9 10 11
3 2 2	No

Problem G. Game on Sequence

Input file:	`standard input`
Output file:	`standard output`
Time limit:	6 seconds
Memory limit:	512 megabytes

Grammy is playing a game with her roommate Alice on a sequence A with n non-negative integers $A_1, A_2, \ldots, A_n$. The rules of the game are described as follows.

1. They play the game by moving the single token on the sequence, initially the token is at position k.
2. Grammy takes the first move, and they take moves alternatively.
3. In any move with the token at position i, the current player must move the token to the next position j such that $j > i$ and A_j differs from A_i on at most one bit in binary representation.
4. The player who can't make any legal move loses the game.

They play this game many times and the sequence can be modified many times. Grammy wants to ask you for some initial states who will win the game if both play optimally.

Input

The first line of input contains 2 integers n and m ($1 \le n, m \le 200\,000$), denoting the length of the sequence and the number of operations.

The second line contains n integers $A_1, A_2, \ldots, A_n$ ($0 \le A_i \le 255$), denoting the sequence A.

The next m lines each contains 2 integers op ($1 \le op \le 2$) and k, denoting each operation:

- $op = 1$ means a modification on the sequence. Grammy will append an integer k ($0 \le k \le 255$) at the end of the sequence so the sequence becomes $A_1, A_2, \ldots, A_{N+1}$ where N is the current length of the sequence before modification.
- $op = 2$ means a new game starts with the token at position k ($1 \le k \le N$), where N is the current length of the sequence. You need to predict the winner of this game.

Output

For each operation with $op = 2$, output one line containing "`Grammy`" if Grammy will win, or "`Alice`" if Alice will win when they play optimally.

Example

standard input	standard output
5 5 1 2 3 4 5 1 6 2 5 1 7 2 5 2 1	Alice Grammy Alice

Problem H. Fly Me To The Moon

Input file:	`standard input`
Output file:	`standard output`
Time limit:	4 seconds
Memory limit:	512 megabytes

In order to unravel the mystery of their life experience, Nasa and Tsukasa plan to fly to the Moon with the help of spacecrafts from the Earth. When calculating and simulating the flight orbit of the spacecrafts, they assume that the Earth is located at $(0, 0)$ and the Moon is located at $(1000, 1000)$ in the universe.

Although the whole journey is so long and difficult, thanks to advanced aerospace technology, there are space stations at all positions with integer coordinates except the Earth and the Moon in the universe. So they can take a break on their long journey and interchange other spacecrafts in these stations.

There are n types of brand new spacecrafts in each space station, and the i-th type of spacecrafts can take d_i units of fuels and always fly towards the Moon for reducing cost. More precisely, they can sail from (x, y) to $(x + dx, y + dy)$, where dx and dy are non-negative integers and $0 < dx^2 + dy^2 \leq d_i^2$, with the i-th type of spacecrafts.

During the journey, they can choose whether to land on some space stations and have a rest. Note that if they choose to take a break in a space station, they will change to a new spacecraft in that space station when they set off.

However, recently m space stations are closed for maintenance, and they cannot stay on these closed stations. Also the spacecrafts in these closed stations are not available, too. Out of concerns about this matter, Nasa is worried whether they could reach the moon successfully.

Therefore, they turn to seek your assistance. You need to help them calculate the number of different ways to fly to the Moon. Since the answer may be too large, you only need to output the answer modulo 998 244 353. Note that two ways are considered different if there exists a stage such that they choose two different types of spacecraft or fly towards two different space stations.

Input

The first line of the input contains two integers n $(1 \leq n \leq 1000)$ and m $(0 \leq m \leq 1000)$, indicating the number of spacecrafts in each space station and the number of closed space stations in the universe.

Then the second line contains n integers $d_1, d_2, \cdots, d_n$ $(1 \leq d_i \leq 1000)$, indicating the units of fuels that each type of spacecrafts can take.

Each of the following m lines contain two integers x and y $(0 \leq x, y \leq 1000)$, indicating the space station at (x, y) is closed. It is guaranteed that the given m locations are not the same as the locations of the Earth and the Moon and they are pairwise distinct.

Output

Output the number of different ways to fly to the Moon modulo 998 244 353.

Examples

standard input	standard output
`1 0` `1`	`472799582`

`1 1`
`1`
`500 500`
	`447362327`

`1 0` `2`	`277036758`

Note

For the first sample case, the answer is $\binom{2000}{1000}$.

For the second sample case, the answer is $\binom{2000}{1000} - \binom{1000}{500}^2$.

Note that $\binom{n}{k}$ is a binomial coefficient, which gives the number of k-element subsets of an n-element set.

Problem I. Nim Cheater

Input file:	`standard input`
Output file:	`standard output`
Time limit:	4 seconds
Memory limit:	8 megabytes

The memory limit of this problem (8 megabytes) is unusual!

Alice and Bob are playing the Nim game. In this game, there are several piles of stones, where each pile may contain multiple stones. Two players take turns to remove stones. The player who can't take any legal move first loses. In each move, the player can select a pile of stones, then remove a positive number of stones from it.

They will play n games in the next n days, one game per day. Initially, there are no piles of stones. On the i-th day, exactly one event will happen, and then they will play a new game. Alice always takes moves first, and both of the players will play optimally. Bob wants to win the game by cheating. Before each game, Bob can pay to delete several piles of stones such that Alice can never win. And after each game, Bob will restore all the deleted piles back.

On each day, one of the following two events will happen:

- "`ADD a[i] b[i]`" ($1 \leq a[i] < 16\,384$, $1 \leq b[i] \leq 100\,000$): Alice puts a new pile of $a[i]$ stones as the rightmost pile, which will cost Bob $b[i]$ dollars to delete it.
- "`DEL`" : Alice removes the rightmost pile. It is guaranteed that such pile always exists.

You are the best friend of Bob, please help Bob determine which piles of stones to delete for each game such that the total cheating cost is minimized. Note that Bob can always win by deleting all the piles.

Input

The input contains only a single case.

The first line of the input contains a single integer n ($1 \leq n \leq 40\,000$), denoting the number of days.

Each of the next n lines describes an event, the i-th of which denotes the event happened on the i-th day.

It is guaranteed that the number of "`ADD`" events will never exceed $20\,000$.

Output

Print n lines, where the k-th ($1 \leq k \leq n$) line contains a single integer, the minimum number of dollars that Bob needs to pay on the k-th day.

Example

standard input	standard output
7 ADD 3 10 ADD 2 4 ADD 1 5 ADD 2 1 DEL DEL ADD 3 5	10 14 0 1 0 14 4

Problem J. Jewel Grab

Input file:	`standard input`
Output file:	`standard output`
Time limit:	3 seconds
Memory limit:	512 megabytes

The museum in Byteland has n jewels on display. These jewels are arranged in a row, labeled by $1, 2, \ldots, n$ from left to right. Every precious stone is of one of n distinct colors. The color of the jewel in the i-th leftmost place is c_i while the value of it is v_i.

Grammy, the master thief, aims to steal some jewels from this museum. The museum is secured by some quite expensive alarms that unfortunately can not be hacked.

Grammy invented a device: a robotic hand that can grab some jewels without triggering any of the alarms. The hand will start at the s-th leftmost place, moving right one by one, taking all the jewels lying below the hand in the whole grab procedure (including the s-th jewel), and finish at any place Grammy likes. Grammy could easily take all the jewels using the device, but she knows that the more she takes, the harder it will be to get rid of them. She decided that the safest way is to take a set of jewels such that no two taken jewels share the same color. But soon she realized that in such a way she would miss many jewels, so she would like to control the device to skip no more than k jewels in total, in such a way the skipped jewels will not be taken.

Grammy lost most of her money betting on programming contests. So she may do the grab many times. There will be m events of two types, detailed below:

- "`1 x c v`" ($1 \le x \le n$, $1 \le c \le n$, $1 \le v \le 10^9$): The museum replaces the jewel at the x-th leftmost place with a jewel whose color and value are c and v respectively.
- "`2 s k`" ($1 \le s \le n$, $0 \le k \le 10$): Grammy plans to do a new grab. She will control the hand to start at the s-th leftmost place, and will skip no more than k jewels. Please write a program to compute the maximum total value of jewels she can take in this grab. Note that the museum will always put backup jewels to all the stolen places before the next event.

Input

The input contains only a single case.

The first line of the input contains two integers n and m ($1 \le n, m \le 200\,000$), denoting the number of jewels and the number of events.

In the next n lines, the i-th line ($1 \le i \le n$) contains two integers c_i and v_i ($1 \le c_i \le n$, $1 \le v_i \le 10^9$), describing the i-th jewel.

Each of the next m lines describes an event in formats described in the statement above.

Output

For each event of the second type, print a single line containing an integer, the maximum total value of jewels that can be achieved.

Example

standard input	standard output
5 6 1 3 2 4 3 1 2 2 3 5 2 1 0 2 1 1 2 1 2 1 4 3 3 2 3 1 2 2 2	8 8 12 3 9

Problem K. Candy Ads

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	512 megabytes

The Ingenious Candy Processing Company (ICPC) recently hits the market with a new kind of candy. The ICPC is planning to display n advertisement posters on the screen at the center of the market, labeled by $1, 2, \cdots, n$. The screen consists of multiple pixels, and each ad poster will occupy $w \times h$ pixels. Specifically, the k-th ad poster will occupy all the pixels (i, j) such that $x_k \le i \le x_k + w - 1$ and $y_k \le j \le y_k + h - 1$ from the morning of the l_k-th day to the night of the r_k-th day.

Picture from Wikimedia Commons

You are a middleman in the market. Your job is to determine which ad posters to be accepted such that no two ad posters will occupy the same pixel at the same time. You are given extra m requests from the ICPC, the k-th of which is that if both the a_k-th ad poster and the b_k-th ad poster are rejected, the ICPC will cancel the whole trade.

Please determine which ad posters to display such that the trade won't be canceled, or determine it is impossible.

Input

The input contains only a single case.

The first line of the input contains three integers n, w and h ($1 \le n \le 50\,000$, $1 \le w, h \le 2\,000$), denoting the number of ad posters and the size of each ad poster.

In the next n lines, the i-th line ($1 \le i \le n$) contains four integers l_i, r_i, x_i and y_i ($1 \le l_i \le r_i \le 2\,000$, $1 \le x_i, y_i \le 2\,000$), describing the i-th ad poster.

In the next line, there contains a single integer m ($0 \le m \le 100\,000$), denoting the number of requests.

In the next m lines, the i-th line ($1 \le i \le m$) contains two integers a_i and b_i ($1 \le a_i, b_i \le n$, $a_i \ne b_i$), describing the i-th request.

Output

If it is impossible to make such a trade, print a single line containing "`No`". Otherwise, print "`Yes`" in the first line, then print n digits in the second line, denoting the solution you find. If the i-th ad poster is accepted, the i-th digit should be '`1`', and if it is rejected, the i-th digit should be '`0`'.

If there is more than one solution, any one of them will be accepted.

Examples

standard input	standard output
2 2 2 1 2 1 1 2 3 2 2 1 1 2	Yes 10
3 2 2 1 2 1 1 1 3 1 2 2 3 2 2 3 1 2 2 3 1 3	No

Problem L. Random Permutation

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

An integer sequence with length n, denoted by $a_1, a_2, \cdots, a_n$, is generated randomly, and the probability of being $1, 2, \cdots, n$ are all $\frac{1}{n}$ for each a_i $(i = 1, 2, \cdots, n)$.

Your task is to calculate the expected number of permutations $p_1, p_2, \cdots, p_n$ from 1 to n such that $p_i \le a_i$ holds for each $i = 1, 2, \cdots, n$.

Input

The only line contains an integer n $(1 \le n \le 50)$.

Output

Output the expected number of permutations satisfying the condition. Your answer is acceptable if its absolute or relative error does not exceed 10^{-9}.

Formally speaking, suppose that your output is x and the jury's answer is y. Your output is accepted if and only if $\frac{|x-y|}{\max(1,|y|)} \le 10^{-9}$.

Examples

standard input
2
standard output
1.000000000000

standard input
3
standard output
1.333333333333

standard input
50
standard output
104147662762941310907813025277584020848013430.758061352192