Problem A. Welcome to USTC

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

The University of Science and Technology of China (USTC) is a prominent university in China and enjoys an excellent reputation worldwide. It was established by the Chinese Academy of Sciences (CAS) in 1958 in Beijing, as a strategic action by the Chinese government, to meet China's science and technology needs and increases the country's international competitiveness. CAS integrated its resources with USTC, with the aim of educating top talent in cutting-edge, interdisciplinary science and technology. The establishment of USTC was hailed as "a great event in the history of education and science of China". In 1970, USTC moved to its current location in Hefei, the capital of Anhui Province. Now the University is home to 16,245 gifted students and 2,050 dedicated faculty members, and offers customizable programs and exceptional extracurricular activities to all the students.

USTC has made outstanding contributions to the development of Chinese education, and its excellent research facilities create rich opportunities for our future scientists to reach for more. In 1978, USTC set up the first "Special Class for the Gifted Young" and the first graduate school, which were considered the most remarkable achievements in modern Chinese higher education history. USTC is also the only university that operates two national laboratories on campus, the National Synchrotron Radiation Laboratory (NSRL), which is the first national laboratory in China; and the Hefei National Laboratory for Physical Sciences at the Microscale (HFNL), one of the six national research centers approved by the Ministry of Science and Technology (MOST) of PRC.

To welcome the new USTCers, Zard, a Computer Science Students' Union member, wishes to upload one string containing a substring 'USTC' to the BBS. However, something strange happened in the BBS system. The string was inserted by many other characters randomly. Zard now wants to correct the string to ensure the modified string contains a substring 'USTC'. Since the BBS had been without maintenance for a long time, Zard can't change the string directly, but he can only execute one instruction at a time to swap two adjacent characters.

Could you please let Zard know how many instructions he needs to execute to achieve his goal?

Input

A string S ($4 \leq |S| \leq 1e4$). S contains only four uppercase letters, 'U', 'S', 'T', and 'C'; the others are lowercase letters. It is guaranteed that "USTC" is a subsequence of S, i.e. it is guaranteed that 'U' is in front of 'S', 'S' is in front of 'T', and 'T' is in front of 'C'.

Output

One integer n - the minimum instructions number Zard will need to execute.

Example

standard input	standard output
welcometUSoTC	2

Note

USTC provides a wide range of opportunities to help students develop a global perspective and cultural sensitivity, and has earned its international reputation by collaborating with more than 210 top universities and research organizations all over the world. In the Nature Index 2020, USTC was ranked Top 4 on academic in the global universities and Top 1 in China; in the U.S. News Rankings 2020, the 3rd place of the best global universities in China.

Problem B. Genshin Impact

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

"Woc*, O!"

You're right, but "Genshin Impact" is a new open-world adventure game developed independently by Mihuyo. The game takes place in a fantasy world called "Teyvat", where those chosen by the gods will be granted the "Vision" to channel the power of the elements. You will play a mysterious character named "traveler", in the freedom of travel to meet different personalities, unique ability of companions, and together to defeat strong enemies, to find lost relatives. At the same time, gradually discover the truth of the "Genshin Impact".

Z will control Amber to adventure in the game. Amber has a spell that can be cast every y seconds which has a $\frac{1}{p}$ probability to make the target burn for x seconds (if the target is already burning, the burning duration will be reset to x seconds). Now Z wants to know the expected value of how much of the total duration is covered by the burning duration if he keeps casting the spell continuously on an enemy.

Input

Each test consists of multiple test cases. The first line contains a single integer T — the number of test cases. ($T \leq 1000$)

Each of the following T lines contains three integer x, y, p. ($1 \leq x, y, p \leq 1000$)

Output

T lines. For each line, output a real number indicating the expected value of how much of the total duration is covered by the burning duration.

Your answer is considered correct if its relative error does not exceed 10^{-6}. Formally, let your answer be a and the standard answer be b, your answer is accepted if and only if $\frac{|a-b|}{b} \leq 10^{-6}$.

Example

standard input	standard output
1 2 2 2	0.500000000000000

Problem C. Yixian

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

Z is a fan of the Chinese game "Yixian Card", but he always fails. You, as a programming expert, decide to help him.

The game play is as follows: Initially, both players hold 6 cards each, and then take turns to attack each other using the cards they hold in turn according to a fixed order until one player dies, i.e. the HP is less than or equal to 0.

In other words, assuming Z plays first, Z plays his first card, then his opponent plays his first card, then Z plays his second card, then his opponent plays his second card, and so on. If all 6 cards are played, the cycle starts again from the first.

Each turn has a Standby Phase and a Battle Phase, and each turn starts with the Standby Phase, followed by the Battle Phase. The game will not end until the entire current phase is resolved after the death of any of the two players, and the game lasts for a maximum of 100 rounds.

Since you are also good at Hack Game, you can cheat to know Z's opponent's cards and their order 1 seconds before the end of the turn. So you need to figure out Z's best card order through programming quickly and then output it.

If Z can beat his opponent and there are multiple answers, then the answer with the least number of rounds needed to fight should be output. If the number of rounds is the same, the answer with the smallest dictionary order should be output.

If Z cannot defeat his opponent then output the answer with the least HP difference (i.e. his opponent's remaining HP - his own remaining HP) at the end of the game when the battle ends. If the HP difference is also the same, then output the answer with the smallest dictionary order.

Input

13 lines.

The first line contains 4 integers $h1, h2, x1, x2 (1 \leq h1, h2 \leq 2000, 0 \leq x1, x2 \leq 1e8)$, representing the maximum HP of Z, the maximum HP of Z's opponent, the XiuWei of Z, and the XiuWei of Z's opponent respectively. When the battle starts, both players are with full HP, and the player with higher XiuWei starts first. If both players have the same Xiuwei, you will cheat to help Z start first.

Lines 2 - 7 represent the 6 cards of Z, numbered 1 - 6 respectively.

Lines 8 - 13 represent the 6 cards Z's opponent has, and they will be played in the input order.

Lines 2 - 13 each start with $n (1 \leq n \leq 10)$, representing that the effect of this card consists of n instructions, followed by n strings of length 4 (containing uppercase letters and numbers), with spaces between the instructions. The instructions are resolved sequentially in front-to-back order.

The possible instructions and their effects are as follows: (x, y in the instructions are numbers between 1 - 9)

LQCx: This card costs x LingQi points to be activated. At the beginning of the Battle Phase, if a player has insufficient LingQi to active this card, this player will gain 1 LingQi but the current Battle Phase will be skipped without resolving the instructions after LQCx, and you will try to activate this card again in the next round. This instruction will only appear at the beginning of each card's instructions set.

LQGx: Gain x LingQi.

ATTx: Deal x damage to your opponent.

MyTx: Deal x damage to the enemy each time, y times in total.

JYGx: Gain x JianYi. (A player will lose all JianYi the next time after the player resolve any of the ATTx or MyTx instructions, and each JianYi a player has adds 1 to all x's in the player's next ATTx or MyTx)

JYDD: Doubles the value of JianYi you already have. (It is guaranteed that a card with either JYGx or JYDD instructions will not contain any ATTx or MyTx instructions)

DEFx: Gain x Defense. (Each Defense a player has can cancel out 1 damage by ATTx or MyTx from the other player. Defense is halved each of the player's Standby turn, rounded down)

BKDF: Break the Defense. A necessary and sufficient condition for instructions after BKDF to continue resolving is that the ATTx or MyTx instruction before (not necessarily adjacent) BKDF deals HP damage to the opponent, otherwise this card's effect resolves directly without resolving the instructions after BKDF in this card. (It is guranteed that there is at least one ATTx or MyTx command before BKDF)

CONT: Continuity. A necessary and sufficient condition for the instructions after CONT in this card to be resolved is that the player has played a card with a CONT instruction during his last turn, otherwise the card's effect resolves directly without resolving the instructions after CONT in this card.

AGAI: Action again. An additional card can be played in order during the Battle Phase of the turn, and this instruction can only take effect once during a player's turn.

WEKx: Add x levels of weakness to your opponent. (When a player's weakness level is greater than 0, multiply the x in the player's ATTx or MyTx by 75%, rounding down. At the end of the round, reduce the player's weakness level by 1.)

FRGx: Fragile. Add x fragile levels to the opponent. (When a player's fragility level is greater than 0, increase the x in the enemy's ATTx or MyTx by 50%, rounded down. At the start of the player's Standby Phase, reduce the player's fragility by 1 level.)

POIx: Add x poison levels to the opponent. (For every 1 poison level a player has during the player's Standby Phase, the player loses 1 HP.)

1. Damage calculation is in the following order: JianYi, Weakness, Fragile.

2. BKDF , CONT, AGAI instructions appear at most once in one card.

3. The battle begins with both players at full HP and without any buffs, debuffs or LingQi.

Output

3 lines.

Output YES or NO in the first line, representing if Z can beat his opponent.

In the second line, output the minimum number of rounds needed if Z can defeat his opponent. Output the least HP difference between the two players at the end of the battle if Z cannot defeat his opponent.

Output six numbers from 1 to 6, representing the order in which the cards Z should play in the third line.

Examples

standard input	standard output
31 36 120 140 2 POI1 CONT 2 CONT ATT5 4 CONT ATT2 BKDF AGAI 2 CONT ATT3 1 WEK2 1 DEF1 2 ATT6 LQG1 1 JYG2 1 JYDD 3 LQC2 FRG1 M3T2 3 ATT2 BKDF ATT9 1 DEF2	YES 23 123456
31 37 120 140 2 POI1 CONT 2 CONT ATT5 4 CONT ATT2 BKDF AGAI 2 CONT ATT3 1 WEK2 1 DEF1 2 ATT6 LQG1 1 JYG2 1 JYDD 3 LQC2 FRG1 M3T2 3 ATT2 BKDF ATT9 1 DEF2	NO 18 132456

Note

Z's opponent has a higher XiuWei, so the opponent starts first.

Turn1 (Opponent):

Deals 6 damage to Z and gains 1 LingQi. (Z: HP = 25. Opponent: HP = 36, LingQi = 1)

Turn2 (Z):

Adds 1 poison level to the opponent. (Z: HP = 25. Opponent: HP = 36, LingQi = 1, Poison = 1)

Turn3 (Opponent):

During the Standby Phase, loses 1 HP because of level 1 poison. (Z: HP = 25. Opponent: HP = 35, LingQi = 1, Poison = 1)

During the Battle Phase, gains 2 JianYi. (Z: HP = 25; Opponent: HP = 35, LingQi = 1, Poison = 1, JianYi = 2)

Turn4 (Z):

Due to having used a card with CONT instruction last turn, deals 5 damage to the opponent. (Z: HP = 25; Opponent: HP = 30, LingQi = 1, Poison = 1, JianYi = 2)

Turn5 (Opponent):

During the Standby Phase, loses 1 HP because of level 1 poison. (Z: HP = 25; Opponent: HP = 29, LingQi = 1, Poison = 1, JianYi = 2)

During the Battle Phase, doubles his JianYi. (Z: HP = 25; Opponent: HP = 29, LingQi = 1, Poison = 1,

JianYi = 4)

Turn6 (Z) :

- Deals 2 damage to the opponent. (Z: HP = 25; Opponent: HP = 27, LingQi = 1, Poison = 1, JianYi = 4)

- As damage has been dealt to the enemy, instructions after BKDF can be resolved.

- Resolving AGAI, plays another card (Card 5) this turn. Deals 3 damage to the opponent. (Z: HP = 25; Opponent: HP = 24, LingQi = 1, Poison = 1, JianYi = 4)

Turn7 (Opponent):

During the Standby Phase, loses 1 HP because of level 1 poison. (Z: HP = 25; Opponent: HP = 23, LingQi = 1, Poison = 1, JianYi = 4)

During the Battle Phase, skips this Battle Phase due to the insufficient LinqQi, but gains 1 LingQi. (Z: HP = 25; Opponent: HP = 23, LingQi = 2, Poison = 1, JianYi = 2)

Turn8 (Z):

Adds 2 levels of weakness to the opponent. (Z: HP = 25; Opponent: HP = 23, LingQi = 2, Poison = 1, JianYi = 4, Weakness: 2)

Turn9 (Opponent):

During the Standby Phase, loses 1 HP because of level 1 poison. (Z: HP = 25; Opponent: HP = 22, LingQi = 2, Poison = 1, JianYi = 4, Weakness: 2)

During the Battle Phase:

- Adds 1 level of fragile to Z. (Z: HP = 25, Fragile = 1; Opponent: HP = 22, LingQi = 2, Poison = 1, JianYi = 4, Weakness: 2)

- For the damage calculation, initially $x = 2$; 4 JianYi makes $x = 2 + 4 * 1 = 6$; 2 Weakness makes $x = \lfloor 6 * 0.75 \rfloor = 4$; Z's 2 levels of fragile makes $x = \lfloor 4 + 4 * 0.5 \rfloor = 6$. As a result, M3T2 deals $3 * 6 = 18$ damage to Z. After resolving M3T2, loses all JianYi. (Z: HP = 7, Fragile = 1; Opponent: HP = 22, LingQi = 2, Poison = 1, Weakness: 2)

At the end of this turn, reduces 1 weakness level. (Z: HP = 7, Fragile = 1; Opponent: HP = 22, LingQi = 2, Poison = 1, Weakness: 1)

Turn10 (Z):

During the Standby Phase, reduce 1 fragile level. (Z: HP = 7; Opponent: HP = 22, LingQi = 2, Poison = 1, Weakness: 1)

During the Battle Phase, gains 1 Defense. (Z: HP = 7, Defense = 1; Opponent: HP = 22, LingQi = 2, Poison = 1, Weakness: 1)

Turn11 (Opponent):

During the Standby Phase, loses 1 HP because of level 1 poison. (Z: HP = 7, Defense = 1; Opponent: HP = 21, LingQi = 2, Poison = 1, Weakness: 1)

During the Battle Phase:

- Damage calculation for command ATT2. Initially $x = 2$, weakness makes $x = \lfloor 2 * 0.75 \rfloor = 1$; Z's 1 Defense cancels out the damage.

- Since no damage was dealt to Z, instructions after BKDF can not be resolved and will be skipped.

At the end of this turn, reduces one level of weakness. (Z: HP = 7, Defense = 1; Opponent: HP = 21, LingQi = 2, Poison = 1)

Turn12 (Z):

Since 6 cards have been played for one round, starts again from the first card. Adds 1 poison level to the opponent. (Z: HP = 7, Defense = 1; Opponent: HP = 21, LingQi = 2, Poison = 2)

Turn13 (Opponent):

During the Standby Phase, loses 2 HP because of level 2 poison. (Z: HP = 7, Defense = 1; Opponent: HP = 21, LingQi = 2, Poison = 2)

During the Battle Phase, gains 2 Defense. (Z: HP = 7, Defense = 1; Opponent: HP = 21, Defense = 2, LingQi = 2, Poison = 2)

And so on until the battle is over.

Problem D. Find the Vector

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	256 megabytes

$\mathbf{x}$, $\mathbf{y}$ and $\mathbf{z}$ are vectors indexing from 0 to $n-1$. Each element of $\mathbf{x}$ and $\mathbf{y}$ is either 0 or 1, and each element of $\mathbf{z}$ is a non-negative integer.

The ith element of $\mathbf{z}$, satisfies $\mathbf{z}_i = \mathbf{x} \cdot (\mathbf{y} \ll i)$.

Here $\cdot$ denotes the vector dot product operator, and $\ll$ is the left rotation operator. i.e. $\mathbf{a} \cdot \mathbf{b} = \sum_i^n \mathbf{a}_i \mathbf{b}_i$, and if $\mathbf{b} = \mathbf{a} \ll k$, $b_i = a_{(i+k) \bmod n}$.

Given n, $\mathbf{y}$ and $\mathbf{z}$, your target is to find $\mathbf{x}$ that satisfies the above condition. It is guaranteed that the test data is generated randomly and $\mathbf{x}$ has a unique solution.

Input

The first line contains an integer n. ($1 \le n \le 10^5$).

The second line contains a vector $\mathbf{y}$ with n either 0 or 1 integers.

The third line contains a vector $\mathbf{z}$ with n non-negative integers. ($1 \le \mathbf{z}_i \le 10^9$)

Output

One line contains vector $\mathbf{x}$.

Example

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

Note

x = (1 0 1 1 0)

y = (1 1 1 0 1)

z[0] = (1 0 1 1 0) * (1 1 1 0 1) = 1+0+1+0+0 = 2

z[1] = (1 0 1 1 0) * (1 1 0 1 1) = 1+0+0+1+0 = 2

z[2] = (1 0 1 1 0) * (1 0 1 1 1) = 1+0+1+1+0 = 3

z[3] = (1 0 1 1 0) * (0 1 1 1 1) = 0+0+1+1+0 = 2

z[4] = (1 0 1 1 0) * (1 1 1 1 0) = 1+0+1+1+0 = 3

–>

z = [2 2 3 2 3]

x satisfies the condition.

Problem E. My friends' friends are my friends

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

Our cute little Ustcy took a big task. He was asked to hold the ICPC competition in Hefei. Little Ustcy can't do it alone! So he asks some of his friends for help. But this task is so difficult that Ustcy's friends who received the task have to ask their friends to help. And their friend's friends ask for their friend's friend's friends and so on. Can our little Ustcy hold the ICPC competition in Hefei with the help of his friends' friends' ... friends? Little Ustcy does not know, and that is why he asks you to figure it out.

The problem is formulated as follows:

Given an undirected connected graph with n vertices $v_1, v_2...v_n$, representing Ustcy and all the students who are being asked to help, and m weighted edges with weight w at each edge. There are also two weights a and b at each vertex.

Initially, there is an influence value of M at node s that can be arbitrarily subdivided. When the influence value of q_p passes from s to p through the path $t_1, t_2, ..., t_k$, it decays to $q_p/(w_{t_1} + w_{t_2} \dots + w_{t_k})$. If the influence value at vertex p is not less than a_p then the $p-$th student will make b_p contribution value to ICPC competition in Hefei.

Holding ICPC in Hefei requires C contribution value in total, is there a way that the influence can flow on the graph such that eventually, the influence value at vertex p is m_p and $\sum_{m_p \geq a_p} b_p \geq C$?

If so, output "Yes"; Otherwise, output "No".

Input

The first line contains $n, m, s, C, M (1 \leq n \leq 10^4, 1 \leq m \leq 10^6, 1 \leq s \leq n, 0 \leq C \leq 10^4, 0 \leq M \leq 10^{12})$ representing the number of vertices, the number of edges, the indices of who little Ustcy is, the contribution value required for holding ICPC, and the total influence value Ustcy has.

Each of the following n lines contains two integers a, b, representing this student needs a influence value to make b contribution value. $(0 \leq a \leq 10^4, 0 \leq b \leq 10^4)$

And for the next m lines, each line contains three integers u, v, w, representing the edge between u and v, and the weight on it is w. $(1 \leq u, v \leq n, 1 \leq w \leq 10^4)$

Output

Output "Yes" or "No".

Example

standard input	standard output
5 4 1 9 15 1 7 4 1 3 5 2 2 5 3 1 2 1 1 3 1 3 4 2 3 5 2	Yes

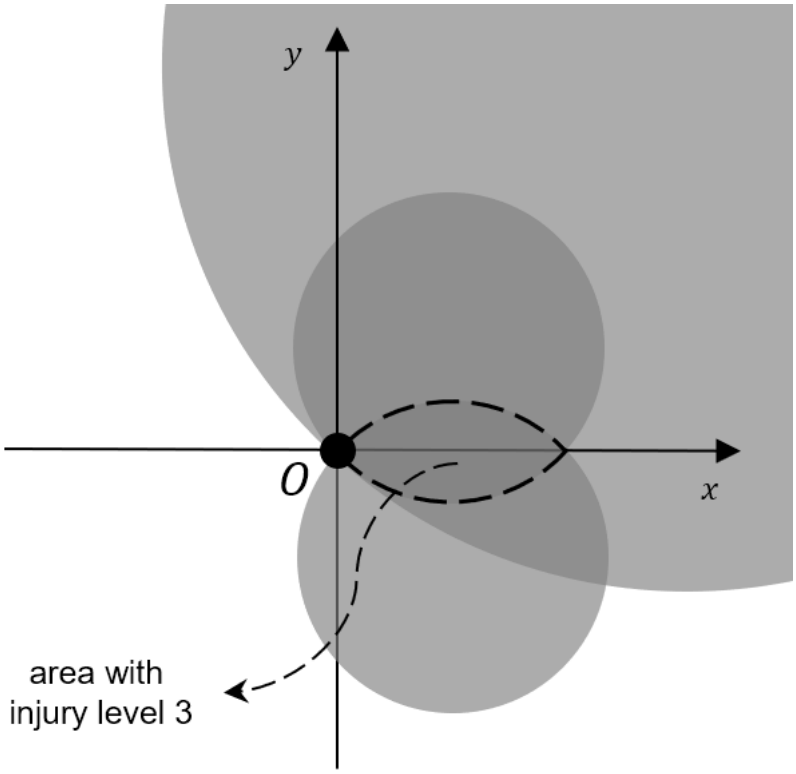
Problem F. Rescue

Input file:	`standard input`
Output file:	`standard output`
Time limit:	6 seconds
Memory limit:	256 megabytes

Hundreds of years in the future, aliens invade the Earth, when only one country remains. The aliens conduct n bombardments near the country's capital O. The capital O can be considered as a point on the plane, and each bombardment forms a circular area on this plane, where the level of injuries increases by 1. For the purpose of provocation, the circle formed by each bombardment must pass through the point O. Capital O has anti-aircraft devices, so it will not be affected by the bombardments.

The humans decide to fight back, but before they do so, they need to rescue the wounded. They decide to rescue the people in the area with the theoretical maximum injury level (i.e., the injury level of n) first. The humans want to know the distance of the farthest point from the capital on the area with the theoretical maximum injury level so that they can rationalize their rescue plan.

Given the coordinates of the center of n bombings relative to the capital O, determine if there is a region with the injury level of n, if not, output "No", otherwise output the distance from the capital to the farthest point of the region.



Input

The first line contains an integer n. ($1 \le n \le 10^6$)

Each of the following n lines contains two integers $x, y (0 \le |x|, |y| \le 10^4, (x, y) \ne (0, 0))$,representing the coordinates of the ith bombing center relative to the capital O.

Output

If there is no region with the injury level of n, print "No", otherwise print the distance from the capital to the farthest point of the region.

Your answer is considered correct if its absolute or relative error does not exceed 10^{-6}. Formally, let your answer be a and the standard answer be b, your answer is accepted if and only if $\frac{|a-b|}{\max(1,b)} \leq 10^{-6}$.

Examples

standard input	standard output
3 10 10 1 1 1 -1	2.000000000
2 0 1 0 -1	No

Problem G. Game Plan

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

Recently, Ming has become interested in a game with n characters. According to his preferences, Ming numbers those characters from 1 to n. The smaller the number of characters, the more Ming prefers.

The game releases two characters per day (two characters may be the same, and characters that have already been released may be released again). Ming has limited funds to acquire only one of the two characters each day.

Ming gets a list of characters available in the following t days. Now Ming comes to you to plan his character selection. He wants the character with the smallest number among the characters he does not own to have the largest number possible after t days.

Can you tell him the character he does not own with the smallest number?

Input

The first line contains two integers n and t, indicating the total number of characters in the game and the number of days to be played.

Each of the following t lines contains two integers a_i, b_i, representing the numbers of the two characters that will be released in the ith day.

$1 \le n \le 10^9$

$1 \le t \le 10^6$

$1 \le a_i,\ b_i \le n$

Output

Print n integer representing the number of the smallest character among the characters Ming does not own after t days.

If Ming has all n characters, print $n + 1$.

Examples

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

Note

Sample1: Ming can own all 5 characters.

Sample2: Select the characters numbered 2, 1, 3, 4, and 6, respectively, and the smallest numbered character that cannot be owned is 5.

Problem H. Jackpot

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

There is a slot machine in front of Nana's house. This slot machine has n identical reels with $k+1$ different symbols on each reel, one of which is a panda and one of which is a dragon. Each time you pull the slot machine rocker, n reels will spin to appear different symbols, and when n reels all appear the panda symbol, you can get the slot machine Jackpot.

After many days of observation, Nana found that this slot machine has a corresponding hidden button for each reel. When the button is pressed, the corresponding reel will be selected. After pulling the rocker, the symbols of the other $n-1$ reels will only exchange positions with each other, and the selected reel has the following characteristics:

Suppose the panda symbol is numbered 1, the dragon symbol is numbered $k+1$, and the other symbols are numbered from 2 to k. If the current symbol on the reel is numbered $j(2 \leq j \leq k+1)$, then after one pulling of the rocker, the symbol on that reel is numbered i, satisfying $i < j$, i.e., a symbol with a smaller number will appear with equal probability. If the current symbol on the reel is a panda, it will not change.

Assume that Nana starts playing this slot machine with all n reels showing dragons and he must choose one and only one reel at a time to cheat. Nana is smart enough, what is the expected minimum number of times he will need to pull the rocker to get one Jackpot?

Input

Each test consists of multiple test cases. The first line contains a single integer T — the number of test cases. ($T \leq 1000$)

Each of the T lines contains two integers n and k. ($1 \leq n \leq 1e9, 2 \leq k \leq 1e3$)

Output

T lines.

Each of the T lines contains one integer t, representing the expected minimum number of times Nana will need to pull the rocker to get one Jackpot.

It can be shown that the answer can be expressed as an irreducible fraction $\frac{p}{q}$, where p and q are integers and $q \equiv 0 (\text{mod } (10^9+7))$. The output integer t is equal to $p \cdot q^{-1} \bmod (10^9+7)$. In other words, output such an integer t that $0 \leq t \leq (10^9+7)$ and $t \cdot q \equiv p (\text{mod } (10^9+7))$.

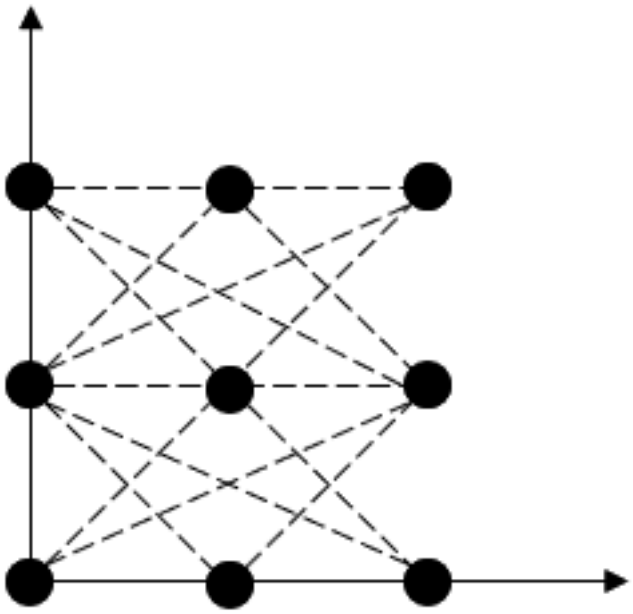
Example

standard input	standard output
2 1 2 2 2	500000005 3

Problem I. Counting Lines

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

As shown in the figure, given a $n * n$ array of points on a two-dimensional coordinate axis. Define the set of straight lines L as follows, for any line $l \in L$, l passes through at least two points on the array, and the angle between l and the x axis (the angle between the two lines is taken to be less than $\pi/2$) does not exceed $\pi/4$. Please find $|L|$.



Input

Each test consists of multiple test cases. The first line contains a single integer T — the number of test cases.

Each of the T lines contains an integer n. ($1 \leq T \leq 1000, 2 \leq n \leq 10^6$)

Output

For each test case, print a single line containing $|L|$ mod 998244353.

Example

standard input	standard output
1 3	13

Problem J. Produce the Problems

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

You have been invited to produce the problems for a programming competition in your spare time. But after a week of overtime, you don't have time to polish the problems carefully. Now you have to roll the dice to decide how to combine data structures and combinatorial mathematical tools layer by layer like Russian nesting dolls to produce problems quickly.

However, when you've finished your work. You realize that if you continuously use more than a layers of data structures or more than b combinatorial math tools continuously in a problem, you will be suspected of producing the problems as Russian nesting dolls.

Now you need to rearrange the data structures and combinatorial mathematical tools used in each problem to ensure that the way you produce the problem is not found. As moving a data structure or combinatorial mathematical tool to another problem would result in more workload, you can only swap it with another data structure or combinatorial mathematical tool within the same problem. **And, to minimize the possibility of being suspected, you want the count of "continuous use" to be across the sequential problems.**

Given the number of problems n, the number of data structures, and the number of combinatorial mathematical tools you can use for each problem, can you rearrange them such that the problems you produce are not suspected?

Input

The first line contains three integers n, a, b.

The second line contains n integers d_i indicating the number of data structures for each problem.

The third line contains n integers c_i indicating the number of combinatorial mathematical tools for each problem.

$1 \le n \le 10^6, 0 \le a, b, c_i, d_i \le 10^6$

Output

Print 1 if you can do it, otherwise 0.

Example

standard input	standard output
2 2 1 5 5 2 2	1

Note

Let the data structure be D and the combinatorial mathematical tools be C, using parentheses to separate the problems. A legitimate arrangement is (DDCDDCD)(DCDDCDD).

Problem K. Quarantine

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

There is a sudden outbreak of a virus in a city. Due to limited resources, the government can only test the streets of an area for the virus at a given moment. If multiple streets in the area are detected as being infected with the virus, the area will be immediately quarantined at that moment. The government needs to investigate the quarantine situation of an area at some moment and needs you to assist them.

Input

The first line contains two integers $n, m (1 \le n \le 10^9, 1 \le m \le 10^5)$, representing that there are n streets and m moments in total.

Each of the following m lines contains some integers, the first of which is k, representing the type of event. k is an integer among $1, 2, 3, 4$. (Initially, $A = 0$ in the following events.)

- $k = 1$, followed by 2 integers $l, r (1 \le l \le r \le n)$, representing streets $[\max(l - A, 1), \min(r + A, n)]$ are infected with the virus.
- $k = 2$, followed by 3 integers $l, r, k (1 \le l \le r \le n, 0 \le k \le r - l + 1)$, representing streets $[\max(l - A, 1), \min(r + A, n)]$ are tested. (**Streets that are being quarantined can still be tested**) If there are at least k consecutive streets in the area infected with the virus, the whole area will be quarantined.
- $k = 3$, followed by 2 integers $l, r (1 \le l \le r \le n)$, representing streets $[\max(l - A, 1), \min(r + A, n)]$ are completely recovered. (**If there is a virus infection, the virus will disappear at this moment, and if there is a quarantine, the quarantine will be released at this moment for all these streets.**)
- $k = 4$, followed by 2 integers $l, r (1 \le l \le r \le n)$, representing querying the number of the quarantined streets within $[\max(l - A, 1), \min(r + A, n)]$. Update A after querying. (i.e., Let A be the query result.)

Output

Output the query result for each $k = 4$.

Example

standard input	standard output
10 6 1 1 5 2 3 8 3 3 4 5 4 1 6 3 1 1 4 1 6	2 3

Note

10 streets and 7 moments. $A = 0$ initially.

At the 1st moment, the 1st to 5th streets are infected with the virus.

At the 2nd moment, the 3rd to 8th streets are tested, the 3rd to 5th consecutive 3 streets are infected so the 3rd to 8th streets are quarantined.

At the 3rd moment, the 4th to 5th streets are completely recovered.

At the 4th moment, the 1st to 6th streets are queried, the 3rd and 6th streets are quarantined, so the result is 2. Then let $A = 2$.

At the 5th moment, $l = 1, r = 3$, the 3rd street is completely recovered.

At the 6th moment, $l = 1, r = 8$. The 6th, 7th, 8th streets are quarantined, so the result is 3. Then let $A = 3$.

Problem L. Combination

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

Given a, k, w, you are asked to solve $\sum_{i=k}^{a} \binom{i}{k} \times \lfloor \frac{i}{k} \rfloor^w \pmod{998244353}$.

Input

Each test contains multiple test cases. The first line contains the number of test cases T.

Each of the following T lines contains three integers a, k, w.

$a \in [1, 10^9]$

$k, w \in [1, 10^3]$

$k \leq a$

$T \leq 10^3$

Output

For each test case print one integer — the answer.

Example

standard input	standard output
1 846 10 2	811409981

Problem M. Mahjong

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

In this problem, we consider a simplified version of mahjong.

We say a multiset S is valid, if S is consisted of integers from 1 to 9, and $cnt_i(S) \leq 4(1 \leq i \leq 9)$ (where $cnt_i(S)$ is the number of i in S).

We say a valid multiset S is winning, if S can be splited into $S_1, S_2, \ldots S_5$, where $S_1 \in \{\{i, i\} \mid 1 \leq i \leq 9\}, S_2, S_3, S_4, S_5 \in \{\{i, i, i\} \mid 1 \leq i \leq 9\} \bigcup \{\{i, i+1, i+2\} \mid 1 \leq i \leq 7\}$.

We say a valid multiset S is listening, if $|S| = 13$.

We say a listening multiset S is listening i, if $S \bigcup \{i\}$ is winning. And we define listening set of S as $\{i \mid S \text{ is listening } i\}$.

Now you are given a set ls, and you need to find a listening multiset S such that the listening set of S is ls, or output it's impossible.

Input

There are multiple test cases in this problem.

The first line of input contains one integer $T(1 \leq T \leq 1000)$, representing the number of test cases.

Each of the following T lines contains one string s. If s is "0 then ls is empty. Otherwise s consists all elements in ls. It is guaranteed that $|s| \leq 9$ and $s_i \in \{'1', '2', \ldots '9'\}, s_i \neq s_j (i \neq j)$.

Output

T lines.

For i-th test case, if the answer if impossible, you should output "impossible". Otherwise you should output a string consisted of all elements in S.

Example

standard input	standard output
4 0 1 129 123456789	3677788889999 1777788889999 impossible 1112345678999