Problem A. Exotic . . . Ancient City

Input file:	`standard input`
Output file:	`standard output`
Time limit:	3 seconds
Memory limit:	256 megabytes

"Welcome to the ancient city Xi'an!"

Coming down the gangway and coming up the stairs, Rikka feels a strong exoticism among the simplified Chinese characters and clean frameless designs.

It all started with that letter, an invitation to Xi'an for a one-week trip from the author named LCR. The mysterious name refers to a girl who is probably from Xi'an or Yuyang — Rikka got such information after a week of hard work. As the lord of Evil Eye, she has a great interest in this, even stronger than that to the city with a millennium history which used to be the capital of 13 dynasties.

Finally, Rikka reached this less valued city, looking forward to an amazing meeting quixotically.

But LCR never appeared. Bored Rikka noticed a map of Xi'an on the wall of the hall. The roads inherit Tang Chang'an city's mesh layout, just like a chessboard. Rikka has felt its convenience since visiting Kyoto, but she also knows about the problems. Crossings block the traffic and the old roads occupy the space for new efficient designs.

This has reminded the girl about her perfect unique solution. Her traffic system includes a set of teleporter vertices V and a set of some bidirectional horizon-wormhole edges E linking them. People can teleport from one terminal to the other in no time. The vertices form an $n \times (m+1)$ grid, that is, there are n rows and $(m+1)$ columns, containing $n \cdot (m+1)$ vertices in total. Let $\langle x, y \rangle$ $(x, y \in \mathbb{N}, x \in [1, n], y \in [1, m+1])$ be the vertex at the x-th row of the y-th column. In her initial design, every edge is between two vertices in distinct adjacent columns, and for each pair of adjacent columns, the edges among them are similar. That is, if $(\langle x, y \rangle, \langle z, y+1 \rangle) \in E$, then for $i = 1, 2, \ldots, m$, $(\langle x, i \rangle, \langle z, i+1 \rangle) \in E$ holds. Also, people can travel by the edges between any pair of vertices in each two-column subsystem, which means, for $i = 1, 2, \ldots, m$, if we remove all vertices and edges except those in or between the i-th column and the $(i+1)$-th column, the remaining vertices are still connected. No two edges connect the same pair of vertices.

Actually, the horizon-wormholes are expensive. Rikka knows each edge has an integer cost level, but the cost levels are fairly low because the energy of horizon-wormholes is too large and chaotic to measure or calculate accurately. The edges connecting the same pair of rows in every column have the same cost. That is, $\forall x, y, i, j$, $w((\langle x, i \rangle, \langle y, i+1 \rangle)) = w((\langle x, j \rangle, \langle y, j+1 \rangle))$ if $(\langle x, i \rangle, \langle y, i+1 \rangle), (\langle x, j \rangle, \langle y, j+1 \rangle) \in E$, where $w(e)$ is the cost of edge e. Now Rikka wants to delete some (maybe zero) edges in order to minimize the total cost which is the sum of costs of all remained edges. The connectivity among all vertices must hold. That is, we can still travel between each pair of vertices, possibly passing any other columns. However, connecting each two-column subsystem or keeping edges between each pair of adjacent columns the same is not necessary.

Also, she may only use a consecutive part of columns of her initial design, so the answer for each fewer m is needed. Could you help her?

Input

The first line contains three positive integers n, M, e ($1 \leq n \leq 10^5, 1 \leq M \leq 10^5, 1 \leq e \leq 2 \times 10^5$), describing the number of rows, the maximal possible columns and the number of edges in each pair of adjacent columns, respectively. You need to calculate the answer for each subsystem of $(m+1)$ ($1 \leq m \leq M$) columns while the edges between each pair of columns remain.

Each of the following e lines describes a group of edges, containing three positive integers u, v, w ($1 \leq u, v \leq n, 1 \leq w \leq 30$), meaning that for $i = 1, 2, \ldots, m$ there is an edge $(\langle u, i \rangle, \langle v, i+1 \rangle) \in E$ with a cost of w. No two edges connect the same pair of vertices, and people can travel between any pair of vertices in the subsystem of each m.
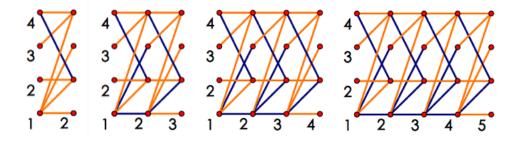
Output

Output M lines. The m-th line contains a single integer, the minimum total cost for the subsystem of $(m+1)$ columns.

Examples

standard input	standard output
4 4 8	62
3 4 12	80
1 1 20	98
1 3 22	116
4 2 12	
4 4 2	
2 2 2	
1 2 2	
1 4 2	
6 6 15	19
1 2 1	28
1 3 1	37
3 4 1	46
2 4 1	55
6 3 2	64
6 5 2	
3 5 2	
2 3 2	
4 3 2	
6 4 2	
5 4 2	
4 6 2	
6 6 2	
5 5 3	
5 1 3	

Note

The subsystems of all possible m ($1 \le m \le M$) in sample 1 are shown below. The images show one optimal way to delete the roads: the deleted roads are painted blue (as dark grey if the document is printed in black and white), and the others are painted orange (as light grey if the document is printed in black and white).

Problem B. Mysterious . . . Host

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

Finally, Rikka got an excellent design which not only creates efficient traffic but also costs little money. But Rikka didn't care whether the trite officers accepted it — LCR had been behind her for long, with a chaste smile, a garland, and a white dress. Gradually calmed down by the smile, the girls reached out their hands and began their journey. They ended up at a self-study room in the Middle School Attached to Northwestern Polytechnical University after the romantic meeting.

The formulas on the whiteboards and papers have reminded Rikka (about her study, of course), so she is asking LCR to play a game with her now (in order to review combinatorics).

LCR has an n-order permutation and Rikka tries to guess it. At first, Rikka should choose a set of n-order permutations. LCR will make some queries then. Each time, LCR gives a segment $[L, R]$ ($1 \leq L \leq R \leq n$) and for each chosen permutation Rikka answers if the segment is *consecutive* (defined in the following paragraphs) in it. In the end, Rikka will win the game if and only if there exists a chosen permutation which has the same answer to each query as LCR's original one.

Rikka wonders how many permutations she has to choose at least to ensure winning the game. For future games, she needs the answer for each positive integer n not greater than N. She only needs the answer modulo a certain prime P because its exact value may be too large.

The segment $[L, R]$ is *consecutive* in an n-order permutation p if and only if there doesn't exist three integers x, y, z such that $1 \leq x, y, z \leq n$, $p_x < p_y < p_z$, $x, z \in [L, R]$, $y \notin [L, R]$, where p_i ($i = 1, 2, \ldots, n$) is an integer in $[1, n]$, which means the i-th element in the permutation p.

Input

The only line contains two integers N ($1 \leq N \leq 5000$), P ($1 \leq P < 2^{30}, \exists k \in \mathbb{N}, P = k \cdot 2^{14} + 1$), the maximal order of the permutations and the divisor, separated by a space. It is guaranteed that P is prime.

Output

Output N lines; for $n = 1, 2, \ldots, N$, the n-th line contains one integer, the minimum number of permutations which Rikka needs to choose for order n, modulo P.

Example

standard input	standard output
10 65537	1
	1
	3
	12
	52
	240
	1160
	5795
	29681
	23951

Note

If $n = 3$, there are 6 permutations and 6 possible queries. Rikka's responses in these situations are as follows:

	$(1,2,3)$	$(1,3,2)$	$(2,1,3)$	$(2,3,1)$	$(3,1,2)$	$(3,2,1)$
$[1,1]$	True	True	True	True	True	True
$[2,2]$	True	True	True	True	True	True
$[3,3]$	True	True	True	True	True	True
$[1,2]$	True	False	True	True	False	True
$[2,3]$	True	True	False	False	True	True
$[1,3]$	True	True	True	True	True	True

The answer should be 3. For example, Rikka can choose $(1, 2, 3), (1, 3, 2), (2, 1, 3)$, so that no matter which permutation LCR has, there exists a chosen permutation which has the same answer to each query as LCR's.

Problem C. Heretical ... Möbius

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

Rikka was walking around the school building curiously until a strange room with a door number of 404 caught her eyes.

It seemed like a computer room — there were dozens of computers lying orderly, but papers, pens, and whiteboards everywhere built up a nervous atmosphere. Suddenly, Rikka found some mysterious codes displayed on a computer which seemed to have nothing different from others — is this a message from *inner world*?

Excited Rikka started her exploration. The message was generated by a program named *for_patterns_in_mobius* which outputted a string s of length 10^9, containing the value of $|\mu(x)|$ for $x = 1, 2, \ldots, 10^9$ in order.

Suddenly, Rikka heard footsteps outside. She quickly took a screenshot and left. The screenshot recorded a string t of length 200, perhaps a substring of s. Now Rikka wonders if it is really a substring of s, and if so, where it first appears in s.

Could you help her to decipher the codes?

Input

There are 10 lines in total. Each line contains 20 characters, each of which is either "0" or "1". t is the concatenation of them — the result of concatenating them in order.

Output

Output a single integer in the only line. If t is a substring of s, output the first position it appears in s, that is, the minimum positive integer p such that all the digits $|\mu(p + i)|$ for $i = 0, 1, \ldots, 199$ form the string t. Otherwise output -1.

Examples

standard input	standard output
11101110011011101010 11100100111011101110 11100110001010101110 11001110111011001110 01101110101011101000 11101110111011100110 01100010111011001110 11101100101001101110 10101110010011001110 11101110011011101010	1
01010101010101010101 10101010101010101010 01010101010101010101 10101010101010101010 01010101010101010101 10101010101010101010 01010101010101010101 10101010101010101010 01010101010101010101 10101010101010101010	-1

Note

The definition of $\mu()$ is as follows:

For any positive integer x, let $x = \prod_{i=1}^{k} p_i{}^{c_i}$ be the regular factorization of x, where p_i is a unique prime, c_i is a positive integer, and if $x = 1$ then $k = 0$. Consequently, $\mu(x)$ is defined as

$$\mu(x) = \begin{cases} 0 & \exists c_i > 1, \\ (-1)^k & \text{otherwise} \end{cases}$$

Problem D. Deja vu of . . . Go Players

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

An ordinary sequence rather than magic codes? Rikka's opinion is much more important than the "truth" itself. She soon felt sleepy and fell asleep to take a trip to dreamlands in passing.

Rikka found two elders playing Go game when her sanity got back. The ethereal clouds, hardy vigorous pines, and rugged rocks shocked her. A fiction *isekai*! Excited Rikka's eyes ran around and around and focused on the Go chessboard in the end.

She found that the two players, wearing white and red respectively, were not playing Go game — they divided the black and white chess pieces into some piles, and were taking turns removing them. They kept silent for her questions, so Rikka had to stand there and keep her eyes on the chessboard.

They seemed to be playing an unexpectedly simple game. The red player has n black piles and its opponent has m white piles at the beginning. They take turns removing any positive number of chess pieces from arbitrary one of their assigned piles. Red goes first, and the player who first removes all chess pieces assigned to oneself "wins", and the other player has to drink.

Drinking is illegal for minors in Japan, so Rikka wonders if she can ensure to win when she is the red player.

Input

The first line contains an integer T ($1 \le T \le 100$), the number of test cases. Then T test cases follow.

The input format of each test case is as follows:

The first line contains two integers n, m ($1 \le n, m \le 100$), the numbers of piles of the red and the white player, respectively.

The following line contains n integers a_i ($1 \le a_i \le 10^9$), in which each integer indicates the number of pieces in a black pile of the red player.

The following line contains m integers b_i ($1 \le b_i \le 10^9$), in which each integer indicates the number of pieces in a white pile of the white player.

Output

Output a string in the only line, "Yes" if the red player who moves first can ensure to win, or "No" otherwise, without quotation marks.

Example

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

Problem E. Immortal . . . Universe

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

Time flies fast. Days passed and the game finally ended. Rikka has never drunk with your help, so she is still a reputable Japanese citizen.

The red and white left gracefully. Suddenly, Rikka realized that the white player is LCR rather than any elder — or LCR is the one. She would never know why she didn't get it. Rikka ran after them, but she lost her way while passing fogs.

Spaceships are making their epic voyages across the sky, row upon row of roads and buildings forming concentric circles, reminding humans of alters or Eight-Diagrams. At the end of the fogs, a holy incredible view shows in front of her. The big round planet soon attracts her — Earth, the home of humankind, is hanging like a big mooncake. What a lunar metropolis!

Rikka finds herself in front of a stock exchange. A boy, who is gazing at the screen and knocking on the keyboard, catches her eyes.

There are two types of shares. Both are controlled by consortiums, so their trends are established.

Each type of share can be divided into some stages, and the stages will process in order. One type has n stages and the other one has m of them. In each stage, one will gain or lose a unit of money. The boy has only one unit of money at first, and he has to process a stage of an arbitrary type (unless all stages of that type have processed, and if so, the other type will process) every day. After $(n+m)$ days all transactions will end and he will finally get his money.

The poor boy knows nothing to help him to make decisions, so he chooses randomly. However, after countless bankruptcies, he has got a keen intuition: if he only has one unit of money, he will know the results of all available stages. A stage is available if and only if he can choose it immediately. In this case, he will never choose the type which will lead him to lose money unless all available types are so. But if he has at least two units of money, he will ignore any known information and choose randomly because he has chances. He has to process every day, and he cannot exit before all transactions have been completed because he can get cash only after $(n+m)$ days.

He will go bankrupt whenever his money runs out. It happens if he only has one unit of money and every available type (which has at least one stage not processed) can lead him to lose it in the next stage.

Of course, the kind-hearted girl will try her best to help. She finds something strange in the codes she got before. It is a hash value of the files from the consortiums!

Soon after, Rikka gets two strings of lengths n and m respectively. Each string contains three types of characters "P", "V" and "?", which mean that the corresponding stage can lead to loss, gain or unknown, respectively. However, the boy doesn't believe her.

Rikka gets anxious. She wonders, according to her information, how many possible trends will never lead him to bankruptcy, modulo 998244353.

A possible trend can be described by two strings of lengths n and m containing only "P" and "V", and they can be obtained by changing each "?" in Rikka's strings into "P" or "V". It will never lead to bankruptcy if and only if no matter how he chooses each time, he can never go bankrupt.

Input

The first line contains a string s of length n ($1 \le n \le 5000$), consisting of only "P", "V" and "?".

The second line contains a string t of length m ($1 \le m \le 5000$), consisting of only "P", "V" and "?".

s and t make up Rikka's information.

Output

Output a single integer, the number of possible trends which will never lead the boy to bankruptcy, modulo 998244353.

Examples

standard input	standard output
PV ??	1
??? P?P	3
????? ?????	326

Note

In the first sample, there are 4 possible ways to replace "?" with "P" or "V" as follows:

- "PV" and "PP": The boy has no chance but to lose all his money on the first day.

- "VV": On the first day, since the boy has only one unit of money, he would choose the second type to escape bankruptcy, and thus at the start of the second day, he has two units of money in his hand. It is impossible for him to go bankrupt because there is only one "P" left.

- "VP": On the first day, the boy can only choose the second type. But on the second day, both types can be chosen. If he chooses the second type, he will have to choose the first type on the third day, and then he will go bankrupt.

Since only "VV" would never lead the boy to bankruptcy, the answer to the first sample should be 1.

Problem F. Interstellar . . . Fantasy

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

Unfortunately, the boy finally got bankrupt. Looking at his receding figure, Rikka understood more about the world — but she is still herself. Though, she still sat on the tower and spent time with the low gravity, feeling lost.

She looked at the deep dark sky, where the blue round Earth is shining overhead. Rikka thought about her families, her friends, and her homeland. Is she in her dream or a "real" world? The girl of *chunibyo* felt afraid for the first time since the journey began.

She saw a bright star traveling fast around the earth — maybe a geostationary space station. How could she get there? Daydream started again.

In other words, Rikka wonders the minimum distance she needs to travel from her position s to the position of the star t, while a sphere — the earth staying there as an obstacle. They are in a 3-dimensional Euclidean space. s and t may be at the same position.

Input

The first line contains an integer T ($1 \le T \le 1000$), the number of test cases. Then T test cases follow.

Each test case contains two lines.

The first line contains four integers o_x, o_y, o_z, r, the positions in each dimension of the center of the sphere o and its radius.

The second line contains six integers $s_x, s_y, s_z, t_x, t_y, t_z$, the positions in each dimension of the starting point s and the terminal point t, respectively. Notice that s and t may be at the same position.

It is guaranteed that neither s nor t is strictly inside the obstacle, and each value of a position or a radius in the input belongs to $[1, 1000]$.

Output

Output one number, the minimum distance from s to t without entering the sphere obstacle. Your answer is considered correct if the absolute or relative error doesn't exceed 10^{-6}.

Example

standard input	standard output
2	3.14159265
2 1 1 1	2.64517298
1 1 1 3 1 1	
2 1 1 1	
1 2 2 3 1 1	

Problem G. Omnipotent . . . Garland

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

Rikka finally fell asleep on the top of the lunar tower. She dreamed about LCR's unique garland which she had always treated with great interest. The garland is made of n flowers of two types: *Bauhinia Blakeana* and *Cerasus Yedoensis*. For convenience, we use abbreviations to call them "B" and "C". The n flowers form a ring, that is, each flower has an integer index in $[0, n)$ such that the flowers i and $((i + 1) \bmod n)$ are adjacent.

Rikka has chosen two magic positive integers m, k. Now she wants to divide the garland into m shorter ones such that the length of each sub-garland is a multiple of k, flowers in each sub-garland keep in order as they used to be in the garland, and there exist two distinct "B"s in each sub-garland that are adjacent. You need to answer if there exist valid partitions, and if so, output any of them.

Formally, let $t[i]$ ($i = 0, 1, \ldots, n - 1$) be the type of the flower with index i (either "B" or "C"). A sub-garland containing c flowers can be described as an ascending sequence $A = \{A_0, A_1, \ldots, A_{c-1}\}$ ($A_i < A_{i+1}, i = 0, 1, \ldots, c - 2$), which represents the indices in the original garland of those flowers. We also regard A as the set of all items in the sequence A. Rikka wants to find m sequences $S_1, S_2, \ldots, S_m$ such that $\cup_{i=1}^{m} S_i = \{0, 1, \ldots, n - 1\}$, $\sum_{i=1}^{m} |S_i| = n$, and for $i = 1, 2, \ldots, m$, $|S_i|$ is a multiple of k and there exists x, y ($x, y \in \mathbb{Z}$) meeting the conditions:

- $0 \le x, y < |S_i|$

- $x \ne y$

- $x \equiv (y + 1) \pmod{|S_i|}$

- $t[S_{i_x}] = t[S_{i_y}] = $ "B"

Input

The first line contains an integer T ($1 \le T \le 10^5$), the number of test cases. Then T test cases follow.

The input format of each test case is as follows:

The first line contains three integers n, m, k ($1 \le n, m, k \le 10^6$), the length of the garland, the number of sub-garlands and the factor of sub-garlands' lengths.

The second line contains a string t of length n, where the i-th character is either "B" or "C", representing $t[i]$ ($i = 0, 1, \ldots, n - 1$), the type of the flower i.

It is guaranteed that the sum of n in all test cases is at most 10^6.

Output

Answer each test case in order. For each test case, the output format is as follows:

The first line contains a string "Yes" or "No" (without the quotation marks). Output "Yes" if there exists a valid partition, or "No" otherwise.

If the answer is "Yes", output the sub-garlands in the following m lines. In each line, the first integer is the length of that sub-garland $|S_i|$. The following $|S_i|$ integers are the indices in the original garland of the flowers in it, in ascending order.

Example

standard input	standard output
4	Yes
6 2 2	2 0 1
BBCCBB	4 2 3 4 5
6 2 3	Yes
BBCCBB	3 0 1 2
4 4 1	3 3 4 5
BBBB	No
12 2 3	Yes
BBCCBCCBBCCC	6 0 1 2 3 5 6
	6 4 7 8 9 10 11

Problem H. Saintly . . . Coins

Input file:	standard input
Output file:	standard output
Time limit:	5 seconds
Memory limit:	512 megabytes

Rikka found a futuristic coat covering her body when waking up.

"It's cold at a high place like here."

It is the sound from the bankrupt boy standing there.

"Don't show that face... 'Bankrupt' doesn't mean that for me. I do have run out of my money, but I still have a job... And nobody can pull me out of my chair in the net bar for debts. "

Rikka knows that he has managed to smile to her. She is too weak, inexperienced and naive to say anything, but at least she could do something... The clashing coins remind her.

"Do you know 'Coin Fighting'? It used to be popular on the earth..."

"I'm a lunar native."

"Sorry... But I thought it may be... Good for you now..." Rikka's voice is reducing. Maybe her words hurt the boy who must be extremely sensitive to "coins" now.

"What's its rule?"

...

"Thank you, earthling." He reached his hand out and little lovely Rikka caught it, with thanks to and for both young juveniles. "We will meet again at the crossing of our fate."

"Jyaou Shingan Wa Saikou Desu!"

There are m coin stacks. Each of them has n coins initially, but their capacities are unlimited. They are lined up, forming an $n \times m$ grid as columns. Let (i, j) be the i-th position from bottom to top in the j-th stack from left to right. A position (x, y) is *less than* (x', y') if and only if $x < x'$ or $(x = x', y < y')$.

There are two categories of coins: ordinary ones and special ones. Ordinary coins are divided into 6 types by their denominations: $\{1, 5, 10, 50, 100, 500\}$; special coins are divided into two types: $\{X, Y\}$. Hence there are 8 types of coins in total.

Each ordinary coin is either *inactive* or *active*, while special coins are **always** *inactive* even if they should be *activated*.

Ordinary coins can be merged into a higher level or removed by some rules. Each denomination has a basic merging threshold, of $\{5, 2, 5, 2, 5, 2\}$ respectively. If a *four-connected component* (explained in the section "Notes", the same below) of the same denomination contains at least one *active* coin, and its size is not less than the corresponding threshold, then it is *removable*.

While removing a *removable* component, one can get a score increment of its size. Besides, if the same denomination of the removed coins is less than 500, **one** additional *active* coin of the denomination one level higher will be generated and placed at the *least* position among the places of the removed coins. Nothing will be placed when the same denomination of the removing coins is 500. Notice that no matter how many coins are removed, no more than one coin will be placed.

A game is made up of some *rounds*. Each *round* is described in order as follows:

1. *Inactivate* all coins at the beginning of the *round*.

2. Choose a coin-type t. Then all coins of type t at the top of a stack will be *picked*; do this in each stack repeatedly until nothing changes. Let $c(t)$ be the number of *picked* coins when choosing type t. An ordinary type t can be chosen only if $c(t) \geq 1$, while a special type t must satisfy $c(t) \geq 2$ for choosing.

- If t is ordinary, a stack x should be chosen and the *picked* coins will be put at the top of the stack immediately and *activated*.

- If t is special, a non-empty stack x with the top of an ordinary coin should be chosen, which also means one cannot choose the type t in the previous step if there is no such stack x. Let y be the type of the coin at the top of the chosen stack.

  Then, in case that t is special, the *picked* coins will be removed and thus the player will get a score increment of their amount.

  Meanwhile, all coins of type y will be removed immediately and thus the player will get a score increment of their amount.

  After that, if t is the type X and the denomination of y is less than 500, *active* coins of the denomination one level higher than y will be generated and take the places of all the coins of type y just removed.

3. *Activate* each *inactive* coin whose position is *four-adjacent* to at least one position where a coin was just removed (in step 2 and step 6, including the positions where high-level coins have been placed).

4. Coins fall down. That is, if a coin is *over* an adjacent blank position, it will be moved there; do this repeatedly until nothing changes. Obviously, the final status does not depend on the order of moving.

5. *Inactivate* each coin which is not in any *removable* component.

6. Remove all *removable* components **together**, get their score increments and place additional generated coins. If nothing could be removed, then the *round* ends, or jump to step 3 otherwise.

As mentioned above, a valid *operation* is determined by (t, x), the type and the stack chosen, respectively. A valid *operation* must gain at least a score incerment of 1. If there is no valid *operation* for a *round*, then the game ends.

In each round, the boy chooses a *best* operation among all valid ones. For two different valid *operations* $o_1 = (t_1, x_1), o_2 = (t_2, x_2)$, he compares them as follows:

1. If the categories of t_1 and t_2 are different, the special one is *better* than the ordinary one.

2. In case of a tie, if the increments of scores are different, the higher one is *better* than the lower one.

3. In case that a tie still remains, if t_1 and t_2 are different, the one whose type is appeared at left in $\{X, Y, 1, 10, 100, 5, 50, 500\}$ is *better*.

4. In other cases, the one with smaller x is *better*.

Could you please play a referee for them?

Given the initial stacks, please implement the boy's *operations*. You need to calculate the total number of *rounds*. And for each *round*, the type and the stack chosen are wondered, and so is the score he gets.

Input

The first line contains two integers n, m ($1 \le n \le 2000, 1 \le m \le 20$), the initial number of coins in each stack and the number of stacks, respectively.

Each of the following n lines contains a string consisting of $\{A, B, C, D, E, F, X, Y\}$, where characters $\{A, B, C, D, E, F\}$ refers to denominations $\{1, 5, 10, 50, 100, 500\}$ respectively, and the j-th character in the i-th line describes the type of the coin at (i, j). Notice that the character matrix is upside down for input.

It is guaranteed that there are at most 20 special coins (i.e. the coins of types X and Y).

Output

Output a single integer K at the first line, the total number of rounds.

Each of the following K lines contains a character and two integers. The character describes the type (in $\{A, B, C, D, E, F, X, Y\}$, as the input) which he chooses in that *round*, and the integers are the index of the chosen stack and the score increment in that *round*, respectively.

Examples

standard input	standard output
6 3 EEF EED ACC AAC AAC AAC	3 A 1 12 D 2 7 F 2 2
6 3 EEF EED ABC AAC AAC XAX	2 X 3 22 F 2 2
6 3 EEF EED ACC AAC AAC YAY	1 Y 3 12
7 2 EE FF FA AD DC EE YY	2 Y 1 9 D 1 2

Note

Two positions $(x_1, y_1), (x_2, y_2)$ are *four-adjacent* if and only if $|x_1 - x_2| + |y_1 - y_2| = 1$.

Two positions a, b are *four-connected* if and only if there exists a sequence of positions $a, c, \ldots, b$ beginning with a and ending with b such that every two adjacent positions in the sequence are *four-adjacent*.

A set of coins S is *four-connected* if and only if positions of every two coins in it are *four-connected*.

A *four-connected component* is a *four-connected* set of coins such that there doesn't exist a *four-connected* proper superset of it.

A position (x_1, y_1) is *over* (x_2, y_2) if and only if $x_1 = x_2 + 1, y_1 = y_2$.

In step 2, when you *pick* a coin, it is moved out from the stacks and put into a buffer.

Problem I. Misunderstood . . . Missing

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

Warm sunshine, cool wind and a fine day, while the girl watching is pursuing in chaos. Rikka reached out her hand and got the garland on her head, finding LCR with the immortal smile. The dream ended up waking, but the doubts will never disappear. In the end, without knowing about LCR, Rikka was invited to Shuyuan Men, a street of Chinese traditional arts in Xi'an.

"Is it enough to use the stored wires?"

"No problem... Those leaders are only concerned about expanding EC Final for the school's and their 'achievements'. All chores are ours. It is fine to simply connect those wiring boards in the series for each row."

Their conversation engaged Rikka. Feeling strange, she decided to follow them. But before all, she needs to beat the devil in her heart.

Rikka has an aggressivity A and an increment D of it, which are both 0 initially. There are n rounds in total. For $i = 1, 2, \ldots, n$, at the beginning of i-th round Rikka's aggressivity A increases by the increment D, and then she can do one of the following:

1. Attack and cause a damage of $(A + a_i)$.

2. Use the *Omnipotent Garland* from LCR to increase the increment D by b_i.

3. Use her *Schwarz Sechs Prototype Mark II* to increase the aggressivity A by c_i.

Rikka wonders the maximal possible damage she could cause in total. Could you help her?

Input

The first line contains a single integer T ($1 \le T \le 10$), the number of test cases. Then T test cases follow.

The input format of each test case is as follows:

The first line contains a single integer n ($1 \le n \le 100$), the number of rounds.

The following n lines contain $\{a_i\}, \{b_i\}, \{c_i\}$ for $i = 1, 2, \ldots, n$. The i-th line among them contains three integers a_i, b_i, c_i ($1 \le a_i, b_i, c_i \le 10^9$) separated by spaces in order.

It is guaranteed that the sum of n in all test cases is at most 100.

Output

Output T lines; each line contains one integer, the answer to that test case.

Example

standard input	standard output
3	6
2	10
3 1 2	24
3 1 2	
3	
3 1 2	
3 1 2	
3 1 2	
5	
3 1 2	
3 1 2	
3 1 2	
3 1 2	
3 1 2	

Problem J. Philosophical . . . Balance

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

Some power influenced Rikka to stand LCR up. Is it a coincidence that LCR brought her to the middle school attached to the EC Final host?

Rikka is trying to enter the NWPU, but the guard who has been possessed by the devil doesn't let her in. Now she has an idea of forging a pass.

The key is the pass ID, a special string for access-handshaking. The guard has a string s of length n as the secret key; when he checks a pass, he chooses a suffix of it (that is, a substring containing the last character), and calculates the length l of the longest common prefix of the pass ID and the suffix. l is proportion to the probability to let her pass.

Now Rikka has got the secret key in some secret way. She wants to choose a suffix as the pass ID, too. Since the suffix which the guard will choose is unknown, Rikka would choose her pass ID randomly. That is, Rikka would design a probability distribution for the suffixes (i.e. a series of real numbers $\{p_i\}, i = 1, 2, \ldots, n$, such that $p_i \geq 0, \sum_{i=1}^{n} p_i = 1$, which means she would choose the suffix of length i, denoted by s_i, with a probability of p_i), and maximize the minimum mathematical expectation of l for any suffix the guard chooses.

Could you please calculate the maximum value of the minimum mathematical expectation of l? Precisely speaking, what you should calculate is

$$\max_{\{p_i\}} \left(\min_{j=1}^{n} \left(\sum_{k=1}^{n} p_k \mathrm{lcp}(s_k, s_j) \right) \right),$$

where $\mathrm{lcp}(s_k, s_j)$ means the length of longest common prefix of s_k and s_j.

Input

The first line contains one integer T ($1 \leq T \leq 10^5$), the number of test cases. Then T test cases follow.

Each test case contains a string s of length n ($1 \leq n \leq 2 \times 10^5$), consisting of only lowercase letters, the secret key.

It is guaranteed that the sum of n in all test cases is at most 5×10^5.

Output

Output T lines; each line contains a decimal number, the answer to that test case.

Your answer is considered correct if the absolute or relative error doesn't exceed 10^{-9}.

Example

standard input	standard output
3	0.66666666667
aba	1
aaaaaaaaaaa	0.48
abcd	

Problem K. Desperate . . . Fire Survive

Input file:	`standard input`
Output file:	`standard output`
Time limit:	3 seconds
Memory limit:	256 megabytes

Rikka is running with her every cell to the hall where the contest will be held.

Well, EC Final is growing bigger and bigger... Thus more and more computers have been connected into the weak 2000W main wires. Heat is accumulating in every row.

When Rikka enters the hall, she finds LCR there rearranging the wires with the volunteers. However, the girl might not be able to do anything helpful but observing.

"Listen. We have no time to calculate parameters for the new circuit. Are you good at data structures? Help us, please..."

The circuitry is a sequence of n nodes, where there are m possible levels of nodes in total, numbered from 1 to m. Since a k-level node has a power limit twice of a $(k-1)$-level one, Rikka could merge two adjacent k-level nodes into a $(k+1)$-level one if $k < m$. She can also remove any node at any time from the circuitry while the order of rest nodes remains.

The volunteers have q queries in total. Each query contains a segment $[l, r]$ and an integer level k. Rikka needs to count how many sub-segments of the assigned segment (i.e. a segment $[x, y]$ such that $l \le x \le y \le r$) can provide a k-level node, which means it is possible to turn circuitry sequence $[x, y]$ into a single k-level node by merging adjacent nodes at the same level and removing nodes, where these two types of operations can be performed multiple times in any order. Notice that the level must be exactly k rather than higher or lower.

Input

The first line contains three integers n, m, q $(1 \le n, m, q \le 2 \times 10^5)$, the length of the circuitry sequence, the maximal level and the number of queries, respectively.

The second line contains n integers $A_1, A_2, \ldots, A_n$ $(1 \le A_i \le m)$, the levels of nodes in order.

Each of the following q lines contains three integers l, r, k $(1 \le l \le r \le n, 1 \le k \le m)$, describing a query.

Multiple integers in the same line are separated by spaces.

Output

Output q lines; each contains one integer, the answer to that query.

Examples

standard input	standard output
5 5 5 1 1 1 1 1 1 5 1 1 5 2 1 5 3 1 5 4 1 5 5	15 10 3 0 0
5 5 5 4 3 2 1 1 1 5 1 1 5 2 1 5 3 1 5 4 1 5 5	9 10 9 6 1

Problem L. Eventual . . . Journey

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

LCR is really an incredible being.

Thinking so, sitting on the plane and looking at the sea, Rikka savours the fantastic journey. A fire never happened. It must be interesting to share it with her mates, and another travel is also fascinating.

But before all, home is the best.

She travels by the JR lines. There are n stations in total, and m public bidirectional railway lines are built among them. Each station belongs to either JR West or JR East. Both JR West and JR East have their own private railways connecting all stations owned by themselves.

Rikka has some through tickets and two types of special passes: ICOCA for JR West and Suica for JR East. She pays a through ticket each time and does one of the following:

1. Travel from one terminal to another via a public railway line.

2. Travel to any station which has the same owner as the current one, using one of her special passes. A pass can be used for multiple times.

Rikka wonders, for each start station, the sum of the minimal numbers of tickets she needs to pay to reach every possible one of the other stations.

Input

The first line contains two integers n, m ($1 \le n \le 10^5, 0 \le m \le 10^5$), the numbers of stations and public railways.

The next line contains n integers A_i ($A_i \in \{0, 1\}, i = 1, 2, \ldots, n$), separated by spaces, describing the owner of each station. $A_i = 0$ if the station i belongs to JR west, and vice versa.

The following m lines describe all the public railways, each of which contains two integers u, v ($1 \le u, v \le n, u \ne v$), representing a bidirectional railway connecting u and v. It is guaranteed that no two public railways connect the same pair of stations, and Rikka is able to travel between every pair of stations. Notice that the private railways are not given directly in the input, and Rikka may have to use her passes rather than traveling only through the public railways.

Output

Output n integers in one line, separated by spaces. The i-th integer is $\sum_{j=1}^{n} D(i, j)$, where $D(u, v)$ is the minimal number of tickets needed to travel from u to v.

Examples

standard input	standard output
3 2 1 0 0 1 2 1 3	2 2 2
5 3 1 0 1 0 1 1 2 2 3 4 5	5 5 5 6 5