Problem A. Ariel

Input file:	`stdin`
Output file:	`stdout`
Time limit:	2 seconds
Memory limit:	256 megabytes

King Triton really likes watching sport competitions on TV. But much more Triton likes watching live competitions. So Triton decides to set up a swimming competition in the kingdom Merfolk. Thousands of creatures come to take part in competition, that's why it is too difficult to take the first place. For the King's beloved daughter Ariel this competition is the first in her life. Ariel is very kind, so she wants to give a lot of gold medals. Ariel says, that it is unfair to make a single ranking list for creatures that are so different. It is really a good result to be the fastest small fish without tail in Merfolk! Ariel chooses k important traits (such as size, tailness, rapacity and so on). A creature can either possess a trait or not (there are no intermediate options). A score is given for each creature (it doesn't matter how it was calculated) and the list of possessed traits $f_1, \ldots, f_y$ is also given. Ariel wants to know the place occupied by creature a in a competition among creatures, who have the same traits $h_1, \ldots, h_t$. So if creature a doesn't have a trait h_i, then all creatures in the competition are without this trait. If creature a has a trait h_i, then all creatures in the competition have this trait. Other traits doesn't matter. The winner of the competition is a creature with the maximum score.

Input

The first line contains n ($1 \le n \le 10^4$) and k ($1 \le k \le 10$). The next n lines contain information about creatures: $score$ ($1 \le score \le 10^9$), y ($0 \le y \le k$) — the number of possessed traits, and y numbers f_i ($1 \le f_i \le k$) — ids of possessed traits. All f_i in one line are different. The next line contains m ($1 \le m \le 10^5$) — the number of queries from Ariel. The next m lines describe queries: a ($1 \le a \le n$) — the id of a creature, then t — the number of traits, then t numbers h_i. All h_i in one line are different.

Output

For each query output the place of a creature a in ranking list amount the corresponded creatures. If several creatures have the same score all of them take the same place.

Examples

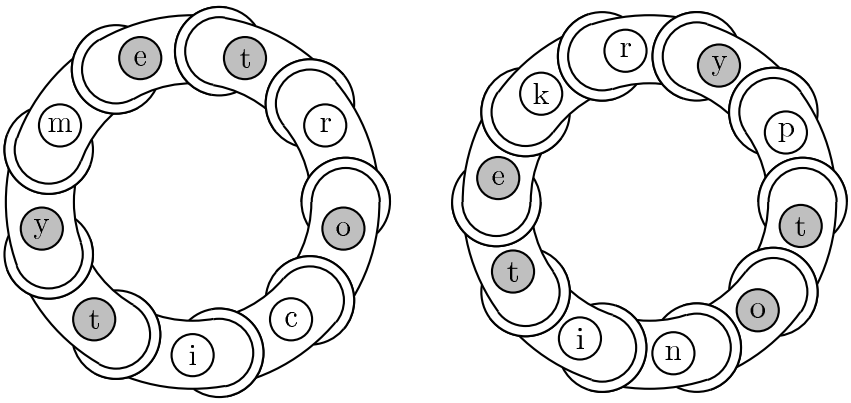
stdin	stdout
3 2 100 1 1 50 1 2 30 2 1 2 12 1 2 1 2 1 1 1 1 1 2 1 0 2 0 2 1 1 2 1 2 2 2 2 1 3 0 3 2 1 2 3 1 2 3 1 1	1 1 1 1 2 1 1 1 3 1 2 2
3 2 100 0 10 0 100 0 3 1 0 2 0 3 0	1 3 1

Problem B. Bracelets

Input file:	`stdin`
Output file:	`stdout`
Time limit:	6 seconds
Memory limit:	256 megabytes

Finally, Megamind has devised the perfect plan to take down his arch-nemesis, Metro Man! Megamind has designed a pair of circular power bracelets to be worn on his left and right wrists. On each bracelet, he has inscribed a sequence of magical glyphs (symbols); each activated glyph augments Megamind's strength by the might of one grizzly bear!

However, there's a catch: the bracelets only work when the subsequences of glyphs activated on each bracelet are identical. For example, given a pair of bracelets whose glyphs are represented by the strings "metrocity" and "kryptonite", then the optimal activation of glyphs would give Megamind the power of 10 grizzly bears:



On the first bracelet, the letters "etoty" are activated in clockwise order; the same letters are activated in counterclockwise order on the second bracelet. Generally, the ordering of the letters is important, but the orientation of the activated subsequence on each bracelet (i.e., clockwise or counterclockwise) may or may not be the same—and don't forget that the bracelets are circular!

Help Megamind defeat Metro Man by determining the optimal subsequences of glyphs needed to activate his bracelets.

Input

The input file will contain one space-separated pair of strings s and t, corresponding to the sequences of glyphs on Megamind's left and right power bracelets, respectively. Each string will consist of only lowercase letters ('a'-'z'). The length of each input string will be between 1 and 1500 characters, inclusive.

Output

Print a single integer: the maximum power (in units of grizzly bears) that Megamind will be able to achieve by activating glyphs on his bracelets.

Examples

stdin	stdout
metrocity kryptonite	10
megamind agemdnim	16
metroman manmetro	16
megamindandmetroman metromanandmegamind	32

Problem C. Cow run

Input file:	stdin
Output file:	stdout
Time limit:	2 seconds
Memory limit:	64 megabytes

Farmer John and Bessie have devised a new exercise game for the cows. The cows are running on a circular track of length M ($2 \leq M \leq 10^9$) starting from the same position. The game proceeds in N ($1 \leq N \leq 14$) rounds using a deck of $8N$ cards each with a number X_i ($0 \leq X_i < M$) written on it.

Each round FJ moves the top 8 cards into a separate pile and selects either the top 4 or bottom 4 cards for Bessie to play with. Bessie then chooses either the top 2 cards or bottom 2 cards of the 4 cards FJ selected. After this FJ calls out the number on the top card, X_{top}, and the cows run a distance of $R \cdot X_{top}$, where R is the total distance the cows have run so far. Bessie then calls out the number on the bottom card, X_{bottom}, and the cows run a distance of X_{bottom}.

FJ is concerned that after the exercise the cows will be too tired to get back to the beginning of the track if they end up too far away. He believes if the cows end up more than a distance of K ($0 \leq K \leq \lfloor M/2 \rfloor$) from their starting position they won't be able to get back home.

It is guaranteed that if FJ plays correctly, he will always be able to ensure the cows can come home, irrespective of the moves made by Bessie! For each round, your task is to determine which half of the cards FJ should choose such that no matter what Bessie does from that point on, FJ can always get the cows home. Bessie will then make the move provided in the input and you can then continue onto the next round. Note that even though Bessie's moves are provided to you in the input, you are to specify moves for FJ that would have worked no matter what Bessie chooses (so it is effectively as if FJ does not really know what Bessie will do during her moves).

Input

Line 1: Three space-separated integers N, M, K

Line 2: A string of N characters. If the i-th character is 'T', it means Bessie will select the top 2 cards in the i-th round. Otherwise, the i-th character will be 'B' to indicate Bessie will select the bottom 2 cards in the i-th round.

Lines 3..$N+2$: Each line contains eight integers representing the 8 cards to be used that round from top to bottom.

Output

A string of N characters where the i-th character is 'T' if FJ should choose the top 4 cards or 'B' if FJ should choose the bottom 4 cards in the ith round. If there are multiple ways to get the cows home, choose the lexicographically first (that is, the string that is alphabetically smallest).

Examples

stdin	stdout
2 2 0 TT 1 0 0 0 0 0 0 1 0 1 1 1 0 0 1 0	TB

Note

The cows must end up exactly where they started to be able to come home. Note that FJ is not aware of what choices Bessie is going to make beforehand. If FJ did know, he could have chosen the bottom half each time.

Problem D. Different vectors

Input file:	stdin
Output file:	stdout
Time limit:	1 second
Memory limit:	64 megabytes

You are given a set of N vectors, each vector consists of K integers. Vector X is equivalent to Y (denoted $X \equiv Y$) iff there exist a bijection $f : \mathbb{Z} \to \mathbb{Z}$ and an integer r, such that $X[i] = f(Y[(i + r) \bmod K])$ for each i in the range $[0..K - 1]$.

For example, $(1, 2, 2, 3) \equiv (22, 3, 4, 22)$, with $r = 2$ and $f(22) = 2$, $f(3) = 3$ and $f(4) = 1$. But $(22, 3, 22, 4)$ is not equivalent to $(1, 2, 2, 3)$.

How many pairwise nonequivalent vectors are there in a given set of N vectors?

Input

First number contains T ($T \leq 10$), number of test cases. Each test case consists of the following. First line consists of N and K ($1 \leq N \leq 10000$, $1 \leq K \leq 100$). N lines follow, the i-th containing K integers describing the i-th vector. The vector values are from the range $[0, 10^9]$.

Output

Output one number: the number of different vectors.

Examples

stdin	stdout
2 3 4 22 3 4 22 1 2 2 3 22 3 22 4 5 5 3 3 3 0 3 8 4 4 4 0 1 1 1 1 1 1 1 8 6 1 1 3 3 3 5	2 3

Problem E. bits-Equalizer

Input file:	`stdin`
Output file:	`stdout`
Time limit:	2 seconds
Memory limit:	64 megabytes

You are given two non-empty strings S and T of equal lengths. S contains the characters 0, 1 and ?, whereas T contains 0 and 1 only. Your task is to convert S into T in minimum number of moves. In each move, you can:

1. change a 0 in S to 1
2. change a ? in S to 0 or 1
3. swap any two characters in S

As an example, suppose $S = 01??00$ and $T = 001010$. We can transform S into T in 3 moves:

- Initially S = 01??00
- Move 1 – change $S[2]$ to 1. S becomes 011?00
- Move 2 – change $S[3]$ to 0. S becomes 011000
- Move 3 – swap $S[1]$ with $S[4]$. S becomes 001010
- S is now equal to T

Input

The first line of input is an integer C ($C \leq 200$) that indicates the number of test cases. Each case consists of two lines. The first line is the string S consisting of '0', '1' and '?'. The second line is the string T consisting of '0' and '1'. The lengths of the strings won't be larger than 100.

Output

For each case, output the case number first followed by the minimum number of moves required to convert S into T. If the transition is impossible, output -1 instead.

Examples

stdin	stdout
3 01??00 001010 01 10 110001 000000	Case 1: 3 Case 2: 1 Case 3: -1

Problem F. Find the sequence

Input file:	`stdin`
Output file:	`stdout`
Time limit:	1 second
Memory limit:	64 megabytes

Mislav and Marko have devised a new game, creatively named Rotate. First, Marko imagines a number sequence of length N and divides it into sections, with each section containing K numbers (K evenly divides N). The first section contains numbers in the first K positions in the sequence, the second section the following K positions, and so on.

Then, Marko asks Mislav to apply a number of operations on the sequence, with each operation being one of the following two types:

1. Rotate the numbers in each section to the left/right by X positions
2. Rotate the whole sequence to the left/right by X positions

Notice that an operation of type 2 can change the numbers belonging to each section. After applying all the operations, Mislav reveals the final sequence to Marko. Marko's task is finding Mislav's starting sequence. He has asked you for help.

Input

The first line of input contains three positive integers: N ($1 \leq N \leq 10^5$), the length of the sequence, K ($1 \leq K \leq 10^5$), the size of each section, and Q ($1 \leq Q \leq 10^5$), the number of operations.

Each of the following Q lines contains two integers: A ($1 \leq A \leq 2$), the operation type, and X ($-10^5 \leq X \leq 10^5$), the number of positions to rotate by. A negative number represents rotation to the left, while a positive one represents rotation to the right.

The last line of input contains N space-separated integers Z_i ($0 \leq Z_i \leq 10^5$) representing the final sequence (after applying all operations).

Output

The first and only line of output must contain the required starting sequence.

Examples

stdin	stdout
4 2 2 2 2 1 1 3 2 1 0	0 1 2 3
8 4 4 1 3 1 15 1 -5 2 -1 6 10 14 19 2 16 17 1	6 10 14 1 2 16 17 19
9 3 5 1 1 2 -8 2 9 1 1 2 -4 3 1 8 7 4 5 2 6 9	5 3 6 9 7 1 8 2 4

Note

Clarification of the first example: the starting sequence is `0 1 2 3`. After the first operation, the sequence is `2 3 0 1`, and after the second operation, it becomes `3 2 1 0`. Ths corresponds to the final sequence.

Problem G. Good elements

Input file:	`stdin`
Output file:	`stdout`
Time limit:	1 second
Memory limit:	64 megabytes

You are given a sequence A consisting of N integers. We will call the i-th element **good** if it equals the sum of some three elements in positions strictly smaller than i (an element can be used more than once in the sum).

How many good elements does the sequence contain?

Input

The first line of input contains the positive integer N ($1 \leq N \leq 5000$), the length of the sequence A.

The second line of input contains N space-separated integers representing the sequence A ($-10^5 \leq A_i \leq 10^5$).

Output

The first and only line of output must contain the number of good elements in the sequence.

Examples

stdin	stdout
2 1 3	1
6 1 2 3 5 7 10	4
3 -1 2 0	1

Problem H. Highways

Input file:	`stdin`
Output file:	`stdout`
Time limit:	1 second
Memory limit:	64 megabytes

The island nation of Flatopia is perfectly flat. Unfortunately, Flatopia has a very poor system of public highways. The Flatopian government is aware of this problem and has already constructed a number of highways connecting some of the most important towns. However, there are still some towns that you can't reach via a highway. It is necessary to build more highways so that it will be possible to drive between any pair of towns without leaving the highway system.

Flatopian towns are numbered from 1 to N and town i has a position given by the Cartesian coordinates (x_i, y_i). Each highway connects exactly two towns. All highways (both the original ones and the ones that are to be built) follow straight lines, and thus their length is equal to Cartesian distance between towns. All highways can be used in both directions. Highways can freely cross each other, but a driver can only switch between highways at a town that is located at the end of both highways.

The Flatopian government wants to minimize the cost of building new highways. However, they want to guarantee that every town is highway-reachable from every other town. Since Flatopia is so flat, the cost of a highway is always proportional to its length. Thus, the least expensive highway system will be the one that minimizes the total highways length.

Input

The input consists of two parts. The first part describes all towns in the country, and the second part describes all of the highways that have already been built.

The first line of the input contains a single integer N ($1 \leq N \leq 750$), representing the number of towns. The next N lines each contain two integers, xi and yi separated by a space. These values give the coordinates of ith town (for i from 1 to N). Coordinates will have an absolute value no greater than 10000. Every town has a unique location.

The next line contains a single integer M ($0 \leq M \leq 1000$), representing the number of existing highways. The next M lines each contain a pair of integers separated by a space. These two integers give a pair of town numbers which are already connected by a highway. Each pair of towns is connected by at most one direct highway.

Output

Write to the standard output a single line for each new highway that should be built in order to connect all towns with minimal possible total length of new highways. Each highway should be presented by printing town numbers that this highway connects, separated by a space.

If no new highways need to be built (all towns are already connected), then the output should be empty.

Examples

stdin	stdout
9 1 5 0 0 3 2 4 5 5 1 0 4 5 2 1 2 5 3 3 1 3 9 7 1 2	1 6 4 9 5 7 8 3 7 3

Problem I. I WIN

Input file:	`stdin`
Output file:	`stdout`
Time limit:	1 second
Memory limit:	256 megabytes

Given an $n \times m$ rectangular tile with each square marked with one of the letters W, I, and N, find the maximal number of triominoes that can be cut from this tile such that the triomino has W and N on the ends and I in the middle (that is, it spells WIN in some order). Of course the only possible triominoes are the one with three squares in a straight line and the L-shaped ones, and the triominoes can't overlap.

Input

First line contains two integers n and m with $1 \leq m, n \leq 22$. The next n lines contain m characters each (only the letters W, I and N).

Output

Output a single integer: the maximum number of nonoverlapping WIN-triominoes.

Examples

stdin	stdout
4 4 WIIW NNNN IINN WWWI	5
5 5 NINWN INIWI WWWIW NNNNN IWINN	5

Problem J. Journeys on the Moscow Underground

Input file:	stdin
Output file:	stdout
Time limit:	2 seconds
Memory limit:	256 megabytes

World's economy depression of the 2008 affected everybody in the world and harmed a lot of businesses.

But Rich Scrooge McDuck's business suffered most of all. His gold factories started yielding a loss and even went bankrupt. Moreover, the US government revealed that Scrooge was involved in illegal mortgage practices. He had no other option but to leave USA and go abroad. You know what? He chose Russia!

– Uncle Scrooge, we want to try Moscow underground, buy tickets for us! — said Billy and Willy, who had come on their holidays to visit their poor uncle.

– Okay, wait a moment, please — uncle Scrooge answered pensively. He tried to figure out how to spend the least possible money and please his nephews at the same time.

Since Scrooge has never liked mathematics, he needs your help. It is known that Billy and Willy stay in Moscow for n days. They like underground and want to spend some of days to explore it (one trip a day). All the rest days they are going to visit Gorky Park. To prevent squabbles between Billy and Willy, Scrooge needs to give them separate tickets even if they go to underground together. At the end of each day nephews should return their tickets back to uncle. Scrooge has special relations in Moscow underground and can buy tickets for A passages and B days cheaper than they are sold to ordinary people. Certainly, he wants to buy minimum possible number of tickets, so he needs to determine which tickets to give Billy and Willy in the morning. You are hired to help McDuck solve this tricky problem!

Note that the single ticket allows you to use underground no more than A times and the number of days between the first and the last passage should be less than B.

Input

In the first line of the input there is an integer n ($1 \le n \le 200$) and integers A and B ($1 \le A, B \le 20$). The second and third lines contain numbers $a_1, a_2, \ldots, a_n$ and $b_1, b_2, \ldots, b_n$, respectively ($a_i, b_i \in \{0, 1\}$). Billy uses the underground on the i-th day if and only if $a_i = 1$. Willy uses the underground on the i-th day if and only if $b_i = 1$.

Output

Output one number — the minimum number of tickets that Scrooge should buy for Billy and Willy.

Examples

stdin	stdout
2 5 5 1 1 0 0	1
2 5 5 1 0 0 1	1
11 20 10 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0	3