University of Engineering and Technology

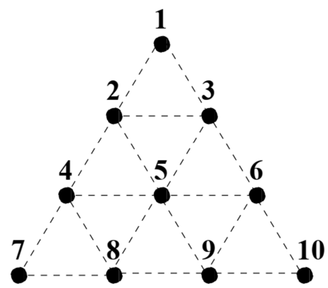
ICPC 2022

Northern Round

16-10-2022

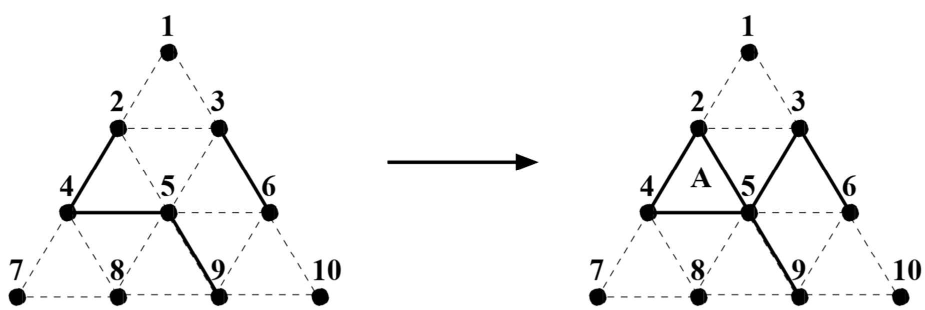
A. TWAR

T-War is a game that 2 competitors played on the following triangular grid:



Supposed A and B, take turns filling in any dotted line connecting two dots, A starting first. Each dotted line is filled only once. If the line filled by a player completes one or more triangles, he owns the completed triangles and is awarded another turn. The game ends after all dotted lines are filled in, and the competitor with the most triangles wins. The diference in the number of triangles owned by the two players is not important.

For example, if A fills in the line between 2 and 5 in the partial game on the left below:



Then, she owns the triangle labelled A and takes next turn to fill in the line between 3 and 5. B can now own 3 triangles (if he wishes) by filling in the line between 2 and 3, then the one between 5 and 6, and finally the one between 6 and 9. B would then make one more move before it is A's turn again.

Now you are given a number of moves that have already been made. From the partial game, you should determine which player will win assuming that each player plays a perfect game from that point on.

INPUT

In the first line, you will be given a number of tests. For each test: the first line is an integer $6 \leq m \leq 18$ indicating the number of moves that have been made in the game. The next m lines indicate the moves made by the two players in order, each of the form i j (with $i < j$) indicating that the line between i and j is filled in that move. You may assume that all given moves are legal.

OUTPUT

For each game, print in one line the game number and the result as shown below. If A wins, print the sentence “A wins.". If B wins, print “B wins."

Sample Input	Sample Output
2 6 2 4 4 5 5 9 3 6 2 5 3 5 7 2 4 4 5 5 9 3 6 2 5 3 5 7 8	Game 1: B wins. Game 2: A wins.

B. MOVE

You have a number **m**, a starting number $\mathbf{s_0}$ and two sequences of numbers a_i and b_i. Your goal is to go from s_0 to 0 in as few moves as possible. In each move, you choose an i, then multiply your current number by a_i, add b_i to it, and reduce the result modulo m. That is

$s_j = (s_{j-1} * a_{ij} + b_{ij}) \% m$.

INPUT

The first line of input contains three integers $0 < m \leq 1000000$, $0 \leq n \leq 10$, and $0 < s_0 < m$. The next n lines each contain two integers, a pair $0 \leq a_i \leq 1000000000$ and $0 \leq b_i \leq 1000000000$.

OUTPUT

Output the shortest number of moves needed to reach 0 starting from s_0. If it is not possible to reach 0 in any number of moves, output -1.

Sample Input	Sample Output
5 2 1 2 1 3 1	2

C. VENDING

The university finally decided to install some popular vending machines at various strategic places in the campus. In fact, to compensate for the previous lack of machines, they decided to install as many machines as possible. Surprisingly enough, the campus is not going to be choked with new machines because there are some quite serious legal limitations regarding the locations of the machines. The university has marked all possible vending machine locations and their respective coordinates on the campus map. Additionally, they also have to respect manufacturer security rule: The distance between any two vending machines has to be at least 1.3 meters. Help the university to establish the maximum possible number of vending machines which can be installed in the campus.

INPUT

There are several test cases. Each test starts with a line containing one integer N which specifies the number of possible vending machine locations in the map ($1 \le N \le 2000$). Next, there are N lines representing the location coordinates, each line describes one location by a pair of integer coordinates in meters. All locations in one test case are unique. Each coordinate is non-negative and less than or equal to 10^9.

You are guaranteed that all locations form a single connected group, that is, it is possible to start in any location and reach any other location by a sequence of steps, each of which changes exactly one coordinate by 1, without leaving the area suitable for placing vending machines.

OUTPUT

For each test case, print a single line with one integer representing the maximum number of vending machines which can be installed in the campus.

Sample Input	Sample Output
4 0 0 0 1 1 0 1 1 6 0 1 1 0 1 1 1 2 1 3 2 2	2 4

D. MOD10

The function f(x) is defined as follows: If $0 \leq x \leq 9$, then $f(x) = x!$, and if $x > 9$, then $f(x) = (x \bmod 10)! + f(\lfloor x/10 \rfloor)$.

The brackets $\lfloor\rfloor$ denote the floor value of a number (e.g. $\lfloor 2.43 \rfloor = 2$). Exclamation mark denotes the factorial, i.e., $x! = 1 \times 2 \times \cdots \times x$ for $x > 0$ and $0! = 1$.

With a number y, you need to input smallest such non-negative integer x, so that $f(x) = y$ holds.

INPUT

The input consists of one integer y ($1 \leq y \leq 10^9$)

OUTPUT

Output a single non-negative integer x.

Sample Input	Sample Output
3	12
20	2333

E. FUNCTION

Consider the sequence f defined by the following recursive formula:

$$\begin{cases} f(0) = 1 \\ f(2n) = f(n) + f(n-1) & \text{for all positive integers n} \\ f(2n+1) = f(n) & \text{for all non} - \text{negative integers n} \end{cases}$$

Given two positive integers p and q, p, q $\leq 10^4$. Find a non-negative integer n such that:

$$\frac{f(n)}{f(n+1)} = \frac{p}{q}$$

If there existed no value of n, print -1.

If there are multiple satisfied values of n, print the smallest one.

INPUT

Each line of the input consists of two numbers p and q.

OUTPUT

For each pair of p and q, print the result as described above in one line.

Sample Input	Sample Output
2 3 1 4 5 5	5 7 0

EXPLANATION

$f(0) = 1, f(1) = 1, f(2) = 2, f(3) = 1, f(4) = 3, f(5) = 2, f(6) = 3, f(7) = 1, f(8) = 4$

F. GIFT

Tuan gives his son an unweighted directed graph as a birthday gift. The graph contains n vertices, the vertices are numbered from 1 to n. There are also m directed edges in the graph, numbered from 1 to m.

Tuan wants to test his son's IQ by playing a game. The game consists of Q rounds, in each round Tuan will ask his son to do solve of the two following questions:

1. Delete one edge from the graph
2. Find the length of the shortest path from vertex 1 to a vertex v

Tuan wants his son to answer each question of the second type immediately, without knowing any of the future questions.

Notes:

- A path of length t of the graph is a sequence of vertices $v_1, v_2, \ldots, v_t, v_{t+1}$ where there is an edge from v_i to v_{i+1} for every $1 \leq i \leq t$.
- A path of length t is the shortest path from a to b if there does not exist another path from a to b of length smaller than t.

INPUT

The first line in the input contains two integers n and m, the number of vertices and edges of the graph ($1 \leq n \leq 3500, 1 \leq m \leq 35000$).

The i-th of the following m lines contains two integers u_i and v_i where there is an directed edge from u_i to v_i.

The next line contains a single integer Q, the number of questions ($1 \leq Q \leq 5 \times 10$^5).

The remaining Q lines describe the Q questions, each will be in either of the 2 following forms:

- $1\ x$ – delete edge $id = (x + last)\ mod\ m + 1$, assuming $last$ is the previous type 2 question answer. For the first question of type 2, $last = 0$. And if there is no path from 1 to the queried vertex, $last$ = -1. It is guaranteed that each edge is deleted at most once
- $2\ v$ - ask the length of shortest path from vertex 1 to vertex v.

OUTPUT

For each question of the second type, print the answer in a single line. If there is no path from 1 to v in the question, print -1.

Sample Input	Sample Output
4 6 1 4 1 3 4 2 1 2 2 3 3 4 7 1 1 2 3 1 1 2 3 2 2 1 0	2 3 2 -1

2 3	

The real edges deleted are 2, 4, and then 3 sequentially

G. SONG

Thang is writing lyrics for his newly composed song. This time, he's come up with a criterion for writing lyrics. Let's denote string s (containing only lowercase alphabet letters) as the lyrics for the whole song. For each substring t, Thang defines:

- $length(t)$: the number of characters of t
- $frequency(t)$: the number of times t appears in s as a substring.
- $sum(t)$: the sum of value across all characters of t. The value of each letter, in this case, is its alphabet order ($'a' = 1, \dots, \ 'z' = 26$) .

Let's denote the set of unique substring of s as $U(s)$, Thang defines the beauty of lyrics s as:

$$beauty(s) = \sum_{t \in U(s)} length(t) \times frequency(t) \times sum(t)$$

You are to help Thang in composing his song. Thang is giving you a list of lyrics versions and you shall help him to calculate the beauty for each.

INPUT

The first line contains T ($T \leq 10$), the number of lyrics versions you are to calculate their beauty.

The i-th line of the next T lines contains the i-th lyrics version of the song s_i.

It is guaranteed that $\sum |s_i| \leq 5 \times 10^5$.

OUTPUT

Your program should output T lines, the i-th line should contains a positive integer representing the beauty of s_i in modulo $10^9 + 7$.

Sample Input	Sample Output
2 ab aba	9 28
3 thaychuathaychuaem cosaocosaodau toldyoutoldyouso	114760 32242 110823

H. MEDIAN

Mr Phuong is organizing a special "Competitive Programming" module to prepare his students for the upcoming ICPC.

At any time, Mr Phuong wants to calculate the median ability of his class so that he can have a good problem set that fits his students' abilities.

His class has N students, numbered 0 to N-1, with their ability a[i] in increasing order.

If N is odd, the median equals to a[N/2].

If N is even, the median equals to (a[N/2 - 1] + a[N/2]) / 2

There are Q events, each is in 1 of 3 types:

- IN x - a student with ability x joined
- OUT x - a student with ability x left the class
- MEDIAN - Mr Phuong wants to get the median value.

Your task is to help Mr Phuong answer his queries.

INPUT

The first line contains an integer Q - the number of queries.

Then Q lines followed, each describing a query in the above format.

In the beginning, the class has no member and you can assume that when you have query OUT x, there was a corresponding IN x before that.

$1 \leq Q \leq 2.10^5$, $1 \leq x \leq 10^9$

OUTPUT

For each median query, you should print a single number - the median at this time.

Sample Input	Sample Output
22 IN 1 MEDIAN IN 2 MEDIAN IN 3 MEDIAN IN 4 MEDIAN IN 5 MEDIAN OUT 1 MEDIAN IN 6 MEDIAN OUT 2 MEDIAN IN 7 MEDIAN OUT 5	1 1.5 2 2.5 3 3.5 4 4.5 5 5 6

MEDIAN IN 8 MEDIAN	

I. LIS

Leon is a hardworking student who spends his free time pondering over sequences of integers. Today, he is particularly interested in increasing sequences.

A subsequence is formed by removing zero or more elements from a sequence while retaining the order of the remaining elements. The longest increasing subsequence of a sequence is defined as the longest subsequence whose elements are strictly increasing. This subsequence may not necessarily be unique. For example, the longest increasing subsequence of the sequence (1, 2, 4, 3) can either be (1, 2, 3) or (1, 2, 4), and both subsequences have a length of 3.

Leon gives you the following problem:

Given a sequence X, let Y be a subsequence of X. What is the longest possible length of Y such that the length of the longest increasing subsequence of Y does not exceed K?

This problem was too easy for you, so Leon decides to ask you more questions. He starts by providing you with a starting sequence A which contains N integers. Then, he gives you Q questions. You are still going to solve the problem above, but sequence X and the integer K will vary between questions. He gives you the integer K directly, and he also gives you an integer M and says that sequence X is formed by taking the first M elements of sequence A.

For each question, provide the answer to the problem above for the given sequence X and the integer K.

INPUT

The first line contains two integers N and Q, where N is the number of elements in sequence A, and Q is the number of questions you must answer. ($1 \le N \le 50000$, $1 \le Q \le 200000$)

The second line contains N integers $A_1, A_2, A_3, \ldots, A_N$ - the elements of sequence A. ($1 \le A_i \le 50000$).

The next Q lines contain two integers M_i and K_i, meaning you must solve the problem above for the first M_i elements of A and the integer K_i ($1 \le K_i \le M_i \le N$).

OUTPUT

For each question, print one line containing the answer.

Sample Input	Sample Output
11 6 9 6 3 1 5 12 8 4 2 2 2 5 1 7 2 9 1 9 2 11 1 11 11	4 6 5 8 7 11

EXPLAINATION

- Question 1: For the sequence X = (9, 6, 3, 1, 5), one can choose the subsequence Y = (9, 6, 3, 1). The length of the longest increasing subsequence of Y is 1.
- Question 2: For the sequence X = (9, 6, 3, 1, 5, 12, 8), one can choose the subsequence Y = (9, 6, 3, 1, 12, 8). The length of the longest increasing subsequence of Y is 2.
- Question 3: For the sequence X = (9, 6, 3, 1, 5, 12, 8, 4, 2), one can choose the subsequence Y = (9, 6, 5, 4, 2). The length of the longest increasing subsequence of Y is 1.

- Question 4: For the sequence X = (9, 6, 3, 1, 5, 12, 8, 4, 2), one can choose the subsequence Y = (9, 6, 3, 1, 12, 8, 4, 2). The length of the longest increasing subsequence of Y is 2.
- Question 5: For the sequence X = (9, 6, 3, 1, 5, 12, 8, 4, 2, 2, 2), one can choose the subsequence Y = (9, 6, 5, 4, 2, 2, 2). The length of the longest increasing subsequence of Y is 1.
- Question 6: For the sequence X = (9, 6, 3, 1, 5, 12, 8, 4, 2, 2, 2), one can choose the subsequence Y = (9, 6, 3, 1, 5, 12, 8, 4, 2, 2, 2). The length of the longest increasing subsequence of Y is 3.

J. DOTS

Mika loves doodling in her notebook. Today, she's drawn N dots and connected them with N - 1 lines. These lines are drawn in a such a way that she can follow a path from one dot to another without having to lift her pencil.

The dots are numbered from 0 to N - 1. Mika wants to follow a path from dot 0 to some other dot without lifting her pencil. The path can contain a line more than once.

To make things challenging, she assigns N integers E_0, E_1, E_2, … $E_{(N - 1)}$ to each dot. For any dot i, she must ensure that her path does not leave that dot more than E_i times.

For each integer i from 0 to N - 1, determine the length of the longest path Mika can follow to get from dot 0 to dot i that satisfies the conditions above.

INPUT

The first line contains an integer N - the number of dots ($1 \leq N \leq 50000$).

The second line contains N integers E_0, E_1, E_2, …, $E_{(n - 1)}$ ($1 \leq E_i \leq 40000$), where E_i is the maximum number of times the path can leave dot i. It is guaranteed that E_i is greater than or equal to the number of lines that come out of dot i.

The next N - 1 lines contain two integers U_i and V_i ($0 \leq U_i, V_i \leq N - 1$), indicating that there is a line between dots Ui and Vi.

OUTPUT

For each integer i from 0 to N - 1, print one line containing the length of the longest path that ends at dot i.

Sample Input	Sample Output
3 2 6 2 0 1 1 2	8 7 8

K. BUBBLE TEA FANS

Hanh is a great fan of bubble tea – a well-known delicious drink in Asian countries! Consequently, all of his students really enjoy this tasty drink as well.

One day, Hanh holds a bubble tea party, where all of his students are invited. n students attend this event. At the beginning, Hanh asks them to stand in a row and numbers them from 1 to n, from left to right.

Each student has exactly one favourite brand of bubble tea – the bubble tea shop which he/she likes best. Then Hanh comes to several students and asks the same question: "How many students are there in this room (including you) sharing the same favourite brand with you?" He just asks only some of them, not everyone. Everyone who is asked tells Hanh the correct number.

After asking questions and receiving answers from students, Hanh wonders: Is it possible to infer the answers of all students who are not asked? He knows pretty sure the following fact about his students: Students who have the same favourite bubble tea brand always stand together. In other words, if the i^{th} student and the j^{th} student share the same brand, all k^{th} students such that $i \le k \le j$ also favour this brand.

Please help Hanh answer this question.

INPUT

The first line of the input contains an integer n $(1 \le n \le 100)$ denoting the number of students attending Hanh's party.

The second line of the input contains n integers $a_1, a_2, \dots, a_n$ $(0 \le a_i \le n)$ representing the information of these students. If $a_i = 0$, the i^{th} student is not asked, otherwise, a_i equals to his answer.

It is guaranteed that the data is consistent. In other words, there exists a scenario where all students' answers are correct.

OUTPUT

Print YES if Hanh can infer the answers of all students correctly and uniquely. Print NO otherwise.

Sample Input	Sample Output
7 3 0 0 0 4 0 0	YES
4 1 0 0 1	NO

EXPLANATION

In the first sample, the first student says that there are 3 students (including him/her) having the same favourite bubble tea brand with him/her. They must be the first, the second and the third one (Recalling that students favouring the same brand always stand together). Hence, 4 students having the same favourite brand with the fifth one must be the ones with numbers from 4^{th} to 7^{th}. Therefore, the only possible sequence of answers (If all students are asked) is $(3, 3, 3, 4, 4, 4, 4)$.

In the second sample, the second student must not have the same favourite brand with the first one, and the third student must not have the same favourite brand with the last one. However, Hanh does not know whether the second and the third student share the same brand or not. As a result, there are two possible sequences of answers: $(1, 1, 1, 1)$ and $(1, 2, 2, 1)$.

L. SCTAB

You are given table A, which consists of m rows and n columns. Each cell of the table contains an integer whose absolute value does not exceed 100. You can apply the operation sort(k) which arranges the rows of the table in ascending order according to the value of the cell in the k-th column. If there are several rows that have the same value in the k-th column, then the relative order of the rows does not change.

You are also given table B, which is formed by applying a sequence of operations sort(k_1), sort(k_2), ..., sort(k_s) on table A. Find the shortest sequence of operations that when applied on table A also results in table B.

INPUT

The first line contains 3 integers m, n, and s. ($m \le 20$, $n \le 9$, $s \le 100$).

The next m lines contain n integers, denoting the cells in table A.

The next line contains s integers k_1, k_2, k_3, ..., k_s ($1 \le k_i \le n$), denoting a sequence of operations that can be applied on table A to form table B.

OUTPUT

In the first line, print the length of the shortest possible sequence of operations that results in table B. In the next line, print the sequence of operations. If there are multiple sequences that have the same length, print the lexicographically smallest sequence.

Sample Input	Sample Output
2 2 4 1 2 2 1 1 2 1 2	1 2