The 2022 ICPC Vietnam

Southern Provincial Programming Contest

University of Science, VNU-HCM

October 30th, 2022

Contest length: 5 hours

The problem set consists of 13 problems in 17 pages (*excluding the cover page*):

- Problem A: **Cypriot Game**
- Problem B: **Transportation System Renovation**
- Problem C: **Migration**
- Problem D: **Open Sesame!**
- Problem E: **Sum of Squares**
- Problem F: **Expected Tokens**
- Problem G: **Longest Subsequence**
- Problem H: **Beauty of a Number**
- Problem I: **Metro and Buses**
- Problem J: **Not a Classic String Problem**
- Problem K: **HTML**
- Problem L: **Addition**
- Problem M: **Two Specials**

Problem A
Cypriot Game

Time Limit: 1 second
Memory Limit: 512 megabytes

Cuong and Mai are playing a game with numbers, originated in Cyprus - a small country in Europe. The game starts with a pair of positive integers (x, y). They will take an alternative turn transforming this pair, Cuong will take the first turn. In each turn, a player may perform one of the following moves:

- If x is even, $(x, y) \rightarrow \left(\frac{x}{2}, y + \frac{x}{2}\right)$;
- If y is even, $(x, y) \rightarrow \left(x + \frac{y}{2}, \frac{y}{2}\right)$;

There might be 3 outcomes of the game:

- If the game reaches a state where both x and y are odd, a player will have no valid move and he/she will lose the game.
- If the game reaches a state where it has reached before, the game will end with a draw.

Assume that both Cuong and Mai playing optimally, you are given n, your task is to count the number of starting states (x, y) such that $1 \leq x, y \leq n$ and the game will end with each of the 3 outcomes.

Input

The input contains only one integer n ($1 \leq n \leq 10^9$).

Output

You should print 3 integers in 3 lines:

- The number of starting states that Cuong will win;
- The number of starting states that will lead to a draw;
- The number of starting states that Mai will win.

Sample Input	Sample Output
2	1 2 1

Explanation

There are 4 starting states:

- $(1, 1)$ Cuong has no move, thus Mai wins this game;
- $(2, 1)$ Cuong has only one move to make it $(1, 2)$, then Mai has only one move to make it $(2, 1)$, thus the game ends with a draw;
- $(1, 2)$ - this game will be similar to the previous one;
- $(2, 2)$ Cuong has two moves to make the pair $(1, 3)$ or $(3, 1)$. In either case, Mai has no more valid move. Thus Cuong wins this game.

Problem B
Transportation System Renovation

Time Limit: 1 second
Memory Limit: 512 megabytes

Ho Chi Minh city is the largest and most populated city in Vietnam. The city has n junctions (numbered from 1 to n), connected by m bidirectional roads (numbered from 1 to m). Each road connects two junctions in the city, and each road has a certain carrying capacity. A truck with weight w can travel on a road with a capacity c if $w \le c$.

The city is preparing an renovation plan on the transportation system. During the renovation process, there might be an increase or decrease in the capacity of some roads.

You are given the current transportation system and q queries, each query is either:

- $C\ i\ x$: Change the capacity of the road i to x.
- $S\ a\ b\ w$: Check if a truck with weight w can travel from a to b where a and b are junctions.

Input

Input starts with 3 integers n, m and q $(1 \le n \le 1\,000, 1 \le m \le 100\,000)$.

The i^{th} line of the next m lines contains 3 integers $u\ v\ c_i$ indicating there is a road with capacity c_i from u to v.

The next q lines contain q queries in the above format. It is guaranteed that the capacity of the roads never exceeds 10^9 and there should be no more than 2000 queries of type C.

Output

For each query of type S, you should print the answer YES or NO.

Sample Input	Sample Output
3 4 6 1 2 3 2 3 3 2 1 1 1 2 1 S 1 2 4 S 2 3 2 C 1 4 S 1 2 4 C 2 1 S 2 3 2	NO YES YES NO

Problem C
Migration

Time Limit: 1 second
Memory Limit: 512 megabytes

City X is about to face a natural disaster, so it is necessary to migrate out of this city. Currently, residents are staying at N starships with unlimited space in the city. However, the government realized that there was only enough fuel for K starships to fly out of the city.

Therefore, the local government immediately planned to combine some residents from N starships to K starships to fly out of the city. The fuel cost for emergency relocation w_i people from the i^{th} starship at coordinate x_i to the j^{th} starship at coordinate x_j is equal to $|x_i - x_j| \times w_i$. Find the solution that needs the lowest fuel cost for relocation.

Input

The first line contains two space-separated integers describing the respective values of N and K $(N, K \leq 5000)$.

Each line of the subsequent lines contains two space-separated integers describing the respective values of x_i and w_i $(1 \leq w_i, x_i \leq 10^6)$.

Output

Minimum cost for relocation people.

Sample Input	Sample Output
3 1 20 1 30 1 40 1	20
3 1 11 3 12 2 13 1	4
6 2 10 15 12 17 16 18 18 13 30 10 32 1	182

Problem D
Open Sesame!

Time Limit: 1 second
Memory Limit: 512 megabytes

Archaeologists find treasure buried deep in the Atlantic world. However, this treasure is protected by cryptographic layers and each layer corresponds to a number. After years of research, archaeologists have found a string S as well as hints to find the password in a mysterious book.

In particular, the i^{th} cryptographic stage will have a magic number k, the cipher being the result of the function $f(w_k)$ where w_k is the k^{th} alphabetically smallest palindromic substring of S. The function f is $f(p) = \sum_{i=1}^{|p|} (p_i \times a^{l-i}) \ mod \ m$ where p_i is the ASCII value of the i^{th} character in the string p, $a = 100001$, and $m = 10^9 + 7$.

However, over the years, there will be some blurring places leading to w_k not existing, then the answer is -1.

Input

The first line contains 2 space-separated integers describing the respective values of N (the length of the string S) and Q (the number of cryptographic layers) ($N, Q \leq 10^5$).

The second line contains a single string denoting S. It is guaranteed that the string S consists of only lowercase English alphabetic letters (*i.e.* 'a' to 'z').

Each of the Q subsequent lines contains a single integer denoting the value of K for a query ($K \leq N \times (N + 1)/2$).

Output

For each layer, print the password if the magic number K in the book is correct. Otherwise, print -1.

Sample Input	Sample Output
5 7 abcba 1 2 3 4 6 7 8	97 97 696207567 98 29493435 99 -1

Problem E
Sum of Squares

Time Limit: 1 second
Memory Limit: 512 megabytes

Mr. Nhat has a problem consisting of N steps. At the i^{th} step, Nhat is given 2 numbers A_i and B_i, then Nhat has to choose a real number R_i such that R_i is not larger than the numbers chosen in the previous steps. The score of the problem is defined as $\sum_i (A_i - R_i * B_i)^2$ of all N steps.

What is the smallest score that Mr. Nhat can get?

Input

The first line contains an integer N. $(2 \le N \le 5 \times 10^5)$

The second line contains N space-separated integers $A_1, A_2, \dots, A_N$

The third line contains N space-separated integers $B_1, B_2, \dots, B_N$.

$(1 \le A_i, B_i \le 1000)$

Output

The smallest score Mr. Nhat can get. Your answer is considered correct if its absolute or relative error does not exceed 10^{-6}.

Sample Input	Sample Output
2 2 5 1 8	0.000000000000000
5 7 9 1 4 3 9 8 6 13 1	12.247238031469687
10 66 23 51 81 60 7 26 127 66 8 9 88 77 12 2 38 7 63 90 111	17698.696831405897683

Problem F
Expected Tokens

Time Limit: 1 second
Memory Limit: 512 megabytes

You are given a set of n coins, and the probability of turning head for the i^{th} coin is p_i. The game is as follows:

- You are going to play k games. At the beginning, you have x tokens.
- In each game, you choose one coin in the coin set and always pick head. If the coin you choose in that game turns head, you win the game, otherwise, you lose.
- In case you win a game, you will receive a_i tokens ($a_i \geq 0$).
- In case you lose a game, you will lose l_i percent of your current amount of tokens.

Your task is to choose k coins in the set of n coins and the order of the coins you use in the k games to maximize the expected amount of tokens. Note that each coin can be used only once.

Input

There are multiple test cases in the input.

The first line of each test case contains three space-separated integers n, k, and x .

$(1 \leq k \leq n \leqslant 100, 0 \leq x \leq 10^6)$.

Each of the next n lines specifies the properties of the i^{th} game with three space-separated integers a_i, l_i, and p_i $(0 \leq a_i, l_i, p_i \leq 100)$. The input terminates with a line containing 0 0 0 which should not be processed.

Output

For each test case, output a single line containing the maximum expected amount of your final tokens rounded to exactly two digits after the decimal point.

Sample Input	Sample Output
2 2 100 10 0 50 100 10 20 2 1 100 10 0 50 100 10 20 0 0 0	117.00 112.00

Problem G
Longest Subsequence

Time Limit: 1 second
Memory Limit: 512 megabytes

You are given a permutation of n numbers from 1 to n, which are $p_1, p_2, \ldots, p_n$, and m pairs of indices (u, v) where $1 \le u, v \le n$. You can perform a series of moves based on these pairs. In each move, you can choose one pair and swap two numbers at the positions u and v in the permutation p. One pair can be used as many times as you want.

An increasing subsequence of length k in p is

$$p_{j_1} < p_{j_2} < \cdots < p_{j_k} \text{ and } j_1 < j_2 < \cdots < j_k$$

Let max_k be the length of the longest increasing subsequence of p.

Your task is to perform multiple moves to get a permutation p that has the maximum value of max_k.

Input

The first line contains two integers n and m.

The second line contains the permutation $p_1, p_2, \ldots, p_n$.

In the next m lines, each line contains two integers u, v.

Constraints

$$1 \le n \le 10^4; 0 \le m \le \left(10^5, \frac{n(n-1)}{2}\right)$$

Output

The output contains only one integer, which is the maximum value of max_k.

Sample Input	Sample Output
6 2 5 2 4 6 3 1 5 6 1 5	4

Explanation

We should make two moves $(5, 6)$ and $(1, 5)$ to get the permutation $(1, 2, 4, 6, 5, 3)$.

Problem H
Beauty of a Number

Time Limit: 1 second
Memory Limit: 512 megabytes

Dang loves numbers, but each number has its own beauty in Dang's eyes. The beauty of a number is defined as the multiplication of all its digits. For example, the number 123 has the beauty of $1 \times 2 \times 3 = 6$.

Dang wants to know what the most beautiful number in the range of $[l, r]$.

Your task is to help Dang find the beauty of the most beautiful number in the range of $[l, r]$.

Input

The first line contains l and the second line contains r. The numbers of the digits of l and r are not greater than 10^5.

Output

The output contains only the beauty of the most beautiful number in the range of $[l, r]$. The output should be modulo by $10^9 + 7$.

Sample Input	Sample Output
1 30	18

Explanation

The most beautiful number in the range $[1, 30]$ is 29 and its beauty is 18.

Problem I
Metro and Buses

Time Limit: 3 Seconds
Memory Limit: 512 megabytes

Ho Chi Minh city has built a metro system. The metro system consists of n stations and m bi-directional lines. The i^{th} line connects the stations a_i and b_i, and has a cost c_i.

It is known that we can get from any station to any other (possibly with transfers), and the cost of any route that consists of several consequent lines is equal to the cost of the most expensive in them. More formally, the cost of the route from the station s_1 to station s_k with $(k-2)$ transfers using stations $s_2, s_3, s_4, \ldots, s_{k-1}$ is equal to the **maximum cost of the lines** from s_1 to s_2, from s_2 to s_3, from s_3 to s_4, and so on until the line s_{k-1} to s_k.

Today, the government decides to build some bus lines (also bi-directional) for all pairs of metro stations that are not directly connected by metro lines. Therefore, between any pair of stations, there is one direct line of either metro or bus.

The cost of the bus is calculated as follows: for each pair of stations a and b that is connected by a bus line, the cost of this line is equal to the **minimum cost of the metro route** from a to b according to the pricing strategy described above.

It is known that with the help of bus lines, you can get from any station to any other with possible transfers, and, similarly to the metro, the cost of a route between any two stations that consists of several bus lines is equal to the cost of the most expensive line.

Due to the increasing competition with the bus system, the metro's manager introduces a new reform and changes the costs of its lines. Particularly, for the i^{th} metro line that connects the stations a_i and b_i, the cost of this line should be equal to the minimum cost of the (bus) route between the stations a_i and b_i by bus. Please help the metro's manager calculate the new costs of metro lines.

Input

The input consists of multiple test cases. The first line contains one integer t $(1 \le t \le 10^4)$ — the number of test cases.

The first line of each test case contains two integers n and m, the number of stations in Ho Chi Minh Station and the number of metro lines $\left(4 \le n \le 2 \times 10^5,\ n-1 \le m \le 2 \times 10^5,\ m \le \frac{(n-1)(n-2)}{2}\right)$.

The next m lines contain the description of the metro lines. The i^{th} line contains three integers a_i, b_i and c_i $(1 \le a_i, b_i \le n, 1 \le c_i \le 10^9)$ — the two stations that are connected with the i^{th} metro line and the price of the i^{th} metro line.

It is guaranteed that no line connects a station with itself, no two lines connect the same pair of stations. It is guaranteed that by using metro lines it is possible to get from any station to any other and by using bus lines it is possible to get from any station to any other.

Let N be the sum of n over all test cases and M be the sum of m over all test cases. It is guaranteed that $N, M \le 2 \times 10^5$.

Output

For each test case you should print m integers in a single line where the i^{th} number is the price of the i^{th} metro line after the reform.

Sample Input	Sample Output
3 4 3 1 2 1 2 3 2 4 3 3 5 5 1 2 1 1 3 1 2 4 1 4 5 2 5 1 3 6 6 1 2 3 2 3 1 3 6 5 3 4 2 4 5 4 2 4 2	3 3 3 1 1 1 2 2 4 4 5 3 4 4

Explanation

In the first test case, bus system will provide lines between these pairs of stations: (1,3), (1,4) and (2,4).

The cost of a route between stations 1 and 3 will be equal to 2, since the minimum cost of the metro route is 2 — the route consists of a line between stations 1 and 2 costing 1 and a line between stations 2 and 3 costing 2, the maximum cost is 2.

The cost of a route between stations 1 and 4 will be 3, since the minimum cost of the Metro route is 3 — the route consists of a line between stations 1 and 2 costing 1, a line between stations 2 and 3 costing 2, and a line between stations 3 and 4 costing 3, the maximum cost is 3.

The cost of a route between stations 2 and 4 will be 3, since the minimum cost of the Metro route is 3 — the route consists of a line between stations 2 and 3 costing 2 and a line between stations 3 and 4 costing 3, the maximum cost is 3.

After the reform, the cost of the Metro line between stations 1 and 2 will be 3, since the minimum cost of the Bus route between these stations is 3 — the route consists of a line between stations 1 and 4 costing 3 and a line between stations 2 and 4 costing 3, the maximum cost is 3.

The cost of the Metro line between stations 2 and 3 will be 3, since the minimum cost of the Bus route between these stations is 3 — the route consists of a line between stations 2 and 4 costing 3, a line between stations 1 and 4 costing 3 and a line between 1 and 3 costing 2, the maximum cost is 3.

The cost of the Metro line between stations 3 and 4 will be 3, since the minimum cost of the Bus route between these stations is 3 — the route consists of a line between stations 1 and 3 costing 2 and a line between stations 1 and 4 costing 3, the maximum cost is 3.

In the second test case, the Bus system will have the following lines: between stations 1 and 4 costing 1, between stations 2 and 3 costing 1, between stations 2 and 5 costing 2, between stations 3 and 4 costing 1, and between stations 3 and 5 costing 2.

Problem J
Not a Classic String Problem

Time Limit: 1 Second
Memory Limit: 512 megabytes

There is a classic string problem: You are given two strings A and B, and your task is to find the number of appearances of string A in string B.

Dr. Hung does not want the problem to be that easy, so he gives you two strings S, T, and q queries. Each query has two pairs of indices (i, j) and (u, v). For each query, Dr. Hung asks you to find the number of appearances of string $X = S_i S_{i+1} \dots S_j$ in string $Y = T_u T_{u+1} \dots T_v$.

Input

The first line contains string S $(1 \le |S| \le 2 \times 10^5)$.

The second line contains string T $(1 \le |T| \le 2 \times 10^5)$.

The third line contains q $(1 \le q \le 5 \times 10^5)$.

In the next q lines, each line contains four integers i, j and u, v.

Output

The output has q lines, each line contains a single integer indicating the number of appearances.

Sample Input	Sample Output
abb ababababb 5 1 2 1 7 2 3 2 9 3 3 4 7 1 2 2 4 1 1 1 9	3 1 2 1 4

Problem K
HTML

Time Limit: 1 Second
Memory Limit: 512 megabytes

Web colors are colors used in displaying web pages. Each color may be specified as an RGB triple, or a common English name used for that color. Colors are specified according to the intensity of their red, green, and blue components, each represented by eight bits. Thus, there are 24 bits used to specify a web color, and a total of 16,777,216 colors can be imagined as web colors. But the HTML 4 specification defines only 16 named colors as shown in the table.

#	Name	Red	Green	Blue
1	White	255	255	255
2	Silver	192	192	192
3	Gray	128	128	128
4	Black	0	0	0
5	Red	255	0	0
6	Maroon	128	0	0
7	Yellow	255	255	0
8	Olive	128	128	0
9	Lime	0	255	0
10	Green	0	128	0
11	Aqua	0	255	255
12	Teal	0	128	128
13	Blue	0	0	255
14	Navy	0	0	128
15	Fuchsia	255	0	255
16	Purple	128	0	128

It is often useful to map one given color to one of the HTML-named colors. The goal of this problem is to perform just such a mapping in the RGB color space. The input to the program consists of a collection of RGB color values to be mapped to the closest HTML-named color.

For a given color, the "closest" color in the HTML color names is a color with the smallest Euclidean distance from the given color. That is, if RGB is the color to be mapped, and $\{R_1G_1B_1, \ldots, R_{16}G_{16}B_{16}\}$ is the set of the HTML colors, the closest color is the one which minimizes the distance expression

$$d = \sqrt{(R_i - r)^2 + (G_i - g)^2 + (B_i - b)^2}$$

where i is an integer from 1 to 16.

Input

There are multiple test cases in the input. Each test case consists of a line containing three integers $0 \le r, g, b \le 255$ which are the Red, Green, and Blue intensities of the color, respectively. The input terminates with $-1\ -1\ -1$ which should not be processed.

Output

For each test case, output a line containing the name of the closest HTML color to the given color. If there is more than one closest color, print the one with a smaller associated number in the above table.

Sample Input	Sample Output

`120 120 10`
`111 112 113`
`5 135 8`
`-1 -1 -1`
	`Olive`
	`Gray`
	`Green`

Problem L
Addition

Time Limit: 1 second
Memory Limit: 512 megabytes

Long is a 6 year-old boy and he is learning addition in a multi-digit column adding way. He usually performs an addition of two integers like this:

$$\begin{array}{r} 321 \\ +\quad 456 \\ \hline 777 \end{array}$$

Sometimes, Long performs addition incorrectly and now it is the time for you to help him before he submits his homework to his teacher. Since Long has already written the addition to his notebook, you can only erase some columns in his addition to make it correct. For example, in the following addition, you can obtain a correct addition by erasing the first and the fourth columns.

$$\begin{array}{r} 71212 \\ +\quad 12348 \\ \hline 93650 \end{array} \rightarrow \begin{array}{r} 122 \\ +\ 238 \\ \hline 360 \end{array}$$

Your task is to find the minimum number of columns needed to be erased such that the remaining columns form a correct addition.

Input

There are multiple test cases in the input.

Each test case starts with a line containing the single integer n, the number of digit columns in the addition ($1 \leq n \leq 1000$).

Each of the next 3 lines contains a string of n digits. The number on the third line is presenting the sum (*not necessarily correct*) of the numbers in the first and second lines. The input terminates with a line containing "0" which should not be processed.

Output

For each test case, print a single line containing the minimum number of columns needed to be erased.

Sample Input	Sample Output
3 321 456 777 5 71212 12348 93650 2 24 32 32	0 2 2 1

5 12299 12299 25598 0	

Problem M
Two Specials

Time Limit: 8 seconds
Memory Limit: 512 megabytes

In order to make Fabulous City a more attractive place for tourists, the government came up with the following plan: choose two roads of the city and call them SPECIALS. Certainly, these SPECIALS will be proclaimed extremely important special places, which should attract tourists worldwide.

The city can be represented as a graph with the vertices as crossroads and the edges as roads connecting two crossroads. In total, there are n vertices and m edges in the graph, you can move in both directions along any road. You can get from any crossroad to any other by moving only along the roads. Each road connects two different crossroads and no two roads connect the same pair of crossroads.

In order to reduce the traffic along the SPECIALS, it was decided to introduce a toll on each SPECIAL in both directions. Now you need to pay 1 coin for one pass along the SPECIAL. The other roads cost you 0 coin.

Analysts have collected a sample of k citizens, and the i^{th} citizen needs to go to work from the crossroad u_i to the crossroad b_i. After two SPECIALS are chosen, each citizen will go to work along the path with minimal cost.

In order to earn as many coins as possible, it was decided to choose two roads as two SPECIALS, so that the total number of coins paid by these k citizens is maximized. Your task is to help the government: according to the given scheme of the city and a sample of citizens, find out which two roads should be made SPECIALS, and how many coins the citizens will pay according to this choice.

Input

Each test consists of multiple test cases. The first line contains one integer t $(1 \le t \le 10^5)$ — the number of test cases.

The first line of each test case contains two integers n and m $\left(3 \le n \le 5 \times 10^5,\ n-1 \le m \le 5 \times 10^5, m \le \frac{n(n-1)}{2}\right)$ - the number of crossroads and roads, respectively.

The next m lines contain the description of roads, the i^{th} line contains two integers s_i and f_i $(1 \le s_i, f_i \le n, s_i \ne f_i)$ — indexes of crossroads which are connected by the i^{th} road. It is guaranteed that no two roads connect the same pair of crossroads, you can get from any crossroad to any other by moving only along the roads.

The next line contains a single integer k $(1 \le k \le 5 \times 10^5)$ — the number of citizens in the sample.

The next k lines contain the description of citizens, the i^{th} line contains two integers u_i and v_i $(1 \le u_i, v_i \le n, u_i \ne v_i)$ - the i^{th} citizen goes to work from crossroad u_i to crossroad v_i.

Let M be the sum of m over all test cases and K be the sum of k over all test cases. It is guaranteed that $M, K \le 5 \times 10^5$.

Output

For each test case, print the answer to the problem.

In the first line, print the total amount of coins that citizens will pay.

In the second line, print two integers x_1 and y_1 - the numbers of crossroads that will be connected by the first SPECIAL.

In the third line, print two integers x_2 and y_2 - the numbers of crossroads that will be connected by the second SPECIAL.

The numbers of crossroads connected by a SPECIAL can be printed in any order, each of the printed roads should be among m roads of the city, chosen roads should be different.

Sample Input	Sample Output
3 5 5 1 2 2 3 3 4 4 5 5 1 6 1 5 1 3 1 3 2 4 2 5 5 3 6 5 1 2 2 3 2 4 4 5 4 6 3 1 6 5 3 2 5	5 1 5 3 2 5 4 2 5 4