少年圖靈計畫
Young Turing Program

0_Hello World

(30 points)

Introduction

YTP Contest has started!

Let’s verify everything first.

Is the internet setting correct?

Is the source code submission working well?

Do you use STDOUT output for program solutions?

Everything is ready! Go get 30 points now!! Go! Go! Go!

Statement

Please write a program to output Hello World!

Input Format

This problem requires no input.

Output Format

[A~Z][a~z], space, and common English punctuation.

Constraints

[A~Z][a~z], space, and exclamation mark "!".

Test Cases

Input 1

(no input)

Output 1

```
Hello World!
```

Illustrations

Input 1 has no input, simply output Hello World!

1_Streetlights

(2 points /8 points)
Time Limit: 1 second
Memory Limit: 256MB

Description

Streetlighting systems are designed to ensure sufficient lighting during dark hours, especially to prevent accidents while driving or other similar situations.

Due to a high number of recent car accidents in a small village, it was discovered through investigations that insufficient streetlight installations were the cause. In response, the village chief decided to address this issue by adding more streetlights to enhance safety. However, due to budget limitations, the proper placement of streetlights became a concern for the village chief.

Based on some optical principles and other physics knowledge, as well as some investigations, the village chief has discovered that there must be at least one street light within every 15 meters for safety. This means that for a street light placed at coordinate x, the illuminated range is $(x-15, x+15)$, and it is necessary to place street lights at $x-15$ and $x+15$ to ensure safety.

In other words, if a road has a length of 100 units (with the starting point at 1 and the endpoint at 100, denoted as the closed interval $[1, 100]$), without any street lights, the minimum requirement to achieve safety is to place street lights at $15, 30, 45, 60, 75, 90$.

Now, given the length of a road denoted by L and the number of streetlights already installed, denoted by N, the task is to determine the minimum number of additional streetlights needed to meet the safety threshold within the interval $[1, L]$.

Input Format

The first line of input contains two positive integers, L and N, representing the length of the road and the number of streetlights, respectively.

The second line contains N positive integers a_i, indicating the positions of the existing streetlights. It is guaranteed that $1 \leq a_i \leq L$, and there are no more than one streetlight at the same position.

Output Format

Output a single integer, denoting the minimum number of additional streetlights required.

Constraints

- $1 \leq L \leq 10^{18}$.
- $1 \leq N \leq 2 \times 10^5$.
- $1 \leq a_i \leq L$ $(1 \leq i \leq N)$.
- There are no more than one streetlight at the same position, that is, $a_i \neq a_j$ for $1 \leq i < j \leq N$.

Subtasks

- Subtask 1 (2 points) $L \leq 10^5$, $N \leq 1000$.
- Subtask 2 (8 points) No additional constraints.

Test Cases

Input 1

```
100 4
60 15 90 30
```

Output 1

```
2
```

Input 2

```
50 1
16
```

Output 2

```
3
```

Input 3

```
10 10
1 2 3 4 5 6 7 8 9 10
```

Output 3

```
0
```

Illustrations

In Example 1, the additional streetlight positions needed to meet the safety threshold are $45, 75$.

In Example 2, the additional streetlight positions needed to meet the safety threshold can be $1, 31, 46$.

In Example 3, the safety threshold is already met, so no additional streetlights are needed.

2_Professor_of_Professors

(10 points)
Time Limit: 1 second
Memory Limit: 256MB

Statement

National Giraffe University (NGU) has N professors. Each professor has at most one advising professor. Professor i claims to be the advising professor of a_i professors. Note that a professor can never be his/her own advising professor.

Professor Cheng from National Okapi University (NOU) is doing research on the professors of NGU recently. He defines Professor x to be a boss of Professor y if and only if there exists some professors $x = v_1, v_2, \ldots, v_k = y$ such that Professor v_i is the advising professor of Professor v_{i+1}, for all $1 \leq i < k$. Note that a professor can never be his/her own boss.

During his research, Professor Cheng discovered that there is exactly one professor that doesn't have an advising professor. Moreover, this professor is also a boss of every other professor. He calls this special professor "Professor Ji".

Let c_i be the number of bosses of Professor i. Professor Cheng wants to know the minimum value of $\max\limits_{1 \leq i \leq N} \{c_i\}$. He also wants to know any possible setting that achieves this minimum.

Input Format

The first line contains a positive integer T representing the number of testcases.

The first line of each testcase contains one positive integer N.

The second line of each testcase contains N integers $a_1, a_2, \ldots, a_N$.

Output Format

For each testcase, output two lines.

The first line consists of one integer, representing the minimum value of $\max\limits_{1 \leq i \leq N} \{c_i\}$.

The second line consists of N integers $p_1, p_2, \ldots p_N$, representing one possible setting that achieves the minimum value of $\max\limits_{1 \leq i \leq N} \{c_i\}$. If Professor i is Professor Ji, then $p_i = 0$. Otherwise, Professor p_i is the advising professor of Professor i. If there are multiple ways to achieve the minimum value, output any one of them.

Constraints

- $1 \leq T \leq 1000$
- $2 \leq N \leq 3 \cdot 10^5$
- $0 \leq a_i \leq N - 1$

- $\sum_{i=1}^{N} a_i = N - 1$
- It is guaranteed that the sum of N over all testcases does not exceed $3 \cdot 10^5$

Test Cases

Input 1

```
3
5
2 1 0 1 0
10
3 0 2 0 1 1 0 0 2 0
9
0 0 0 0 0 0 0 0 8
```

Output 1

```
2
0 1 2 1 4
3
0 5 1 6 9 1 3 3 1 9
1
9 9 9 9 9 9 9 9 0
```

Illustrations

For the first testcase, one solution is to let the Professor 1 be Professor Ji and the advising professor of Professor $2, 4$. Also, let Professor $2, 4$ be the advising professors of Professor $3, 5$ in this order. Under this setting, we have $(c_1, c_2, c_3, c_4, c_5) = (0, 1, 2, 1, 2)$, which gives $\max_{1 \le i \le n} \{c_i\} = 2$.

Note that solutions may not be unique. As an example, for the first testcase,

```
2
0 1 4 1 2
```

is another possible answer.

3_Code_Guardian

(10 points)
Time Limit: 1 second
Memory Limit: 256MB

Statement

With the introduction of ChatGPT by OpenAI, the field of artificial intelligence (AI) has been made rapid progress since this year. Many tasks have begun to leverage AI collaboration for more efficient fulfillment of requirements. However, communication with computers has become a bottleneck in this collaboration. Therefore, using a domain-specific language (DSL) to accelerate prompts has become a crucial aspect. For example, using specialized artistic language on Midjourney to generate images (e.g., "Apply the Impressionist painting technique..."), or utilizing "autotune" in the upcoming Mojo language for automatic optimization based on processor cores.

Inspired by this idea and driven by human laziness as a catalyst for progress, you need to write a program that utilizes AI to assist you in adding function comments and error handling to your code. This will make your future coding task easier by reading and handling errors effectively. To facilitate explanation and processing, let's assume that the code indentation is always two spaces (" "). Empty lines will be represented as "emptyline", and places requiring error handling will be marked as "safe". Your program should perform the following transformations:

- Replace "safe" with "!!!(Analyze(<unparsed code at the same indentation level, stripped of leading and trailing spaces, separated by commas>))". Note: Originally intended as "ErrorCheck," but for easy to represent, "!!!" is used instead.
- Replace "emptyline" with "//(Describe(start line number, end line number))". Note: Originally intented as "FunctionComment," but for convenience and reduced typing, "//" is used instead. To simplify the issue, only emptylines without leading spaces has to be replaced. Also, the emptyline after the function statements should not include in the function.
- If the indentation is incorrect, output "ERR!".
- line starts with 'end' means program end.

Let me help you revise the text to make it more coherent.

Input	Output	A real example for easy to understand

emptyline
ln2
ln3
SPSPln4
SPSPln5
SPSPSPSPln6
SPSPSPSPln7
SPSPSPSPln8
SPSPSPSPsafe
SPSPSPSPln10
SPSPSPSPsafe
SPSPSPSPln12
SPSPln13
SPSPln14
SPSPsafe
ln16
emptyline
emptyline
ln19
SPSPln20
safe
end
	//Describe(2,16)
	ln2
	ln3
	SPSPln4
	SPSPln5
	SPSPSPSPln6
	SPSPSPSPln7
	SPSPSPSPln8
	SPSPSPSP!!!Analyze(ln6,ln7,ln8)
	SPSPSPSPln10
	SPSPSPSP!!!Analyze(ln10)
	SPSPSPSPln12
	SPSPln13
	SPSPln14
	SPSP!!!Analyze(ln4,ln5,ln13,ln14)
	ln16
	emptyline
	//Describe(19,21)
	ln19
	SPSPln20
	!!!Analyze(ln19)
		# main - create markdown output
		@timer
		def main():
		  result = []
		  for line in sys.stdin.readlines():
		    line = line.rstrip()
		    line = line.replce(' ','ⓔ')
		    line = line+'
		'
		    raise if len(line) >= MAX_LINE_SIZE
		    result.append(line)
		    raise if len(result) >= MAX_ARRAY_SIZE
		    print('.',end='')
		  fh = open('missclose.txt','w+')
		  fh.write(''.join(line))
		  raise if filesize('missclose.txt') == 0
		# end of main
		
		# Entry Point
		if **name** == '**main**':
		  main()
		raise if diskfull()

- This first "emptyline" has converted to ask AI to describe code from line 2 to line 16. The second has no code to convert and keep as "emptyline". The third one requires converting line 19 to line 21.
- The first "safe" hs converted to ask AI analyze the code content ("ln6,ln7,ln8"). The second only ask "ln10" as previous was "parsed" by previous "safe". The third one requires analyzing the same nested level code content (ln4,ln5,ln13,ln14). The last one only analyze "ln19" in the second function block.

PS: 'SP' is ' ' (ASCII: 0x20) character

Input format

Multiline of code, splited by "\n" and end with an "end" string.

Output format

Output "ERR!" if the content has indentation error. Otherwise, output the replaced content.

Constraints

Only alpha-numberic characters, spaces (0x20),underline, parenthsis, and carriage returns

Test Cases

Input 1

```
emptyline
ln2
ln3
  ln4
  ln5
```

```
    ln6
    ln7
    ln8
    safe
    ln10
    safe
    ln12
  ln13
  ln14
  safe
ln16
emptyline
emptyline
ln19
  ln20
safe
end
```

Output 1

```
//Describe(2,16)
ln2
ln3
  ln4
  ln5
    ln6
    ln7
    ln8
    !!!Analyze(ln6,ln7,ln8)
    ln10
    !!!Analyze(ln10)
    ln12
  ln13
  ln14
  !!!Analyze(ln4,ln5,ln13,ln14)
ln16
emptyline
//Describe(19,21)
ln19
  ln20
!!!Analyze(ln19)
```

Input 2

```
emptyline
emptyline
ln3
emtpyline_safe_emptyline
```

```
  ln5
  safe
    ln7
    safe
      ln9
        ln10
          ln11
      safe
emptyline
ln14
end
```

Output 2

```
emptyline
//Describe(3,12)
ln3
emtpyline_safe_emptyline
  ln5
  !!!Analyze(ln5)
    ln7
    !!!Analyze(ln7)
      ln9
        ln10
          ln11
      !!!Analyze(ln9)
//Describe(14,14)
ln14
```

Input 3

```
emptyline
emptyline
emptyline
  safe
      safe
end
```

Output 3

```
ERR!
```

Illustrations

Without line 5th, it can reproduce below as output from line 1-4. But, the line 5th has indentation error and then only output "ERR!" instead.

```
emptyline
emptyline
//Describe(4,4)
  !!!Analyze()
```

4_Yuna_and_Quests

(10 points)

Time Limit: 1 second
Memory Limit: 256 MB

Statement

Yuna is an adventurer whose profession is a bear. She always wears bear-themed costumes and undertakes various quests at the Adventurers' Guild. Although her unique attire often leads to discrimination from people she meets for the first time, that's another story.

Yuna has two types of costumes and two types of summoned beasts as her mounts. The costumes are a white bear costume and a black bear costume, while the summoned beasts are Kumayuru and Kumakyuu. These two costumes and summoned beasts have different abilities that assist her in completing various quests at the guild.

Today, Yuna has accepted n quests. In order to improve the efficiency of performing these quests, she represents each quest with two values, a_i and b_i, which indicate the required costume type and mount type, respectively. Specifically:

- If $a_i = 1$, it means she needs to wear the black bear costume.
- If $a_i = 2$, it means she needs to wear the white bear costume.
- If $a_i = 3$, it means the quest has no requirement for the costume type.
- If $b_i = 1$, it means she needs to ride Kumayuru.
- If $b_i = 2$, it means she needs to ride Kumakyuu.
- If $b_i = 3$, it means the quest has no requirement for the mount.

Initially, Yuna is wearing the black bear costume and riding Kumayuru. She hopes to complete the n quests **in order**. However, changing costumes or mounts is troublesome for her, so she wants to minimize the total number of changes. What is the minimum number of changes she needs to make?

Input Format

The first line of input contains an integer n, which represents the number of quests.

The following n lines, each line contains two positive integers a_i and b_i, representing the values of the i-th quest.

Output Format

Please output a single line containing the minimum number of times Yuna needs to make changes.

Constraints

- $1 \leq n \leq 1000$
- $1 \leq a_i, b_i \leq 3$

Test Cases

Input 1

```
3
1 2
2 3
3 2
```

Output 1

```
2
```

Input 2

```
8
3 3
3 3
3 3
3 3
3 3
3 3
3 3
3 3
```

Output 2

```
0
```

Input 3

```
8
2 2
1 1
2 2
1 1
2 2
1 1
2 2
1 1
```

Output 3

```
16
```

Illustrations

In example testcase 1, one of the optimal solutions for Yuna is to initially switch to the Kumakyuu mount and then change the costume to the White Bear Costume after completing the first task. This way, the total number of changes is 2. It can be proven that there is no smaller solution.# Minimum Distance

5_Minimum_Distance

(3 points /3 points /4 points)
Time Limit: 1 second
Memory Limit: 256 MB

Description

There are N points on a 2D plane.

The distance between any two points (a, b) and (c, d) on this 2D plane is defined as $\max(|a-c|, |b-d|)$. Please find an **non-negative integer coordinate** (x, y) such that its maximum distance to these N points is minimized.

Input Format

The first line of the input will contain a positive integer N, representing the number of points.

In the following N lines, each line will contain two non-negative integers x_i and y_i, representing the coordinates of the i-th point.

Output Format

In the first line, please output a non-negative integer r representing the maximum distance between this coordinate and the N points.

In the second line, please output two non-negative integers x and y representing the coordinate that satisfy the condition. If there are multiple coordinates that meet the condition, please output the one with the smallest lexicographical order. (If there are multiple coordinates that satisfy the conditions, pick the one with the smallest x; if there are still multiple coordinates that satisfy the conditions, pick the one with the smallest y.)

Constraints

- $1 \leq N \leq 200\,000$.
- $0 \leq x_i, y_i \leq 10^9$ ($1 \leq i \leq N$).

Subtasks

- Subtask 1 (3 points) $1 \leq N \leq 5000$, $0 \leq x_i, y_i \leq 1000$ ($1 \leq i \leq N$).
- Subtask 2 (3 points) $0 \leq x_i, y_i \leq 1000$ ($1 \leq i \leq N$).
- Subtask 3 (4 points) No additional constraints.

Test Cases

Input 1

```
2
1 2
2 1
```

Output 1

```
1
1 1
```

Input 2

```
7
801 554
766 608
825 81
850 537
897 460
305 595
666 448
```

Output 2

```
296
601 312
```

Illustrations

In Example 1, the points $(1, 1), (1, 2), (2, 1), (2, 2)$ have a maximum distance of 1 from the given points. Among them, the point with the smallest lexicographical order is $(1, 1)$.

6_Weight_Loss_Plan

(15 points)
Time Limit: 1 second
Memory Limit: 256MB

Description

In the Meow Meow Academy, there was an orange cat who loved competitive programming. His favorite activity was sitting in front of the computer, eating snacks, and solving programming problems. However, as he solved more and more problems, he had consumed too many snacks and became overweight. People in Meow Meow Academy started calling him Pang Pang Cat.

To avoid being made fun of by his name, Pang Pang Cat decided to start exercising. He searched for videos on YouTube and made a list of N exercises. These exercises have $N-1$ relevance, with each relevance indicating a pair of exercises that are relevant. Each relevance is bidirectional. There is no transitivity between relevance. If A and B are relevant, and B and C are relevant, it does not imply A and C are relevant. Since these weight-loss exercises were carefully selected by Pang Pang Cat, for any two exercises, s and t, it is always possible to find a sequence of weight loss exercises, $e_1, e_2, \ldots, e_k$, such that $e_1 = s$, $e_k = t$, and for all $1 \leq i \leq k-1$, e_i and e_{i+1} are relevant.

One fitness combination starts with arbitrary exercise and consists of at least 2 non-repeating exercises, with relevance between any two consecutive exercises. Pang Pang Cat's *happiness* will change after completing any pair of relevant exercises.

Let "the highest happiness in the combination" be denoted as M, "the lowest happiness in the combination" be denoted as m, and "the number of actions in the combination" be denoted as n. We define the *difficulty* of a fitness combination as $\lfloor \frac{M}{2} \rfloor \oplus (m \times 2) \oplus n$, where $\oplus$ represents the bitwise XOR operation.

In order to make the weight loss process less difficult and boring, Pang Pang Cat decided to choose two non-identical fitness combinations to workout. Two fitness combinations are non-identical if and only if there is at least one exercise that is different. Finally, Pang Pang Cat is curious if there exists two fitness combination that meets the above criteria with the same difficulty.

Input Format

The first line of the input consists of a positive integer N, which represents the number of exercises.

The following $N-1$ lines each contain three positive integers, U_i, V_i, W_i, indicating that Pang Pang Cat's happiness will become W_i after completing this pair of relevant exercises.

Output Format

If a solution exists, output four integers a, b, c, d, indicating that the fitness combinations from a to b and from c to d has the same difficulty and non-identical sets of exercises. If (a, b, c, d) is a correct solution, then (b, a, c, d), (a, b, d, c), (b, a, d, c) will also be considered as correct. If no solution exists, output -1.

Constraints

- $3 \leq N \leq 2 \times 10^5$.
- $1 \leq U_i, V_i \leq N$ ($1 \leq i \leq N-1$).
- $1 \leq W_i \leq 10^6$ ($1 \leq i \leq N-1$).
- It is guaranteed that for any two exercises, s and t, it is always possible to find a sequence of weight loss exercises, $e_1, e_2, \ldots, e_k$, such that $e_1 = s$, $e_k = t$, and for all $1 \leq i \leq k-1$, e_i and e_{i+1} are relevant.

Subtasks

- Subtask 1 (100 points) No additional constraints.

Test Cases

Input 1

```
5
1 2 1
1 3 2
3 4 3
3 5 4
```

Output 1

```
1 2 2 3
```

Input 2

```
3
1 2 1
1 3 5
```

Output 2

```
-1
```

Illustrations

In sample 1,

The fitness combination $1 \to 2$ has a maximum happiness of $M = 1$, a minimum happiness of $m = 1$, and a total of $n = 2$ exercises. Therefore, its difficulty is $\lfloor \frac{1}{2} \rfloor \oplus (1 \times 2) \oplus 2 = 0$.

The fitness combination $2 \to 1 \to 3$ has a maximum happiness of $M = 2$, a minimum happiness of $m = 1$, and a total of $n = 3$ exercises. Therefore, its difficulty is $\lfloor \frac{2}{2} \rfloor \oplus (1 \times 2) \oplus 3 = 0$.

Since the two fitness combinations have the same difficulty and non-identical sets of exercises, this is a valid solution.

Note that $(2, 3, 3, 2)$ is not a valid solution because the sets of exercises $\{2, 1, 3\}$ and $\{3, 1, 2\}$ are identical.

In sample 2,

The difficulty of the fitness combination $1 \to 2$ is $\lfloor \frac{1}{2} \rfloor \oplus (1 \times 2) \oplus 2 = 0$ ◦

The difficulty of the fitness combination $1 \to 3$ is $\lfloor \frac{5}{2} \rfloor \oplus (5 \times 2) \oplus 2 = 10$ ◦

The difficulty of the fitness combination $2 \to 3$ is $\lfloor \frac{5}{2} \rfloor \oplus (1 \times 2) \oplus 3 = 3$ ◦

Therefore, you must output -1.

7_TOI_Problem

(15 points)

Time Limit: 1 seconds
Memory Limit: 256 MB

Statement

LDC is preparing an TOI mock contest, and she just came up with a problem which matches the style of TOI. The problem statement is as follows:

There are N pairs of positive integers a_i, b_i ($1 \leq i \leq N$). How many functions f which maps positive integers to positive integers satisfy the following two conditions:

- For all positive integers i which satisfies $1 \leq i \leq N$, $f(a_i) = b_i$.
- For all positive integers x, y, $f(x) \cdot \min(x, f(y)) = x \cdot f(\min(x, y))$ holds

If the answer is infinity, output `-1`, otherwise output the answer modulo 998244353.

LDC is asking you to solve this problem to estimate the difficulty. Please write a program that can solve this problem.

Input Format

The first line of the input contains an integer T, which means there are T test cases.

The format of each test case is as follows:

The first line of the test case contains an integer N.

For all integers i which satisfies $1 \leq i \leq N$, the $i+1$-th line of the test case contains two positive integers a_i, b_i separated by a space.

Output Format

For each test case output one line. If the answer is infinity, output `-1`, otherwise output the answer modulo 998244353.

Constraints

- $1 \leq T \leq 2 \cdot 10^5$
- $1 \leq N \leq 2 \cdot 10^5$
- $1 \leq a_i, b_i \leq 10^{18}$
- $a_i \neq a_j (i \neq j)$
- In a single test file, the sum of N is at most $2 \cdot 10^5$

Test Cases

Input 1

```
2
1
1 1
4
8 21
7777714 49
625 52725
1000000000000000000 864197532
```

Output 1

```
1
0
```

Illustrations

For the first test case, only $f(x) = x$ satisfies all conditions.
Remember to modulo 998244353 to the answer.

8_Protection Fees

(5 points /15 points)
Time Limit: 3 seconds
Memory Limit: 512 MB

Description

As a mature university student, in order to live comfortably on campus, you must pay high protection fees to various factions. The more you pay, the more secure your university life will be.

As a new student at National Qinghan University, your first priority is, of course, to figure out how to pay the protection fee. You are aware that the school is divided into multiple factions, forming a simple undirected graph G. Each node in the graph represents a territory occupied by a single faction, and some territories are connected by edges.

If you accidentally pay protection fees to two factions that there is an edge connecting their territories, a clash will occur, and you will be in big trouble.

Therefore, paying protection fees is also a complicated matter, as a small mistake can quickly turn your university life into a tragedy.

There are a total of K factions in National Qinghan University, numbered from 1 to K. In order to request protection from the t-th faction, you must pay a protection fee of c_t.

Because you believe that the more you pay, the more you get, please find out the maximum amount of protection fee you can pay without triggering any conflicts.

Input Format

The first line contains three non-negative integers N, M, and K, representing the number of nodes, the number of edges, and the number of factions in the undirected graph G, respectively.

The second line contains N integers $a_1, a_2, \ldots, a_N$, representing the factions that occupy the i-th node.

In the next M lines, each line contains two positive integers u_i and v_i, representing an edge in G that connects nodes u_i and v_i.

The last line contains K integers $c_1, c_2, \ldots, c_K$, representing the protection fee charged by the i-th faction.

Output Format

Output a single integer representing the maximum protection fee that you can pay without causing conflict.

Constraints

- $1 \leq N \leq 100\,000$.
- $0 \leq M \leq 100\,000$.
- $1 \leq K \leq 44$.
- $1 \leq a_i \leq K$ $(1 \leq i \leq N)$.

- $1 \le u_i, v_i \le N$ ($1 \le i \le M$).
- $1 \le c_i \le 10^9$ ($1 \le i \le K$).
- The graph is simple and does not contain self-loops or multiple edges.

Subtasks

- Subtask 1 (5 points) $K \le 20$.
- Subtask 2 (15 points) No additional constraints.

Test Cases

Input 1

```
5 4 3
1 2 1 2 3
1 2
2 3
3 4
4 5
1 3 3
```

Output 1

```
4
```

Illustrations

For Example 1, we can choose faction $1, 3$ to maximize the protection fee.

9_Fermat_Last_Theorem

(1 points /2 points /4 points /13 points)
Time Limit: 5 seconds
Memory Limit: 512 MB

Description

Archimedes, an ancient Greek mathematician, learned that Gauss, a modern German mathematician, could quickly calculate the sum of 1 to N and felt dissatisfied.

Therefore, Archimedes upgraded the problem to calculate the sum of $\sum_{i=1}^{N} \lfloor \frac{i}{N} \rfloor$. For this, he needed to prove that for any rational number, there exists an integer smaller than it. With this theorem, he proved the famous Archimedes' principle.

However, soon after, Archimedes found that the answer was always a constant. To increase the difficulty, he changed the numerator and denominator of the fraction, requiring the solution to $\sum_{i=1}^{N} \lfloor \frac{N}{i} \rfloor$.

When he wrote down this expression, Newton passed by and used the value of $\lfloor \frac{N}{i} \rfloor$ and Fubini's theorem to solve this problem. Later, he used this result to prove that $\int_{i=1}^{\infty} \frac{1}{x} dx$ diverges and eventually invented calculus.

Archimedes, who was unwilling to fall behind, increased the difficulty again and changed the problem to give Q queries, each time giving L and R and asking for $\sum_{i=L}^{R} \lfloor \frac{N}{i} \rfloor$. However, the British mathematician Cayley believed that this was still not elegant enough and changed the problem to give N matrices M_i, and each query required calculating the product of $\prod_{i=L}^{R} M_i$.

Galois, who is going to duel tomorrow, heard about this problem, he thought that arithmetic operations in integers were too boring. Therefore, he changed addition and multiplication to operations in $GF(2)$, that is, $a + b$ becomes $(a + b) \bmod 2$, and $a \times b$ becomes $(a \times b) \bmod 2$.

The problem gradually becomes complete: Given N matrices $M_0, M_1, \ldots, M_{N-1}$, there will be Q queries next, and each query gives L and R, requiring calculating the result of $\prod_{i=L}^{R} M_i$, and the operations are performed modulo 2.

While everyone is busy calculating, Fermat, an amateur mathematician, is responsible for writing down all the calculated answers. However, because there are too many queries, he almost used up all the blank space in his notebook. Therefore, he wants everyone to XOR all the answers together before telling him.

However, there is still not enough space left in the notebook, so even though Fermat has found an ingenious proof for Fermat's Last Theorem, he still does not have enough space to write it down.

And more than three hundred years later, the British mathematician Andrew Wiles discovered that Fermat needed more than one hundred pages of blank space, which is the end of the story.

Input Format

The first line of the input contains two positive integers, N and k, representing the number of matrices and the size of each matrix, respectively.

The next N blocks each contains k lines describing a $k \times k$ matrix M_i.

The next line contains a positive integer Q, which represents the number of queries.

The following two lines each contain three positive integers A_1, B_1, x_0 and A_2, B_2, y_0, which represent the formulas used to generate the query intervals.

For $i = 1, 2, \ldots, Q$, the query interval $[L_i, R_i]$ can be generated as follows:

$$x_i = (A_1 \times x_{i-1} + B_1) \bmod (10^9 + 7)$$
$$y_i = (A_2 \times y_{i-1} + B_2) \bmod (10^9 + 7)$$
$$L_i = \min(x_i \bmod N, y_i \bmod N)$$
$$R_i = \max(x_i \bmod N, y_i \bmod N)$$

Output Format

Let $P_i = \prod_{j=L_i}^{R_i} M_j$, where the operation is performed modulo 2.

Please output $\bigoplus_{i=1}^{Q} P_i$, where $\oplus$ is the XOR operation.

Constraints

- $1 \leq N \leq 3 \times 10^5$.
- $1 \leq k \leq 3$.
- $0 \leq$ elements in $M_i \leq 1$ ($0 \leq i \leq N - 1$).
- $1 \leq Q \leq 3 \times 10^7$.
- $1 \leq A_1, B_1, x_0, A_2, B_2, y_0 < 10^9 + 7$.

Subtasks

- Subtask 1 (1 point) $N, Q \leq 3000$.
- Subtask 2 (2 points) $N, Q \leq 3 \times 10^5$.
- Subtask 3 (4 points) $Q \leq 3 \times 10^6$.
- Subtask 4 (13 points) No additional constraints.

Test Cases

Input 1

```
3 3
000
000
000
111
111
111
100
000
000
3
1 1 2
1 1 2
```

Output 1

```
011
111
111
```

Input 2

```
3 2
10
00
01
01
10
00
3
604078599 524693017 85841332
671146413 219874579 658380675
```

Output 2

```
10
00
```

Illustrations

In Input 1,

$$M_1 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, M_2 = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}, M_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}. \tag{1}$$

In Input 2,

$$M_1 = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, M_2 = \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix}, M_3 = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}. \tag{2}$$

Please note that the symbol $\oplus$ represents the XOR operation.

For two $k \times k$ matrices U and V, if $W = U \oplus V$, then for $1 \le i \le k$ and $1 \le j \le k$:

$W[i][j] = (U[i][j] \oplus V[i][j])$.

10_Building_Spanning_Tree

(2 points /2 points /16 points)
Time Limit: 3 seconds
Memory Limit: 512 MB

Description

BST usually stands for Binary Search Tree, but in this problem, it refers to Building Spanning Tree.

Specifically, given an undirected graph G with N vertices and M edges, and a spanning tree T on G, where the vertices are numbered from 1 to N, and for $i = 1 \sim M$, the i-th edge is (u_i, v_i).

Please find a permutation of edges S that satisfies the following conditions:

1. Each edge in G appears exactly once in S.
2. Maintain a graph H with N vertices, initially without any edges between vertices. Then, for each edge (a_j, b_j) in S, if a_j and b_j are not connected in H, add the edge to H; otherwise, skip the edge. The final graph H must be equal to T.
3. Among all the edge sequences satisfying the above conditions, S must have the lexicographically smallest order.

For two edge sequences A and B with length M, A is smaller than B in lexicographic order if and only if there exists a k such that $A_k < B_k$, and for all $t = 1 \sim k-1$, $A_t = B_t$.

For two edges (u_i, v_i) and (u_j, v_j), $(u_i, v_i) < (u_j, v_j)$ if and only if $u_i < u_j$ or $u_i = u_j$ and $v_i < v_j$.

Note that although G is an undirected graph, for each edge (u_i, v_i) given in the problem, the order of its endpoints cannot be changed in the edge sequence. For example, if an edge $(2, 1)$ is given in the problem, it must be treated as $(2, 1)$ and cannot be treated as $(1, 2)$.

Input Format

The first line of the input contains two positive integers N and M, representing the number of vertices and edges in graph G, respectively.

In the following M lines, each line contains two positive integers u_i and v_i, representing an edge that connects vertices u_i and v_i in graph G.

The last line contains $N-1$ integers $k_1, k_2, \ldots, k_{N-1}$, representing that T is composed of the edges (u_{k_j}, v_{k_j}) for all $j = 1 \sim N-1$.

Output Format

Please output the sequence S as required in the problem, with each edge occupying one line.

Constraints

- $2 \leq N \leq 100\,000$.
- $N - 1 \leq M \leq \min\left(200\,000, \frac{N \times (N-1)}{2}\right)$.

- $1 \leq u_i, v_i \leq N$ ($1 \leq i \leq M$).
- $u_i \neq v_i$ ($1 \leq i \leq M$).
- $1 \leq k_j \leq M$ ($1 \leq j \leq N-1$).
- It's guranteed that G is a simple connected graph without multiple edges and self loops, and T is a spanning tree of G.

Subtasks

- Subtask 1 (2 points) $N \leq 500$, $M \leq 500$.
- Subtask 2 (2 points) $M \leq N + 500$.
- Subtask 3 (16 points) No additional constraints.

Test Cases

Input 1

```
5 8
5 2
2 3
3 4
4 1
1 2
3 5
3 1
4 5
1 2 3 4
```

Output 1

```
2 3
3 4
4 1
1 2
3 1
5 2
3 5
4 5
```

Input 2

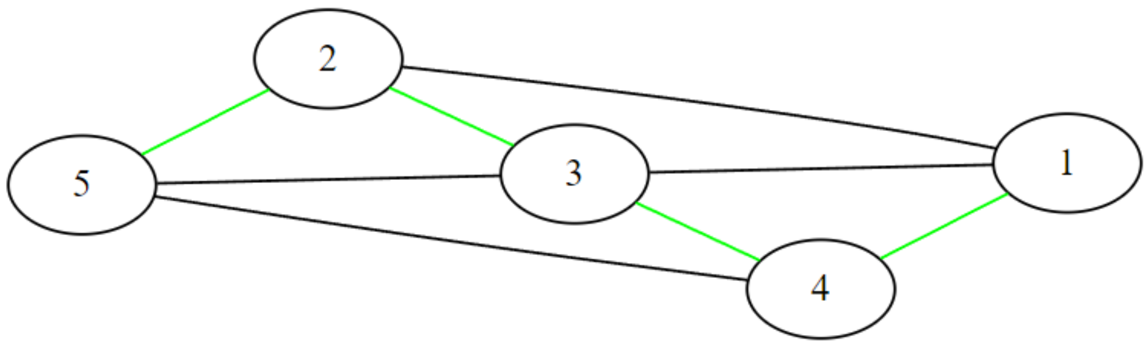
```
6 7
1 2
5 3
4 6
2 4
3 1
6 5
1 6
1 2 3 4 5
```

Output 2

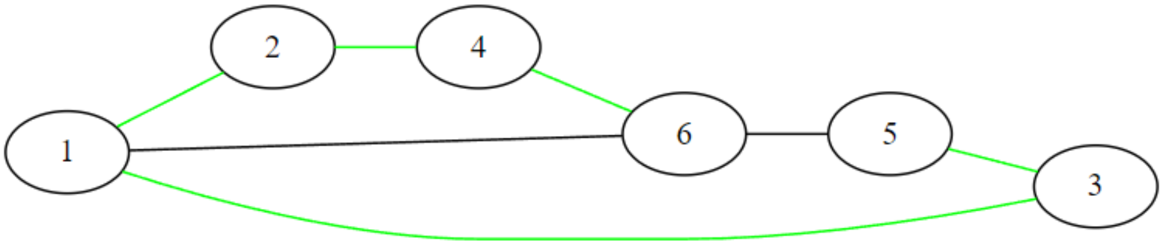
```
1 2
2 4
3 1
4 6
1 6
5 3
6 5
```

Illustrations

The graph in the first test case is shown below, where the green edges represent the edges in the spanning tree T.



The graph in the second test case is shown below, where the green edges represent the edges in the spanning tree T.


1
2
4
6
5
3

11_Oshi no String

(20 points)

Time Limit: 2 second
Memory Limit: 256 MB

Statement

Sarina is a girl who is interested in various things around her. She has an interest in idols she sees on TV and also in the doctor who takes care of her illness every day. Recently, she started exploring different strings and developed an interest in them. She discovered that among many different strings, there are some strings that she particularly likes.

After exploring many strings, Sarina found the following rules for her preferences:

- The longer the string, the more Sarina likes it.
- Among strings of the same length, Sarina prefers the one with a smaller lexicographical order.
- Sarina doesn't like strings that have the same letter appearing more than k times.

Specifically, Sarina will compare the characters in the following order when considering lexicographical order: `abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789_?`.

After understanding her preferences, Sarina wishes to extract a favorite substring from a given circular string. Specifically, her method of extracting the substring is as follows: For a string S of length n, she chooses a pair $[l, r]$ as the starting and ending positions of the substring. If $l \leq r$, she extracts $s_l s_{l+1} \ldots s_r$. Otherwise, she extracts $s_l s_{l+1} \ldots s_n s_1 \ldots s_r$.

As Sarina wants to focus solely on recommending strings, she seeks your help in finding her oshi no string for each string.

Input Format

The first line contains an integer T, which represents the number of test cases.

Each test case consists of one line containing a string S and a positive integer k, as described in the problem.

Output Format

For each test case, output the string that meets the requirements described in the problem.

Constraints

- $1 \leq k \leq |S| \leq 3 \times 10^5$, where $|S|$ represents the length of string S.
- $T, \sum |S| \leq 3 \times 10^5$

Test Cases

Input 1

```
3
cbadcabd 1
aaaedbaaaceea 2
hoshinoaiiii 2
```

Output 1

```
abdc
aacee
hoshinoai
```

Input 2

```
1
woF5ZIpk_?MDOVGq9ES02yKc7iRTP41zBaCXjxuUn3tmh8bWH6sLrYANlQgdJvef 2
```

Output 2

```
aCXjxuUn3tmh8bWH6sLrYANlQgdJvefwoF5ZIpk_?MDOVGq9ES02yKc7iRTP41zB
```

Output 3

```
rasu_mediaMuteki_no_egao
...(skip)...
shiteruA
```

Illustrations

In Example 1, the string cbadcabd can be extracted into the longest subtring of length 4, which includes cbad, badc, dcab, cabd, abdc, acba. Among them, the lexicographically smallest substring is abdc.

In Example 2, any substring can be extracted, and the substrings that meet the requirements are: aCXjxuUn3tmh8bWH6sLrYANlQgdJvefwoF5ZIpk_?MDOVGq9ES02yKc7iRTP41zB.

The complete test data of example 3 can be found in the folder of TestCases by downloading the attachment.

12_Present_for_You

(4 points /16 points)

Time Limit: 3 second
Memory Limit: 512MB

Statement

Vivy has an integer N and a real number $S = \sqrt{1} + \sqrt{2} + \cdots + \sqrt{N}$ on her hand currently.

S can be represented as the sum of integers multiplied by square root of integers. Vivy thinks a representation $(a_1, a_2, \ldots, a_k, b_1, b_2 \ldots, b_k)$ is valid if $S = \sum_{i=1}^{k} a_i \sqrt{b_i}$ and for all $1 \leq i \leq k$, a_i, b_i are integers.

Vivy defines the prettiness of a representation $(a_1, a_2, \ldots, a_k, b_1, b_2 \ldots, b_k)$ as $\sum_{i=1}^{k} a_i^2 b_i$.

Vivy wants to know the value of the maximum prettiness M modulo 21610411 over all possible valid representations. Unfortunately, she is busy stopping ChatGPT from wiping out all of mankind, so she doesn't have time to calculate this value. As a human that is currently being saved Vivy, help her calculate this value as a thank-you gift for her.

Input Format

Input only contains one positive integer N.

Output Format

Output a non-negative integer, representing $M \bmod 21610411$.

Constraints

- $1 \leq N \leq 10^{12}$

Subtasks

- Subtask 1 satisfies that $N \leq 10^6$. (4 points)
- Subtask 2 has no additional constraint. (16 points)

Test Cases

Input 1

```
20
```

Output 1

```
356
```

Illustrations

It can be proven that maximum prettiness is 356 when S is written as

$$
\begin{aligned}
S = & 10\sqrt{1} + 6\sqrt{2} + 3\sqrt{3} + 3\sqrt{5} + \sqrt{6} + \sqrt{7} + \sqrt{10} + \sqrt{11} \\
& + \sqrt{13} + \sqrt{14} + \sqrt{15} + \sqrt{17} + \sqrt{19}
\end{aligned}
$$

13_Kessoku_Band_at_Enoshima

(10 points/15 points)

Time Limit: 10 second
Memory Limit: 512 MB

Statement

The summer vacation has finally arrived! Just the thought of having over a month without having to go to school and socialize makes Bocchi excited. She can't help but look forward to a life where she can stay at home every day and be a guitar hero. However, what excites Bocchi even more is that she has made friends this summer! Having friends means she can finally experience a life of going out and playing with them during the vacation! And so, with these expectations, Bocchi welcomes her first-ever fulfilling summer vacation...

As the time approaches the day before school starts, after a whole summer has passed, none of the friends from the kessoku band have come to invite Bocchi to hang out. This saddens her to the point of starting to regress. When Kita and the group discover Bocchi, she has already devolved into a slug! In order to help Bocchi evolve back into a human, the members of the band decide to go together to Enoshima, to enjoy the last day of summer vacation!

Upon arriving at Enoshima, the band members want to visit various attractions as quickly as possible. However, as Bocchi has devolved into a slug, she is now only capable of moving in the north-south and east-west directions. Therefore, if the band members want to move from point (x_1, y_1) to a destination located at (x_2, y_2), they need to travel a distance of $|x_1 - x_2| + |y_1 - y_2|$ units. Now, there are N attractions on the map, and they hope you can inform them at any time about the distance to the nearest attraction from their current location.

In addition, Kitano's fans on social media occasionally recommend other attractions on Enoshima, giving them the opportunity to reach a certain attraction by moving a shorter distance.

Now, given the locations of N attractions, there are Q events for you to handle. An event may contain $1\ x\ y$, which means the band is currently at (x, y), and they want to know the minimum distance they need to travel to reach an attraction. Alternatively, it may contain $2\ x\ y$, indicating that one of Kitano's fans has informed them about a new attraction, guaranteeing that it is indeed a new one.

Input Format

The first line contains two positive integers N and Q.

Next, there will be N lines, each line containing two integers x_i and y_i, representing the position of the ith attraction.

Following that, there will be Q lines, each representing an event in the format of either " $1\ x_j\ y_j$ " or " $2\ x_j\ y_j$," as mentioned in the problem statement.

Output Format

For each " $1\ x_j\ y_j$" event, output a positive integer representing the minimum distance required to move from (x_j, y_j) to a certain attraction.

Constraints

- $1 \leq N, Q \leq 3 \times 10^5$
- $-5 \times 10^4 \leq x_i, y_i, x_j, y_j \leq 5 \times 10^4$
- It is guaranteed that $N+$ total number of $2\ x_j\ y_j \leq 3 \times 10^5$.

Subtasks

Subtask 1 satisfies that there will only be events of the form $1\ x_j\ y_j$. (10 points)
Subtask 2 has no other restrictions. (10 points)

Test Cases

Input 1

```
5 5
0 4
-2 1
-3 -1
2 1
3 -1
1 -3 3
1 3 2
1 0 0
2 0 1
1 0 0
```

Output 1

```
3
2
3
1
```

Input 2

```
5 7
-3 3
0 0
3 -3
3 3
-3 -3
1 1 2
1 -2 2
1 0 -2
2 0 -3
2 0 3
1 1 2
1 0 -2
```

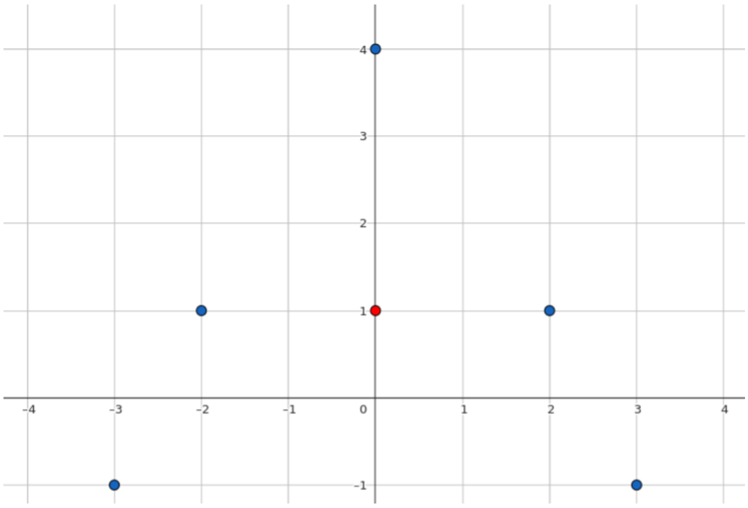
Output 2

```
3
2
2
2
1
```

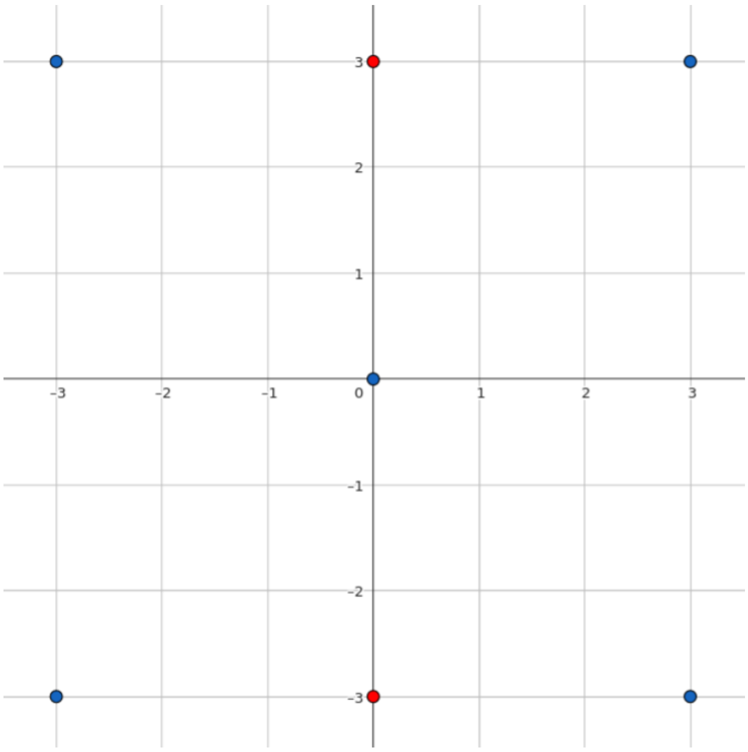
Illustrations

Here are the illustrations of test cases. In the illustration, blue dots represent the initial N attractions, and red dots represent the new attractions provided by Kita's fans.

- Test case 1：



- Test case 2：



14_IQ 999

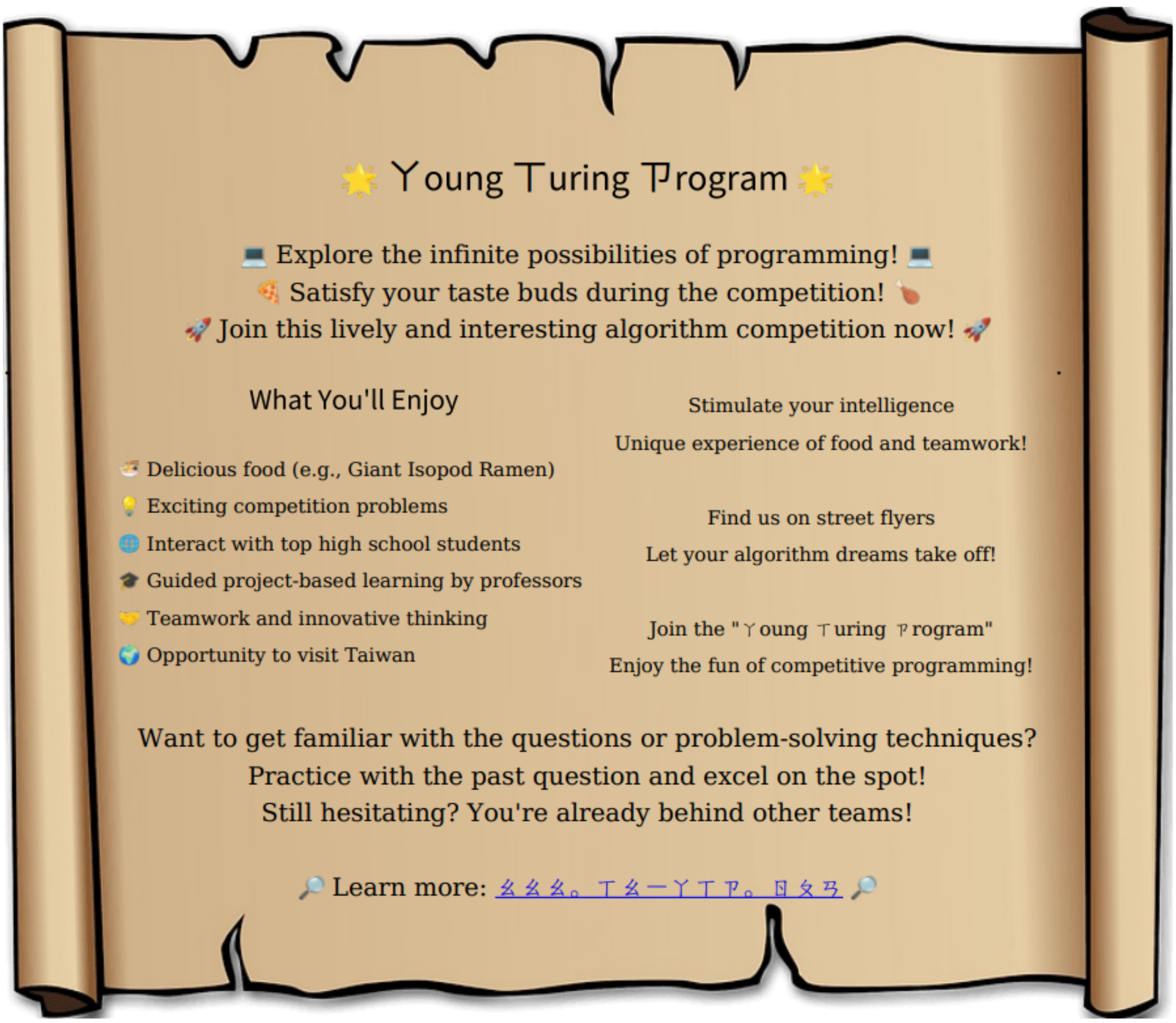
(2 points/ 1 points / 7 points / 4 points / 11 points)

Time Limit: 2 second
Memory Limit: 512 MB

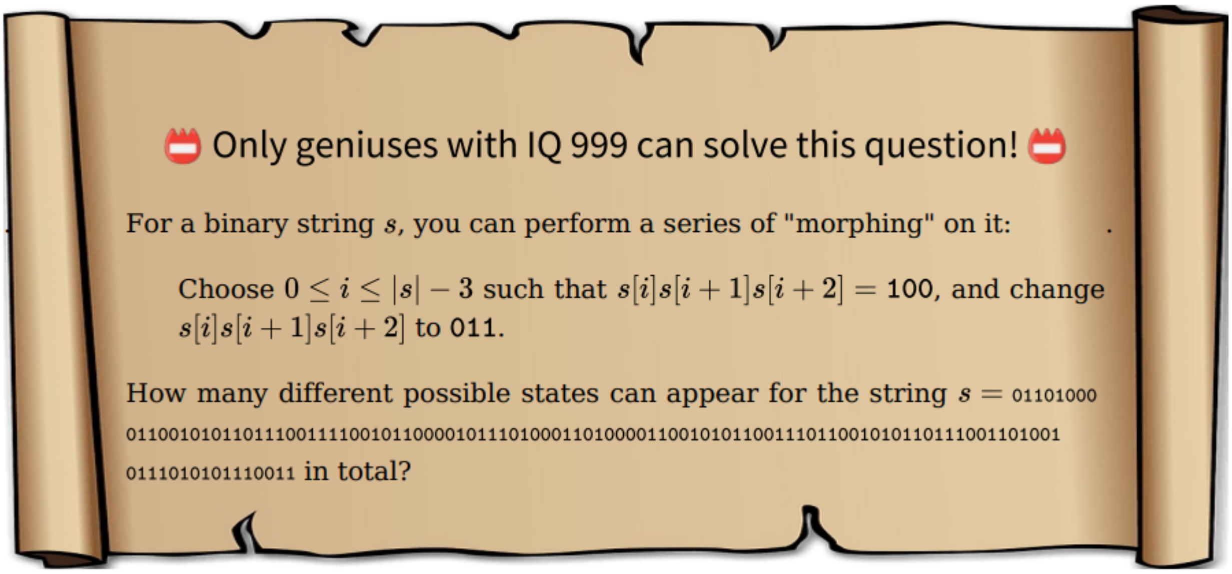
Description

May 13, 2024, Monday: Sunny weather, suitable for boiling water.

Henya's kettle has gone missing, so she went on a long trip to buy a new one. On her way back home, she found a flyer on a utility pole from the "ㄚoung ㄒuring ㄗrogram":



Henya flipped the flyer to the back and indeed found a question:



"It's easy; obviously, it's $172\,800$ dayo!" Henya thought.

Henya decided to test you with this question. She thought the problem shouldn't be that simple, so she made the string length go up to $2\,000\,000$ characters.

"It still seems too easy; let's add some modification operations!" she decided.

So, there will be q queries to modify the character $s[p[i]]$ of the string (change 1 to 0 or 0 to 1). Please output the number of different states the string can have before all the modifications and after each modification, modulo $998\,244\,353$.

Please note that the modifications last forever and are **not** independent of each other.

Input Format

- line 1: s
- line 2: q
- line 3: $p[0] \;\; p[1] \;\; \dots \;\; p[q-1]$
- The 3^{rd} line is empty if $q = 0$.

Output Format

- line 1: The answer before all the modifications.
- line $2 + i \;\; (0 \leq i \leq q-1)$: The answer after $(i+1)^{\text{th}}$ modification.

Constraints

- $1 \leq |s| \leq 2\,000\,000$.
- $s[i] \in \{0, 1\}$ $(0 \leq i \leq |s| - 1)$.
- $0 \leq q \leq 200\,000$.
- $0 \leq p[i] \leq |s| - 1$ $(0 \leq i \leq q - 1)$.

Subtasks

- Subtask 1: $|s| \le 15$, $q \le 100$.
- Subtask 2: $s[0] =$ 1, $s[i] =$ 0 $(1 \le i \le |s| - 1)$, $q = 0$.
- Subtask 3: $|s| \le 3000$, $q \le 3000$.
- Subtask 4: Each modification changes 1 to 0.
- Subtask 5: No additional constraints.

Test Cases

Input 1

```
010100101010001
8
1 11 4 9 3 7 0 11
```

This sample input satisfies the constraints of Subtasks 1, 3, 5.

Output 1

```
12
8
4
2
2
2
2
4
4
```

Input 2

```
1111111
7
6 5 3 1 4 2 0
```

This sample input satisfies the constraints of Subtasks 1, 3, 4, 5.

Output 2

```
1
1
2
3
4
5
4
1
```

Input 3

```
0001111110000101010100011111010
12
26 6 29 11 11 28 11 26 5 13 17 26
```

This sample input satisfies the constraints of Subtasks 3, 5.

Output 3

```
15
30
30
30
24
30
30
24
24
48
72
104
104
```

Input 4

```
011010000110010101101110011110010110000101110100011010000110010101100111011001010110111
0011010010111010101110011
0
```

This sample input satisfies the constraints of Subtasks 3, 5.

Please note that except for the final 0 representing q, all other characters are actually on the same line (you can refer to the `04.in.txt` file in the attachment).

Output 4

```
172800
```

Illustrations

In Example 1, after all the operations are completed, the string will become 100010111110001.
There are four different states that can be reached through a series of morphings:

- 100010111110001
- $100010111110001 \rightarrow \underline{011}010111110001$
- $100010111110001 \rightarrow \underline{011}010111110001 \rightarrow 0110101111\underline{011}01$
- $100010111110001 \rightarrow 1000101111\underline{011}01$

In Example 2, the initial string before any operations is 1111111,
no further morphings can be performed, therefore the answer is 1.
After the 5^{th} operation is completed, the string will become 1010000.
There are five different states that can be reached through a series of morphings:

- 1010000
- $1010000 \rightarrow 10\underline{011}00$
- $1010000 \rightarrow 10\underline{011}00 \rightarrow \underline{011}1100$
- $1010000 \rightarrow 10\underline{011}00 \rightarrow \underline{011}1100 \rightarrow 0111\underline{011}$
- $1010000 \rightarrow 10\underline{011}00 \rightarrow 1001\underline{011}$

Input 4 is the same as the string on the back of the flyer; Henya demonstrated her IQ of 999.

15_Nachoneko_and_the_Math_Homework

(3 points/ 7 points / 13 points / 2 points)

Time Limit: 1.5 second
Memory Limit: 512 MB

Description

Nachoneko mama got tired of drawing and decided to pick up the math homework of the Atlantis Kingdom that her daughter Gura had left nearby. She wanted to challenge herself and solve these problems together with her daughter, despite knowing that neither of them was good at math. Sitting at the desk, she opened the math book and began reading the questions.

However, she quickly got stuck. The problems from the Atlantis Kingdom seemed unfamiliar to her, and she didn't know where to start. That's when she immediately thought of you, a friend who excels in math, and decided to seek your help. She explained her and her daughter's confusion in math and hoped that you could guide them through it.

Let's define a function f that maps an array to an integer as follows:

$$f(B) = \sum_{0 \le i \le j \le |B|-1} \frac{b[i] + b[i+1] + \cdots + b[j]}{\gcd\{b[i], b[i+1], \ldots, b[j]\}} \tag{3}$$

Where $\gcd$ represents the greatest common divisor.

Given an array A of length n, we define $A \backslash a[i]$ as $[a[0], a[1], \ldots, a[i-1], a[i+1], \ldots, a[n-1]]$. For each $i = 0, 1, \ldots, n-1$, please find the value of $f(A \backslash a[i])$. Since the answer could be large, please output the answer modulo $998\,244\,353$.

Input Format

- line 1: n
- line 2: $a[0] \;\; a[1] \;\; \ldots \;\; a[n-1]$

Output Format

- line $1+i$ ($0 \le i \le n-1$): $f(A \backslash a[i]) \bmod 998\,244\,353$

Constraints

- $2 \le n \le 80\,000$.
- $1 \le a[i] \le 998\,244\,352$ ($0 \le i \le n-1$).
- All the inputs are integers.

Subtasks

- Subtask 1: $n \le 300$.
- Subtask 2: $n \le 2000$.

- Subtask 3: $n \leq 16\,000$.
- Subtask 4: No additional constraints.

Test Cases

Input 1

```
4
2 4 6 3
```

This sample input satisfies the constraints of all the subtasks.

Output 1

```
24
21
22
17
```

Input 2

```
10
36 32 27 16 64 30 20 35 42 24
```

This sample input satisfies the constraints of all the subtasks.

Output 2

```
4587
4563
3979
5058
4002
4774
5032
4153
4521
4599
```

Input 3

```
6
249561088 536870912 719323136 998244352 285212672 606076928
```

This sample input satisfies the constraints of all the subtasks.

Output 3

```
6793
5045
5138
6725
8083
7588
```

Illustrations

In the first example:

The first number in the output is the value of $f(A \backslash a[0])$, which is the value of $f([4,6,3])$. The calculation process is as follows:

$$\begin{aligned} f([4,6,3]) &= \frac{4}{\gcd\{4\}} + \frac{4+6}{\gcd\{4,6\}} + \frac{4+6+3}{\gcd\{4,6,3\}} + \frac{6}{\gcd\{6\}} + \frac{6+3}{\gcd\{6,3\}} + \frac{3}{\gcd\{3\}} \\ &= 1+5+13+1+3+1 \\ &= 24 \end{aligned}$$

The third number in the output is the value of $f(A \backslash a[2])$, which is the value of $f([2,4,3])$. The calculation process is as follows:

$$\begin{aligned} f([2,4,3]) &= \frac{2}{\gcd\{2\}} + \frac{2+4}{\gcd\{2,4\}} + \frac{2+4+3}{\gcd\{2,4,3\}} + \frac{4}{\gcd\{4\}} + \frac{4+3}{\gcd\{4,3\}} + \frac{3}{\gcd\{3\}} \\ &= 1+3+9+1+7+1 \\ &= 22 \end{aligned}$$

In the third example, please remember to print the answer modulo $998\,244\,353$.

Afterword

Under your guidance, Nachoneko and Gura solved one math problem after another from the Atlantis Kingdom. They encouraged each other, shared their problem-solving insights, and overcame their difficulties with your help.

In the end, they completed the entire math homework. Nachoneko looked at the completed assignment in her hands with a sense of pride, while Gura felt proud of her progress. They knew that with patience and proper guidance, they could overcome any mathematical challenge.