A. Divide and Multiply by 3

time limit per test: 1 second,
memory limit per test: 256 megabytes

The BUBT playground is quite large. One of the teachers wrote down all the integers from l to r (inclusive) on the playground and decided to assign a task to the students. The assignment is as follows:

There is only one type of operation and In an operation, the following can be done:

Pick two numbers x and y from the playground. Erase them, and in their place write the numbers $3x$ and $\left\lfloor \frac{y}{3} \right\rfloor$. (Here, $\lfloor \cdot \rfloor$ denotes rounding down to the nearest integer).

What is the $\mathbf{minimum}$ number of operations needed to make all numbers on the playground equal to 0?. Solve the assignment to get an AC in the last BIUPC of 2024.

[The teacher has proof that this is always possible in a finite number of operations]

Input

The first line contains an integer t $(1 \leq t \leq 10^3)$ — the number of test cases.

The only line of each test case contains two integers l and r $(1 \leq l < r \leq 10^{12})$.

Output

For each test case, output a single integer — the minimum number of operations needed to make all numbers on the board equal 0.

input
5 1 3 2 4 199999 200000 19 84 1 100000

output
5 6 36 263 1011439

B. Sonkha Totto

time limit per test: 1 second,
memory limit per test: 256 megabytes

The author(You know Who) of this problem loves number theory. So, when deciding about what will be the topic of his BIUPC onsite contest problem, he chose number theory hoping to spread his love for number theory among his beloved juniors. But to his surprise, he wasn't able to come up with an interesting problem for days. Frustrated, he decided to write random conditions, while he was doing this he accidentally wrote a very interesting property and thus the idea for this problem was conceived. The problem goes like this-

Given a sequence A of length n and an integer k,for each i where $1 \leq i \leq n$ find out the number of indices $j(1 \leq j \leq n)$ such that $\frac{LCM(A_i,A_j)}{GCD(A_i,A_j)} = k$

Input

The first line of input will contains two integers $n(1 \leq n \leq 2 \cdot 10^6)-$ the lenght of the sequence A, and $k(2 \leq k \leq 2 \cdot 10^6)$.

The next line contains n integers $A_1, A_2, \ldots\ldots, A_n(1 \leq A_1, .., A_n \leq 2 \cdot 10^6)-$ the elements of the sequence A.

Output

Print n integers where the $i-$th integer is the number of indices $j(1 \leq j \leq n)$ for A_i such that $\frac{LCM(A_i,A_j)}{GCD(A_i,A_j)} = k$

input
7 2 1 9 5 1 13 18 1
output
0 1 0 0 0 1 0

input
9 7 18 10 1 7 17 4 4 7 3
output
0 0 2 1 0 0 0 1 0

C. Majic Number System

time limit per test: 1 second□,
memory limit per test: 256 megabytes

Tasir had always been fascinated by numbers. He spent countless hours pondering their mysteries, trying to create new patterns and systems. One bright afternoon, while doodling in his notebook, he stumbled upon a thought that would change his perspective on numbers forever.

What if I could create a number system that doesn't use all the digits?. What if it only uses the digits 0 and 7?

This sparked a flurry of calculations and scribbles. After hours of work, Tasir invented a unique number system where every positive integer consisted of only the digits 0 and 7 without any leading zeros when written in base 10 . He was ecstatic. He named it the Tasir-7 System.

One evening, Tasir decided to challenge his friends. Can you help him to find the N-th smallest number in the Tasir-7 system?

Input

Contains an integer Number $N(1 \leq N \leq 10^{18})$

Output

Output the result as an integer Number

input
11
output
7077

input
923423423420220108
output
770077070000707070007077077000007070007777007700007077007700

D. Button Bonanza

time limit per test: 1 second,
memory limit per test: 256 megabytes

You have N buttons of random widths that have to be placed on a mobile screen. Given that the width of the screen is W, you have to place the buttons in one or more lines on the screen according to the following rules,

- Start placing buttons from the top left corner of the screen
- You have the freedom to place the buttons in any order
- After choosing a button,
  - If the available space in the current line is equal or more than the width of the button, place the button to the left most of the available space
  - Otherwise, move to next line and start placing buttons from the left side
- For simplicity, assume that there won't be any spacing between the buttons. You do not have to consider the height of the screen.

Your task is to minimize the number of lines occupied by the buttons.

Input

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

The first line of each test case contains two integers $N(1 \leq N \leq 10^5)$ and $W(1 \leq W \leq 10^5)$, the number of buttons and the width of the screen.

Next line contains an array of integers A, where A_i represents the width of the i^th button and $1 \leq A_i \leq W$ for each $1 \leq i \leq N$.

It is guaranteed that the summation of N for all test cases is not greater than 10^5.

Output

Print the minimum number of lines required to place the buttons optimally.

input
2 4 6 3 4 1 2 5 10 2 2 2 2 2
output
2 1

E. Minimum Cost Graph

time limit per test: 2 s.,
memory limit per test: 256 MB

You are given a tree of n nodes connected by n-1 undirected edges, ensuring that there is a unique path between any two nodes. Each node has an assigned weight a_i. For any node v, define $F(v)$ as:

$$F(v) = \sum_{i=1}^{n} a_i \cdot d(i, v)$$

Here, $d(i, v)$ is the number of edges in the shortest path between nodes i and v. Determine which node v minimises $F(v)$. Your task is to compute the smallest value of $F(v)$.

Input

An integer n ($1 \le n \le 10^5$), representing the number of nodes in the tree. Next line given n integers $a[i]$, $a[i+1]$, $a[i+2]$.... $a[n]$ ($1 \le a[i] \le 10^4$), representing the weights of the nodes. Next n-1 line contains two integers u and v ($1 \le u,v \le n$), indicating an edge between nodes u and v.

Constraints

- $1 \le n \le 10^5$
- $1 \le a[i] \le 10^4$

Output

Output a single integer the minimum possible value of $F(v)$.

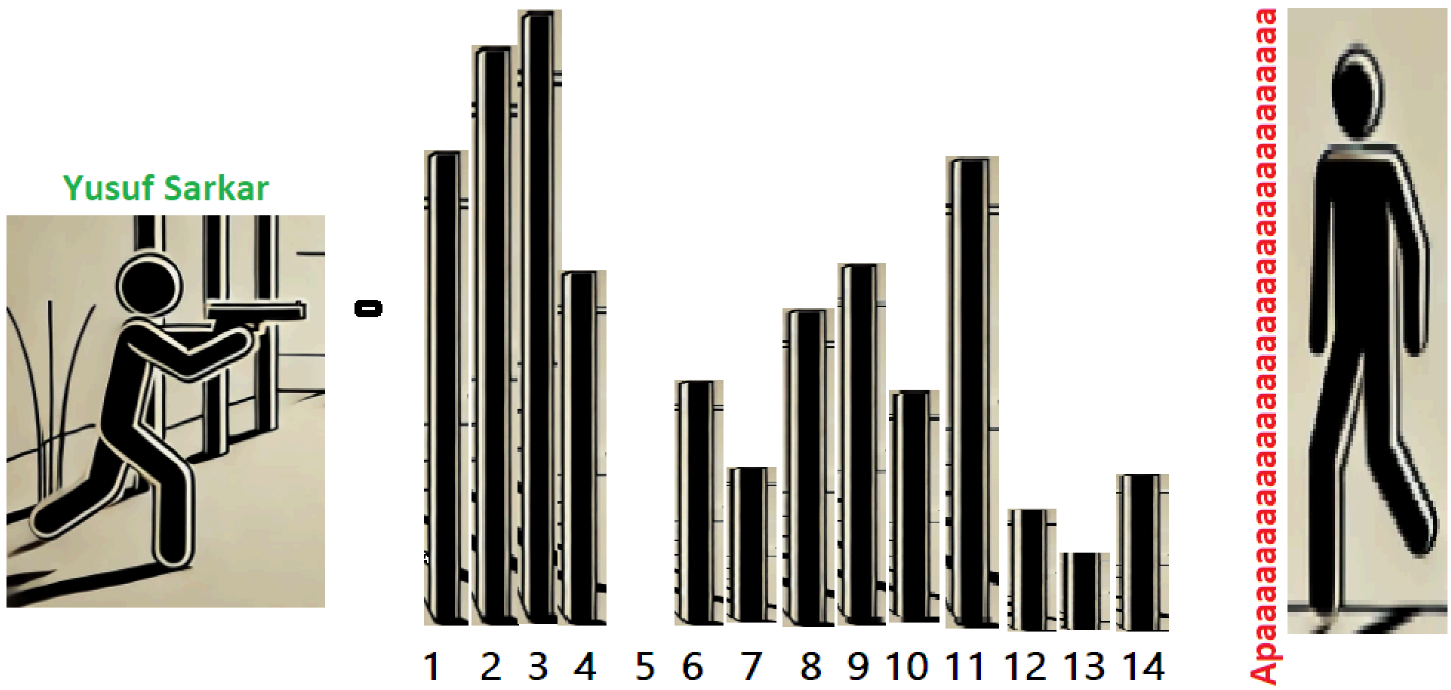
input
5 1 2 3 4 5 1 2 1 3 3 4 3 5
output
14

input
6 1 2 3 4 5 6 1 2 1 3 1 4 1 5 1 6
output
20

F. ইউসুফ সরকার কে এর চরম মুল্য দিতে হবে

time limit per test: 1 second,
memory limit per test: 256 megabytes

Yusuf Sarkar got tired of hearing the phrase 'ইউসুফ সরকার কে এর চরম মুল্য দিতে হবে' and decided to shoot Apaa, but the battlefield isn't straightforward. Between them, there are several pillars with varying heights that might obstruct Yusuf Sarkar's bullets. All the pillars has same strength, and bullets lose power while trying to penetrate the pillars.



Scenario:

1. **Pillars:**
   - There are N pillars between Yusuf and Apaa.
   - Each pillar has a certain height, represented by an array $H_1, H_2, \ldots, H_N$.
   - All the pillars have P strength.
2. **Bullet Dynamics:** There are multiple queries.
   on each query:
   - Yusuf's bullet starts with power S.
   - Yusuf shoots a bullet from a given height L.
   - Any pillar taller than or equal to L will obstruct the bullet.
   - The bullet loses strength P while penetrating one pillar.
   - If multiple pillars obstruct the bullet, the total power lost is the number of such pillars multiplied by P.
     N.B. Each query is independent
3. **Victory Condition:**
   - If the bullet's remaining power after penetrating all the obstructing pillars is greater than 0, Apaa flees.
   - Otherwise, Apaa stays.

Input

The first line contains N — the number of pillars.

The second line contains N integers $H_1, H_2, \ldots, H_N$ — the heights of the pillars.

The next line contains Q — the number of query. The next Q lines contain three integers each:

L — the height from which Yusuf shoots.

S — the initial power of the bullet.

P — the strength of each pillar.

Constraints:

$1 \leq N \leq 10^5$

$1 \leq H_i, L, S, P \leq 10^9$

$1 \leq Q \leq 10^4$

Output

For each query, output one line:

If Hasina flees, print **"Apaa Nai :("** [without quotation].

Otherwise, print **"Apaa Ache :)"** [without quotation].

input
5 4 2 7 5 9 2 6 20 3 3 10 3
output
Apaa Nai :(Apaa Ache :)

For the first shot:

- Yusuf shoots from height 6. The pillars obstructing are $7, 9$.
- Total strength lost: $2 \times 3 = 6$.
- Remaining power: $20 - 6 = 14 > 0$, so Apaa flees.

G. Magic Box

time limit per test: 1.5 seconds,
memory limit per test: 256 megabytes

Problem Setter: G.M Waliullah [Lecturer, CSE BUBT]

In a magic box, there are infinite number of chocolate. Here,

Exactly K number of chocolate's weight is 1 gram,

Exactly K number of chocolate's weight is 2 gram,

Exactly K number of chocolate's weight is 3 gram,

.....

Exactly K number of chocolate's weight is $(N-1)$ gram,

Exactly K number of chocolate's weight is N gram,

and the remaining chocolate's weight is 0.5 gram.

Now you are given Q queries. In each query, you are given a number X. You have to find the largest chocolate which weight is not greater than X. then you have to show the chocolate's weight and throw it out of this magic box.

Input

First line contains three integer N, K, Q.

Next Q line contains a single integer X. Here,

$(1 \leq N, K, X \leq 10^8)$ and $(1 \leq Q \leq 2 \cdot 10^6)$

Output

For each query print the desired chocolate's weight in a single line.

input
5 1 5 1 3 4 3 1
output
1 3 4 2 0.5

input
5 10 1 7
output
5

H. Substring Query

time limit per test: 1 second,
memory limit per test: 256 megabytes

You are given a string S and Q queries. Each query comprises a range $[L, R]$ within the string S and an integer K.

Let, the substring $S[L, R]$ of the string S be the string $P = S_L + S_{L+1} + \ldots + S_R$. Your goal is to find the K'th position in S where the given substring P occurs. If P occurs fewer than K times in S, you should output -1.

Note: The Difference between L and R will be the same for all queries.

Input

The first line contains a string S, consisting only of lowercase Latin letters.

The second line contains an integer Q, representing the number of queries.

The Next Q lines contain a range $[L, R]$ and an integer K.

$Constraints$:

$1 \leq |S| \leq 2 \cdot 10^5$

$1 \leq Q \leq 2 \cdot 10^5$

$1 \leq L \leq R \leq |S|$

$1 \leq K \leq |S|$

Here $|S|$ represents the length of the string S

Output

For each query, print a single integer denoting the K'th position in S where the given substring starts. If the substring occurs fewer than K times in S, print -1.

input
abracadabradabracaabra 5 1 2 3 1 2 5 8 9 1 3 4 1 3 4 4

output
13 -1 1 3 21

For the given string $S =$" $abracadabradabracaabra$ " and the first query $L = 1, R = 2, K = 3$. Let, the substring be $P = S[1, 2] =$" ab " and it occurs in the string S in positions $1, 8, 13$ and 19. For K'th = 3rd position in S where the substring P occurs is 13.

I. Polashs Critical Assessment

time limit per test: 2 seconds,
memory limit per test: 256 megabytes

Polash, an office employee, has been assigned a critical assessment. If he fails to complete it successfully, he will lose his job. The task involves processing an array based on two types of queries

Polash is given an array A of size N and Q queries. Each query is of one of the following types:

Type 1: $1\ \ l\ \ r$ — Calculate and print the sum of elements in the subarray A_l to A_r (1-based indexing).

Type 2: $2\ \ l\ \ r\ \ g$ — Update the array such that for all indices i where $l \leq i \leq r$, the element A_i is replaced by $A_i / gcd(A_i, g)$ where gcd is the greatest common divisor.

Input

The first line of input contains two integers N - the size of the array and Q - the number of queries. The second line of input contains N integers, the elements of the array A.

The next Q lines contain the queries to be processed.

Constraints

$1 \leq n \leq 2 \cdot 10^5$

$1 \leq q \leq 2 \cdot 10^5$

$1 \leq a_i \leq 2 \cdot 10^5$

$1 \leq l \leq r \leq n$

Output

For each query of type 1, output the sum of the subarray A_l to A_r.

input
10 5 3 6 5 9 3 6 11 45 19 3 1 1 3 1 2 5 2 3 5 7 2 3 5 3 1 1 10

output
14 23 102

input
10 5 3 6 9 2 4 6 5 10 7 14 1 1 3 1 2 5 2 1 10 3 2 1 10 2 1 1 10

output

```
18
21
33
```

J. Shizuka and Friends

time limit per test: 1 second,
memory limit per test: 256 megabytes

Shizuka likes spending time with her school friends, sharing laughter and stories during breaks. She has m friends. She loves them so much. Today, she decided to give them some chocolates. She has n boxes of chocolates, $i_{th}(1 \leq i \leq n)$ box has a_i chocolates and h_i happiness value. Initially, Shizuka's happiness is 0. If she chooses i-th box, she has to take all the chocolates from the box and distribute them among her friends. This will increase her happiness by h_i.

However, there's an issue. If any of her friends receive fewer chocolates than any other friend, they will feel **unhappy**. Shizuka doesn't want to cause them any distress.

Now, Shizuka wants to know the maximum happiness she can achieve by distributing the chocolates such that **none** of her friends become **unhappy**.

Input

The first line contains three integers n and m $(1 \leq n, m \leq 10^3)$ — the number of chocolate boxes and the number of friends.

The following n lines contain two integers a_i and h_i $(1 \leq a_i, h_i \leq 10^3)$ — amount of chocolates and happiness value of the i-th box.

Output

Output the maximum happiness Shizuka can achieve by distributing the chocolates so that **none** of her friends becomes **unhappy**.

input
3 4 7 10 3 8 5 5
output
15

input
5 4 5 10 6 12 7 20 8 30 9 25
output
77

input
2 100 5 13 9 8

output
0

In the first test case, Shizuka can select the first and third boxes. There is no optimal option other than this.

In the third test case, There is no way to select boxes, so Shizuka's happiness remains 0.

K. Assignment

time limit per test: 1 second,
memory limit per test: 256 megabytes

Bob has a 5-page assignment that must be submitted within the next 100 minutes. He can write one page in X minutes. Your task is to determine whether Bob can complete all 5 pages of the assignment within the given time limit.

Input

The input consists of a single integer X ($1 \leq X \leq 1000$), representing the time (in minutes) it takes Bob to write one page.

Output

YES if Bob can finish the assignment within 100 minutes. Otherwise, output NO.

input
15
output
YES

input
50
output
NO

L. Task from Batman!

time limit per test: 1 second,
memory limit per test: 256 megabytes

Batman patrols the dark streets of Gotham, determined to protect its citizens from the criminal underworld that lurks in every corner. With his trusted butler Alfred always by his side, Batman finds strength and guidance—though tonight, Alfred is busy with urgent matters, leaving Batman to face the city's darkness alone. As Gotham's shadows grow deeper, Batman must rely on his skills, knowing that Alfred will always be there when he's needed most.

However, there's an urgent problem on his hands—an algorithm that could help him crack a criminal network's code is proving too complex. Batman now turns to you for help, hoping your expertise can solve the puzzle and bring Gotham's enemies to justice before it's too late. He is giving you an array $\mathbf{a}$ of length $\mathbf{n}$.

You have to count the number of $dark$ subarrays in the array a.

A subarray is $dark$ if bitwise OR of it's elements is odd.

† An array b is a subarray of an array a if b can be obtained from a by deletion of several (possibly, zero or all) elements from the beginning and several (possibly, zero or all) elements from the end. In particular, an array is a subarray of itself.

For example, if $a = [5, 4, 1, 2, 3, 6]$, then $[4, 1]$, $[2, 3, 6]$, $[5, 4, 1, 2]$ are some subarrays of a, while $[5, 1, 2]$, $[4, 3]$ and $[1, 2, 6]$ are not.

Input

The first line contains a single integer n $(1 \le n \le 10^6)$ — length of the array.

The second line contains n integers $a_1, a_2, \ldots, a_n$ $(1 \le a_i \le 10^{12})$ — elements of the array.

Output

Output a single integer — the number of $dark$ subarrays in the array a.

input
4 1 4 6 3
output
7

input
3 2 3 4
output
4

input
6 10 8 7 5 6 6
output
15