A. Shizuka and Friends

time limit per test: 1 s.□,
memory limit per test: 256 MB

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.

B. Registration

time limit per test: 1 s.,
memory limit per test: 256 MB

Problem Setter: MD. Saifur Rahman [Chairman, CSE BUBT]

After the preliminary round of the BIUPC, the judges want participants to confirm their registration for the contest. There are n participants ranked based on their preliminary standings. Each participant is given a unique index from 1 to n.

The registration status of these n participants is represented by an array A of length n:

- $A[i] = 1$: The i-th participant has confirmed their registration.
- $A[i] = 0$: The i-th participant has not confirmed their registration.

Participants who fail to confirm their registration leave open spots for participants from the waiting list to join the contest.

You need to determine the **maximum** number of participants from the waiting list who can now register for the contest, assuming that every unconfirmed spot will be filled by someone from the waiting list.

Input

First line contains a single integer n $(1 \leq n \leq 10^5)$ — number of participants ranked based on their preliminary standings.

The next line contains n space separated integers $A[1], A[2], A[3], \ldots, A[n]$ $(0 \leq A[i] \leq 1)$— registration status of those n participants.

Output

Output a single integer — maximum number of participants from the waiting list who can now register for the contest.

input
5 1 0 1 0 1
output
2

input
10 0 0 0 0 1 1 1 1 0 0
output
6

input
3 1 1 1
output
0

C. Majic Number System

time limit per test: 1 s.□,
memory limit per test: 256 MB

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. Minimum Cost Graph(Easy)

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 ($1 \le n \le 2000$), representing the number of nodes in the tree. Next line given n integers $a[i]$, $a[i+1]$, $a[i+2]$.... $a[n]$ ($1 \le a[i] \le 2000$), 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

- $1 \le n \le 2000$
- $1 \le a[i] \le 2000$

Output

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

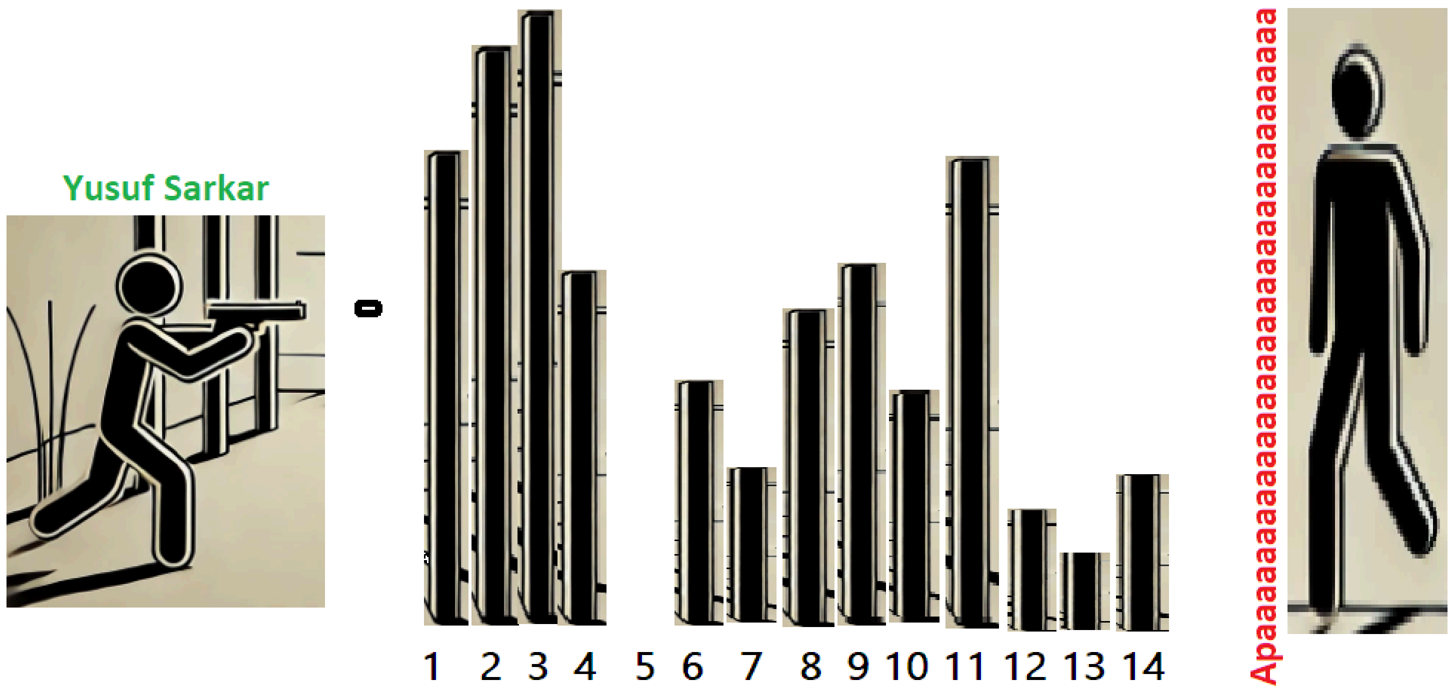
input
5 1 2 3 4 5 1 2 1 3 3 4 3 5
output
14

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

E. ইউসুফ সরকার কে এর চরম মুল্য দিতে হবে

time limit per test: 1 s.□,
memory limit per test: 256 MB

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 $H_1, H_2, \dots, H_N$.
   - 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.

F. Assignment

time limit per test: 1 s.,
memory limit per test: 256 MB

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

G. Task from Batman!

time limit per test: 1 s.,
memory limit per test: 256 MB

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

H. Super Shop Bill

time limit per test: 1 s.□,
memory limit per test: 256 MB

Problem Setter: Ahmed Shafkat [Assistant Professor, CSE BUBT]

Nirob is a young and ambitious student from intake 51, who has a dream of opening his very own super shop. After much planning and hard work, he's finally about to open his shop. However, there's one thing missing – a reliable bill calculating machine!

To help his customers pay the right amount, Nirob needs a machine that can accurately calculate the final bill based on the following rules:

1. There are three types of items in the shop:

- Agriculture items (Type 'A'): These items are tax-free.
- Foreign items (Type 'F'): A 10% tax is applied on the item's price.
- Other items (Type 'O'): A 7.5% tax is applied on the item's price.

2. Additionally, the shop charges a 5% service charge on the basic price of all items (before tax).

Nirob needs your help to create this bill calculating machine that will help him serve his customers with ease!

Input

First line contains an integer n $(1 \leq n \leq 100)$ — the number of items purchased.

The next n lines each contain:

- A character c (either 'A', 'F', or 'O') — the type of item.
- An integer p $(1 \leq p \leq 10^4)$ — the price of a single item.
- An integer q $(1 \leq q \leq 100)$ — the quantity of the item purchased.

Output

Output a single floating-point number representing the total price the customer needs to pay, rounded to two decimal places.

input
3 A 100 2 F 200 1 O 150 3
output
946.25

input
2 F 100 5 A 50 10
output
1100.00

input

```
1
F 500 2
```

output

```
1150.00
```

input

```
1
A 250 4
```

output

```
1050.00
```

input

```
4
O 120 3
F 200 1
A 75 6
O 150 2
```

output

```
1445.00
```

I. Substring Query

time limit per test: 1 s.□,
memory limit per test: 256 MB

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.

J. Rolling Dice

time limit per test: 1 s.□,
memory limit per test: 256 MB

Problem Setter: Maharin Afroj [Lecturer, CSE BUBT]

During a class session, Maharin challenged her students' problem-solving skills. She presented them with an interesting task involving an 8-sided die. The die would be rolled n times, and the students were tasked with determining two values: the maximum possible sum and the minimum possible sum of the numbers obtained from all the rolls.

Input

The only line of input contains a single integer $n(1 \leq n \leq 10^{10})$, the number of times the dice will be rolled.

Output

Output two integers, the maximum and minimum possible sum of the numbers obtained from all the rolls.

input
3
output
24 3

K. Array Partition

time limit per test: 1 s.□,
memory limit per test: 256 MB

You are given an array a containing n integers. Your task is to divide the entire array into three non-overlapping contiguous subarrays, which are subarrays consisting of consecutive elements that do not share any elements while satisfying the following conditions:

1. Each subarray must contain at least one element.
2. Your objective is to minimize the difference between the largest and smallest subarray sums.

Formally, let the sums of the three subarrays be $sum1$, $sum2$, and $sum3$. The goal is to minimize the following expression

$$max(sum1, sum2, sum3) - min(sum1, sum2, sum3)$$

Input

The first line contains an integer n $(3 \leq n \leq 10^6)$, representing the number of elements in the array.

The second line contains n integers, $a_1, a_2, \ldots, a_n (1 \leq a_i \leq 10^9)$.

Output

A single integer representing the minimum possible difference between the maximum and minimum sums of the three subarrays.

input
6 1 2 3 4 5 6
output
3

Optimal Split: $[1, 2, 3]$ $[4, 5]$ $[6]$ → Difference = $9 - 6 = 3$

L. Calf's Journey

time limit per test: 1 s.,
memory limit per test: 256 MB

A playful calf is trying to jump along a coordinate line to reach his favorite grass patch at position n from its starting point at 0. The calf can jump exactly 2 or 3 units in each jump, either forward or backward

Your task is to determine the minimum number of jumps required for the calf to reach the grass patch at position n.

Input

The first line contains an integer $t(1 \le t \le 10^4)$ — the number of test cases. For each test cases contain an integer $n(1 \le n \le 10^9)$ — the position of the favorite grass patch .

Output

For each test case, output one integer — the minimum number of jumps the calf needs to reach the position n.

input
4 1 3 4 12
output
2 1 2 4