AUST CSE Carnival 4.0 Programming Contest - Preliminary (Senior 2)

https://toph.co/contests/training/ubzwydz

Toph

Schedule

The contest will run for **3h0m0s**.

The standings will be frozen for the last **45m0s** of the contest.

Authors

The authors of this contest are ApuOrgho, ishraqfatin7, Paul72, Piru_72, and remon60.

Rules

This contest is formatted as per the official rules of ICPC Regional Programming Contests.

You can use Bash 5.2, Brainf*ck, C# Mono 6.0, C++17 GCC 13.2, C++20 Clang 16.0, C++20 GCC 13.2, C++23 GCC 13.2, C11 GCC 13.2, C17 GCC 13.2, C23 GCC 13.2, Common Lisp SBCL 2.0, D8 11.8, Erlang 22.3, Free Pascal 3.0, Go 1.22, Grep 3.7, Haskell 8.6, Java 1.8, Kotlin 1.9, Kotlin 2.0, Lua 5.4, Node.js 10.16, Perl 5.30, PHP 8.3, PyPy 7.3 (3.10), Python 3.12, Ruby 3.2, Rust 1.57, Swift 5.3, and Whitespace in this contest.

Be fair, be honest. Plagiarism will result in disqualification. Judges' decisions will be final.

Notes

There are 8 challenges in this contest.

Please make sure this booklet contains all of the pages.

If you find any discrepencies between the printed copy and the problem statements in Toph Arena, please rely on the later.

Disclaimer

The contents of this contest have not been reviewed by Toph and do not necessarily represent Toph's views.

A. Monstrous String

Johan Liebert is fascinated by numbers and has posed an interesting problem to his friend, Dr. Tenma. He gives Dr. Tenma a string **S** composed of digits between 1 and 9 (inclusive). Dr. Tenma can insert the '+' symbol into any of the positions between the digits in the string, with the following constraints:

- The '+' symbols must not be consecutive.
- '+' cannot be inserted at the start or end of the string.

Dr. Tenma needs to determine all possible combinations of the string **S** after inserting the '+' symbols, calculate the result of each combination, and then find the sum of all these results.

For example, if **S** ="111", the possible combinations are:

- "1+1+1" which evaluates to 3,
- "1+11" which evaluates to 12,
- "11+1" which evaluates to 12,
- "111" which evaluates to 111.

The total sum of these results is 3+12+12+111=138.

Unfortunately, Dr. Tenma is very bad at solving problems. So he asks for your help. Can you write a program to determine the sum of all possible results for a given string **S**?

Input

A string **S** consisting of digits between **1** and **9**.

Constraints

- $1 \le |S| \le 10$
- All letters in **S** are digits between **1** and **9**, inclusive.

Output

An integer representing the sum of all possible results after inserting '+' symbols in all possible ways.

Samples

Input	Output
111	138

B. Sponsor Hunt

Fatin is a dedicated club member. Recently, he has been looking for sponsors all over.

He has got n sponsors so far. He numbers the sponsors from 1 to n. The amount of money i^{th} sponsor is willing to provide is a_i. The i^{th} sponsor will provide that money on b_i-th month.

Now Fatin wants to know something about the sponsors, so he will ask you q queries, and there are two kinds of queries.
1. He will give you a number x and you should tell him what amount of money he can gather during first x months.
2. He will give you x and y and you should tell him what is the **maximum** amount of money he can gather if he can take at most y sponsors on **each month** during the first x months.

You can take a sponsor only **once** for a query. Now help him find the correct answer.

Input

The first line contains an integer n $(1 \leq n \leq 10^5)$.
The second line contains n integers $a_1, a_2, a_3 ... a_n (0 \leq a_i \leq 10^9)$ - The amount of money i^{th} sponsor is willing to provide.
The third line contains n integers $b_1, b_2, b_3 .... b_n (1 \leq b_i \leq 12)$ - The month when i^{th} sponsor will be able to provide the money.

The fourth line contains an integer $q(1 \leq q \leq 10^5)$ - The number of Fatin's question.

Then follow q line, each line contains a query.
- type x $(type = 1, 1 \leq x \leq 12)$
- type x y $(type = 2, 1 \leq x \leq 12, 0 \leq y \leq 10^5)$

If $type$ is equal to 1 then you should answer for the **first** question else you should answer for the **second** one.

Output

Print q lines. Each line must contain an integer - the answer to Fatin's question. Print the answer in the order of input.

Samples

<u>Input</u>	<u>Output</u>

`5`
`10 20 30 40 20`
`1 2 3 4 1`
`4`
`1 4`
`2 4 1`
`2 4 2`
`2 4 0`
	`120`
	`110`
	`120`
	`0`

Here we can see that,
$1st$ and $5th$ sponsor will provide money on the $1st$ month
$2nd$ sponsor will provide money on the $2nd$ month
$3rd$sponsor will provide money on the $3rd$ month
$4th$ sponsor will provide money on the $4th$ month

In the first query $x = 4$

So he will be able to take all the money during the first 4 month. So the answer will be $(10 + 20 + 30 + 40 + 20) = 120$

In the second query $x = 4$ $y = 1$. So each month he can choose to take money from only 1 sponsor.

On $1st$ month he will choose to take money from $5th$ sponsor
On $2nd$ month he will choose to take money from $2nd$ sponsor
On $3rd$ month he will choose to take money from $3rd$ sponsor
On $4th$ month he will choose to take money from $4th$ sponsor

So he can not choose to take money from the$1st$ sponsor on the $1st$ month as he wishes to maximize the total amount of money. So the answer will be $(20 + 30 + 40 + 20) = 110$

C. Is This A Range Sum Query?

Orgho is fascinated by prime factors. Recently, he learned about prime and composite numbers. Prime numbers are those that are only divisible by 1 and themselves, while composite numbers are divisible by 1, themselves, and at least one other integer greater than 1 and less than the composite number itself.

One day, Orgho came up with an interesting problem and wants you to solve it.

You are given an integer N, which represents the number of distinct prime factors. You are also given a range [L,R]. You need to find the number of integers between L and R(inclusive) that have exactly N distinct prime factors. Note that, 1 has 0 distinct prime factors in it.

Prime factors of an integer are the prime numbers that divide the integer. For example, 12=2×2×3. Here, 12 has two 2s and one 3 as its prime factors, but it has exactly two distinct prime factors: 2 and 3.

Input

- The first line contains a single integer T(number of test cases).
- The next Tlines each contain three integers L, R, and N as described above.

Constraints

- $1 \le T \le 10^5$
- $1 \le L \le R \le 10^6$
- $1 \le N \le 10$

Output

For each test case, output the number of integers between L and R (inclusive) that have exactly N distinct prime factors.

Samples

Input	Output
2 1 10 2 15 30 3	2 1

1. For the first test case, the numbers between 1 and 10 that have exactly 2 distinct prime factors are 6 (2 and 3) and 10 (2 and 5).
2. For the second test case, the number between 15 and 30 that has exactly 3 distinct prime factors is 30 (2, 3, and 5).

D. Horse Travelling

You are currently living in 27 BC - The age of Roman Empire. The Roman Empire was an emperor run period of Ancient Rome which included strongholds and territories across W*estern Asia, Northern Africa*, and various regions of Europe, especially around the *Mediterranean Sea*. The empire was ruled by force, and is famed for having the strongest army in the world, at the time.

But In ancient rome there was only one fast travelling option, The horse. Since you are very fond of travelling, you bought a horse. Your horse can travel u meters in one minute. But unlike you, It has to rest. The horse rests for t minutes after travelling d meters.

You are currently in the heart of Roman Empire, *Rome (Italy*). You want to travel in various cities like *Alexandria, Constantinople*. You are given the total distance to be covered s from rome to the other city. You have to calculate the total time need to reach that city.

Here, s will be a multiple of u and d.

Input

The input consists of multiple test cases. The first line contains a single integer $1 \leq n \leq 10^5$— the number of test cases. Description of the test cases follows.

Each test case contains a single line with four integers u, t, d, s.

$1 \leq (u, t, d, s) \leq 10^9$.

Output

Output a single integer. The total time (minutes) needed to travel.

Samples

Input	Output
1 250 5 1000 80000	715

E. Relief

Nishi lives in Sylhet, which has recently been affected by a flood. Fortunately, Nishi is safe and wants to help by leading the relief efforts. He is collaborating with two individuals, Person1 and Person2:

- Person1 is responsible for distributing relief to males.
- Person2 is responsible for distributing relief to females.

After the relief has been distributed, all the information is provided to Nishi. He then checks if the distribution is **Equitably Distributed**.

A distribution is considered Equitably Distributed if the sum of relief units for any male and female pair lies within the range [L,R] inclusive.

Input

The first line contains four integers N, M, L and R. Here N and M $(1 <= N, M <= 10^6)$ number of males and females people respectively. L and R $(1 <= L, R <= 10^9)$ denote the range.
The second line of each contains N integers $a_1, a_2, ..., a_N (1 \le a_i \le 10^5)$- where $a_i$denote the relief units for the male people.

The third line of each contains M integers $b_1, b_2, ..., b_M (1 \le b_i \le 10^5)$- where b_i denote the relief units for the female people.

Output

Print the maximum number of **Equitably Distributed** relief pairs.

Samples

Input	Output
3 2 5 7 3 2 1 4 5	5
In the provided example, there are five such pairs: [1, 4], [1, 5], [3, 4], [2, 4], and [2, 5]. These pairs satisfy the condition where the sum of each pair lies between 5 and 7 inclusive.	

F. Do You Love Pancakes and Chocolates?

L Lawliet, a huge fan of pancakes, believes that if you use your head, you won't get fat even if you eat a lot of sweets. One day, he went shopping with X yen (the currency of Japan) to buy pancakes and chocolates.

First, he bought one pancake for P yen at a pancake shop. Then, he went to a chocolate shop and bought as many chocolates as possible for C yen each.

Determine how much yen he has left after shopping.

Input

- An integer X representing the total amount of yen L has.
- An integer P representing the cost of one pancake.
- An integer C representing the cost of one chocolate.

Constraints:

- 1≤ X ≤10^{18}
- 1≤ P ≤ 10^{17}
- 1≤ C ≤ 10^{17}
- P + C ≤ X

Output

An integer representing the amount of yen L has left after shopping.

Samples

Input	Output
1000 200 100	0

1. L buys one pancake for 200 yen, leaving him with 800 yen.
2. L then buys as many chocolates as possible with the remaining 800 yen, which is 8 chocolates (each costing 100 yen), spending all his remaining yen.
3. Therefore, he has 0 yen left after shopping.

G. Quick Rescue

In recent times, we have heard about many fire incidents. It's serious, right? So, the AUST authority has taken an initiative to address this. If any incident occurs, they want to clear the campus as quickly as possible to prevent losses. As you are a famous problem-solver, the AUST authority has given you the task of designing the rescue plan. So, what do you actually need to plan? They have given you the positions of the people and the positions of the exit gates. You have to calculate the **total minimum** time required to evacuate everyone from the campus. Here each person can move one unit per second. As soon as a person reaches an exit gate, they can leave instantly without any additional time. Assume that any number of people can exit through a gate at the same time.

Input

The first line contains a single integer $T(1 \leq T \leq 10)$ — the number of test cases.

The second line contains an integer $N(1 <= N <= 10^5)$denoting the number of people and $M(1 <= M <= 10^5)$ denoting the number of exit gates.

Third line will contain N integers $A_1, A_2, \ldots, A_N(1 \leq Ai \leq 10^5)$ where A_i denotes the positions of $i-th$ people.

Fourth line will contain M integers $B_1, B_2, \ldots, B_M(1 \leq Bi \leq 10^5)$ where B_i denotes the positions of $i-th$ exit gate.

Output

Output a single integer - the **total minimum** time needed to empty the campus.

Samples

Input	Output
3 3 2 1 2 3 2 3 3 1 1 2 3 2 3 1 1 1 2 2	1 2 2

Input	Output
For the first test case, there are 3 people at positions 1, 2, and 3, and 2 gates at positions 2 and 3. The person at position 1 can move to position 2 in one second and exit through the gate. The person at position 2 can exit through gate 2 immediately. Similarly, the person at position 3 can exit through gate 3 immediately without delay. Therefore, the total minimum time required is 1 + 0 + 0 = 1 second.	

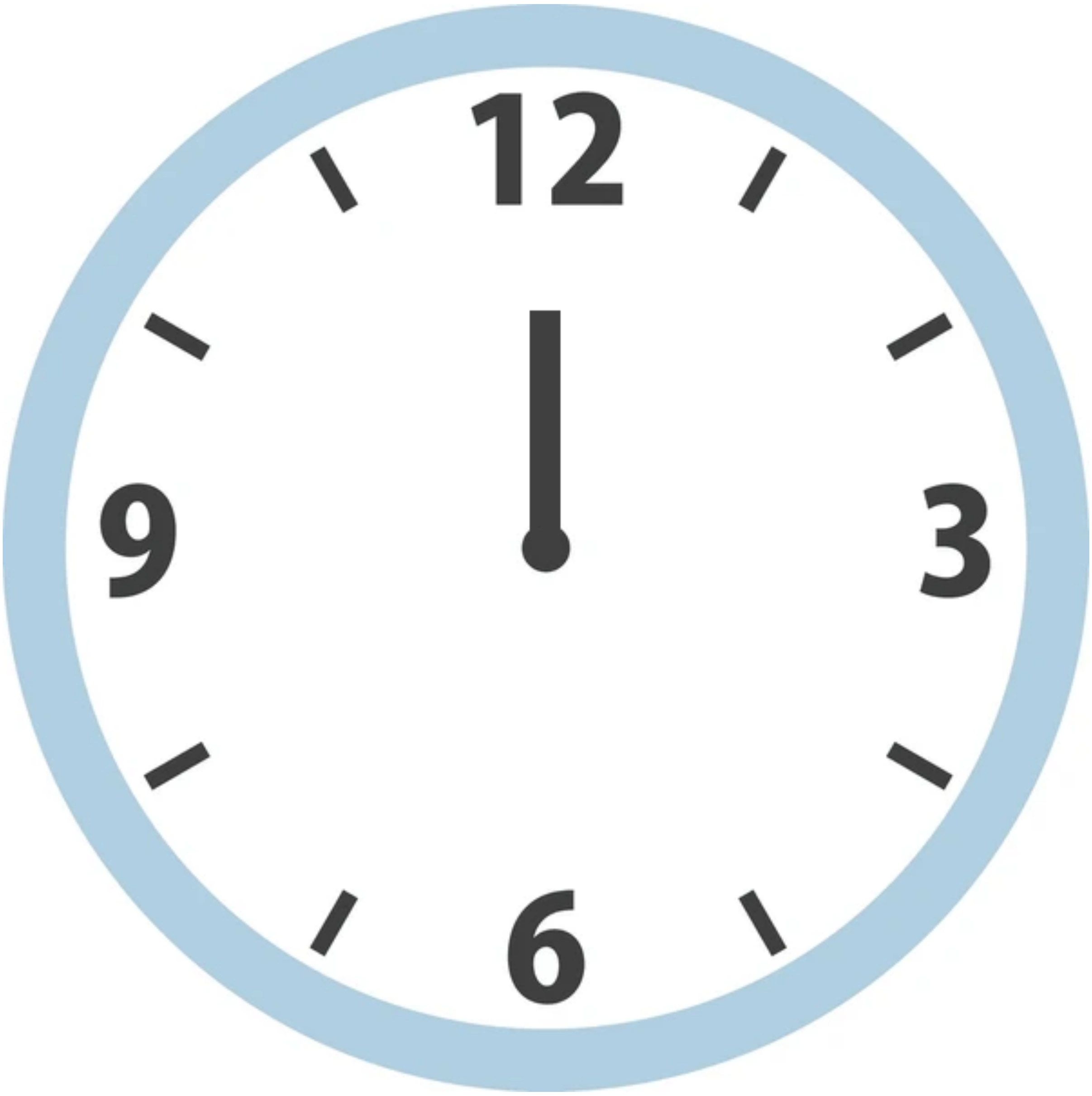
H. Mystery of the Single-Handed Watch

Problem Statement

Tafhim and Raiyan are two friends. Raiyan, an enthusiast of analog watches, stumbled upon a watch with only an hour dial. Despite the absence of a minute dial, Raiyan could deduce the exact time by observing the hour dial's position. Raiyan also knows that the watch is 1 minute slow.

One day, while Raiyan was engrossed in playing CSGO, he suddenly noticed an analog watch in front of him. He found the time shown on the watch very interesting and familiar with the upcoming event **AUST CSE Carnival**. He quickly challenged Tafhim to figure out the time. Tafhim, unfamiliar with analog watches, seeks your help to solve it.

Given that the hour dial of the watch has rotated $\frac{1}{3}$ of a full rotation past the 12 o'clock position and has moved an additional $\frac{3}{4}$ of an hour beyond that, determine the actual time, considering the clock's hour dial position.



Input

There is no input required for this problem. You are to compute the time based on the given conditions.

Output

Output the actual time, considering the clock's hour dial position in the format `HH:MM`, where `HH` represents hours (using a 12-hour clock) and `MM` represents minutes.