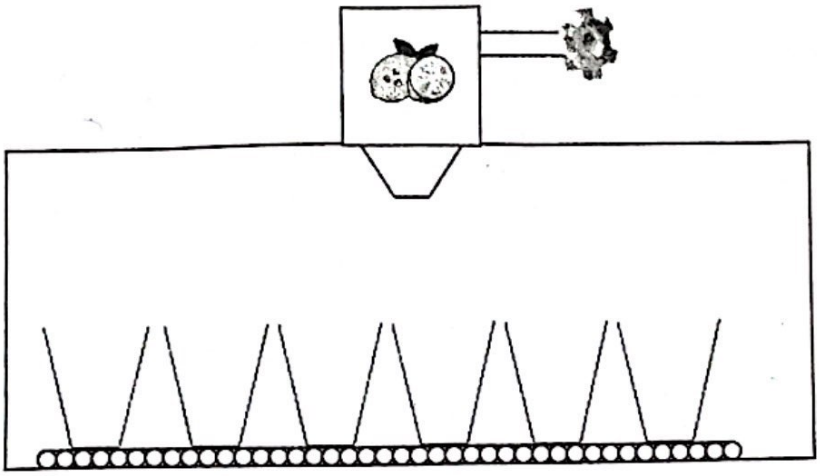
Problem A. Ancient

Input file:	`standard input`
Output file:	`standard output`
Balloon Color:	`Red`

In the hushed sands of time, a story unfolds, tracing back over 5,000 years to the birth of an ancient civilization along the banks of the Nile. Egypt's history, steeped in mystery and grandeur, weaves tales of ancient Egyptians, pyramids, and a timeless legacy that continues to captivate the world to this day.

Farouq, a determined archaeologist, and his friend unite to unravel Egypt's enigmatic past. Their shared journey leads them through the ages, revealing secrets that will reverberate through time, as they uncover the untold tales of the mighty ancient Egyptians and the lost wonders of an ancient world.

Before they start their journey, *Farouq* found a Magical Lemonade Box Machine that works in such a special way! It has a sliding track on which cups walk continuously and it only pours the juice once. That's why *Farouq* wants to make the best profit out of this property - i.e. he must choose the best time to start functioning this machine and stop it -.



The Magical Lemonade Box Machine.

Assume the machine can pour endless amounts of juice and *Farouq* can carry endless **fully filled cups** under a certain weight w. If you know the weight of each cup, can you tell *Farouq* the maximum profit he can get?

Input

The first line will contain the number of test cases t. $(1 \leq t \leq 10^3)$

The first line of each test case contains n and w $(1 \leq n \leq 10^6)$, $(1 \leq w \leq 2.10^9)$ — the number of available cups on the track, and the maximum weight *Farouq* can hold, respectively.

The second line of each test case contains an array of n integers $a_1, a_2, \ldots, a_n$ $(1 \leq a_i \leq 10^9)$ — a_i represents the weight of the $i-th$ cup.

It is guaranteed that the sum of n over all test cases does not exceed 10^6.

Output

The maximum profit *Farouq* can get from the machine.

Example

standard input	standard output
4 5 10 1 2 3 4 5 4 100 3 23 4 10 3 3 1 2 1 5 10 3 6 4 5 2	10 40 3 10

Problem B. Bolbitine

Input file:	`standard input`
Output file:	`standard output`
Balloon Color:	`Black`

In the year 1799, an intriguing stone slab, laden with three unique scripts, surfaced near the town of Bolbitine. An enigma etched in stone, it bore the key to unlocking ancient Egyptian hieroglyphs - the Rosetta Stone. Today, the original stone resides in the British Museum, while a replica stands in its discovery place, the town of Bolbitine, also known as Rosetta or Rashid.

Imagine that you are the soldier who discovered the Rosetta Stone, and you want to transport it to your camp. However, you have to avoid the British troops who are also interested in the stone and have set up checkpoints along the way, so you have to follow some rules:

1. You can step only on **lattice points.**
2. If you are stepping from (x_s, y_s) to (x_e, y_e), there should be no lattice points located on the segment between point (x_s, y_s) and point (x_e, y_e).

Given two-dimensional coordinates, starting point (x_0, y_0), and ending point (x_n, y_n), your job is to go from the starting point to the ending point in **N-steps.**

It is guaranteed that there is an answer.

Input

The first line contains two integers a and b ($-10^4 \le a, b \le 10^4$) — the coordinates of the lattice point where you start.

The second line contains two integers c and d ($-10^4 \le c, d \le 10^4$) — the coordinates of the lattice point where you end.

The third line contains a single integer N ($2 \le N \le 10^5$) — the number of steps.

Output

Print n lines.

On the i–th line, print two integers x_i and y_i ($-10^9 \le x_i, y_i \le 10^9$) separated by a space, denoting your location (x_i, y_i) after the i–th step. $x_n = a$, $y_n = b$ must hold.

If there are multiple solutions, output any.

Examples

standard input	standard output
0 0 2 2 2	1 1 2 2
3 4 4 0 3	4 4 6 1 4 0

Note

A lattice point is a point at the intersection of two grid lines in a regular 2D coordinate plane. As (1, 1) or any point in a coordinate plane.

The starting point and ending point could be the same point.

Problem C. Cleopatra

Input file:	`standard input`
Output file:	`standard output`
Balloon Color:	`Light Blue`

Once upon a time, there was a legendary queen named Cleopatra, ruling over ancient Egypt around 2,000 years ago, from 51 BCE to 30 BCE. Her story seems so ancient to us now. But wait, here's a fascinating twist! The great pyramids, those incredible structures, were already standing tall roughly 3,000 years before her era! That means her time is much closer to ours than it was to the age of those majestic pyramids. It's mind-boggling to think just how ancient those pyramids truly are!

Abanoub and *Basha* are young mathematicians who are fascinated by the ancient Egyptian civilization and its secrets. they have heard that Cleopatra was not only a powerful queen, but also a brilliant scholar who studied mathematics, astronomy, and medicine. Inspired by her, they came up with a new mathematical challenge for their friend *Ayoub*:

Count how many good numbers p exist such that $(2 \le p \le n)$.

A number X is good if and only if:

4

- let s be the minimum number of divisions to convert X to a prime number, $s+1$ is a prime number.

Here is a trick: you can only divide X by prime numbers.

Can you help *Ayoub* solve this problem and win the challenge?

Input

The input contains only one integer n $(2 \le n \le 10^6)$.

Output

Print one integer — the count of good numbers p such that $(2 \le p \le n)$.

Examples

~~24~~ 3

standard input	standard output
4	1
32	18

Note

In the first test case, the only good number is 4, it takes one division to make it prime: $4/2 = 2$, so the number of divisions $s = 1$,

Since $s + 1 = 2$ is a prime number, then 4 is a good number.

1 5

2 5

Problem D. Djoser

Input file:	`standard input`
Output file:	`standard output`
Balloon Color:	`Purple`

In the ancient land of Egypt, a mighty king named Djoser reigned. His name echoed across time, for he embarked on a remarkable endeavor—the construction of a groundbreaking step pyramid at Saqqara. With the ingenious architect Imhotep at his side. This grand monument heralded a new era in Egyptian funerary architecture, captivating the imagination of people throughout history, including *Farouq*, the young archaeologist.

Farouq is visiting the Saqqara pyramid complex, he decided to explore the area. However, he soon realized that he have brought too many items with him, such as water bottles, snacks, and souvenirs. He doesn't want to carry them around anymore, and he wants to dispose of them properly.

He now has a bunch of n trash items that he should get rid of. Fortunately, there exist many trash bins in the area, so he'll inspect for the nearest bin to him and throw his trash to it. For such a trash bin, he knows its radius r and how far its center is placed from him x.

Farouq is setting on bench, *Farouq* will throw the items from his initial position at a certain distance d_i for each trash item he has. Also, he can change his position by at most $|k|$ from his initial position on the bench, but now comes the problem... Not all items are thrown directly inside the bin and some fall outside it, which in turn means he must pick them up himself. That's why he started to think why not change the perimeter of bin borders so that all thrown trash fit into it?

For simplicity of calculations, a perimeter is calculated as a summation of **integer lengths only** for the constructed shape's outer border in which vertical and horizontal distances will have a length of 1, and the diagonal distance of only $\sqrt{2}$ will have a length of 1.

Can you get the **minimum** new perimeter required for this task -if needed-?

Note: a trash item fits into the bin if it is inside the borders of the bin or lies on it and the perimeter is not necessarily to be regular.

Input

The first line contains n, x, r, $|k|$. The number of trash items ($2 \leq n \leq 10^5$), how far the trash bin's center is placed from him, the radius of the bin ($1 \leq r, x \leq 10^7$), and possible change in his position from the initial one ($0 \leq |k| \leq 10^7$).

Followed by n lines, each containing the k_i and d_i, change in position and the distance by which the i^{th} item is thrown $(-k \leq k_i \leq k), (1 \leq d_i \leq 10^7)$

Output

Print the **minimum** new perimeter required if there are trash items that lie outside the bin, otherwise print "-1" without the quotes.

Examples

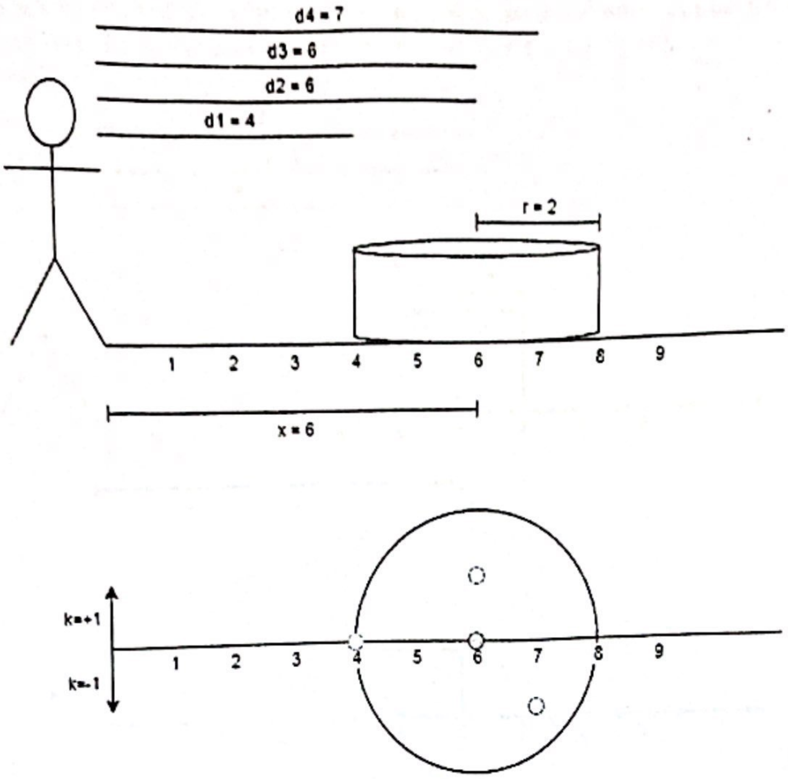
standard input	standard output
4 6 2 1 0 4 0 6 1 6 -1 7	-1
5 6 1 1 1 7 -1 7 1 5 -1 5 0 6	8
3 6 1 1 1 4 0 6 1 6	5

Note

For simplicity, imagine you're in a 2D coordinate plane and the below examples will assist your understanding.

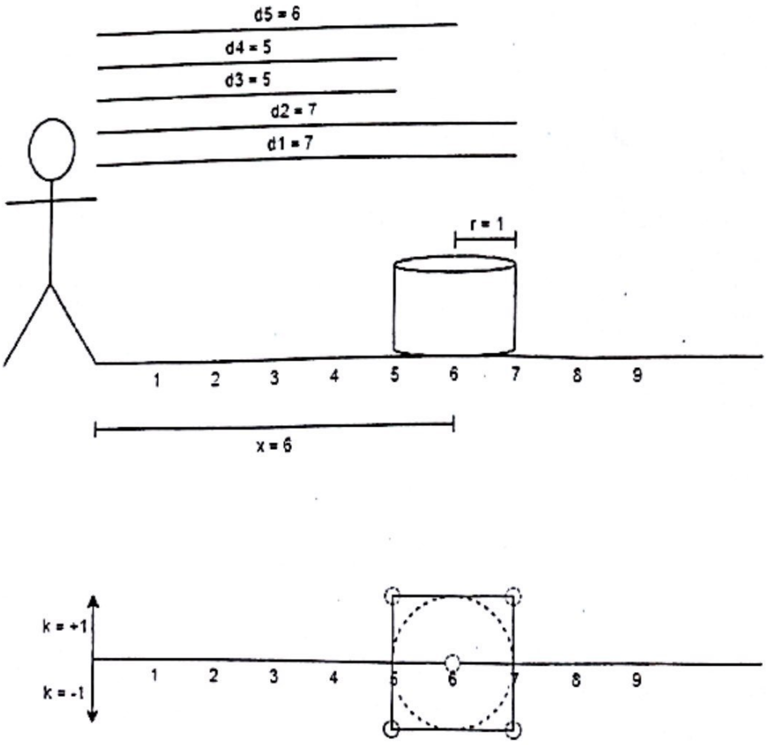
In the first example:

- The first item is thrown with a distance equal to 4 from the initial position.
- The second item is thrown with a distance equal to 6 from the initial position.
- The third item is thrown with a distance equal to 6, 1 distance away from the initial position.
- The fourth item is thrown with a distance equal to 7, -1 distance away from the initial position.



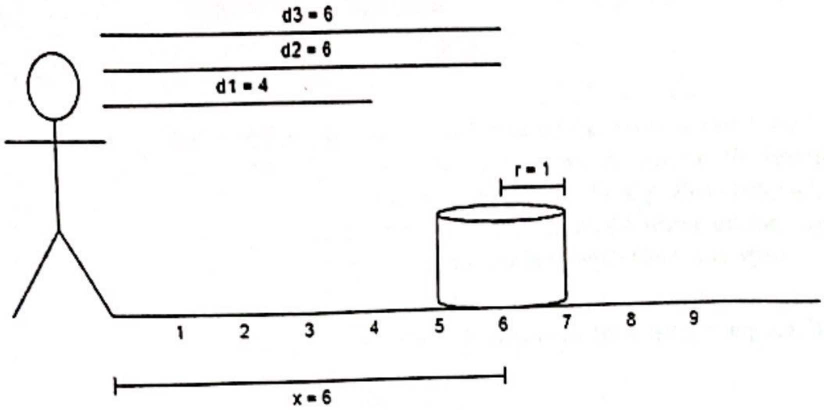
Side and Plan view for the first case.

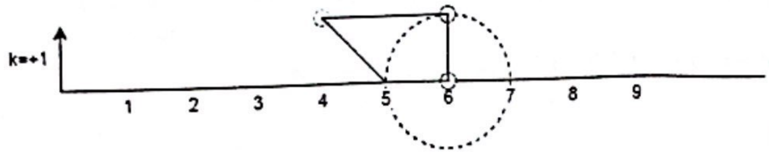
In the second example, after throwing all items but one fell outside the bin, then we will construct the minimum perimeter of $(2+2+2+2) = 8$



Side and Plan view for the second case.

In the third example, after throwing all items, we found an item that fell outside the bin, then we will construct the minimum perimeter of $((\sqrt{2} \equiv 1) + 1 + 1 + 2) = 5$





Side and Plan view for the third case.

Problem E. Edfu

Input file:	`standard input`
Output file:	`standard output`
Balloon Color:	`Dark Green`

In a cozy café, *Zezo* and his friend *Ahmed* chatted excitedly about their love for ancient Egypt. *Ahmed* talked about Edfu, a city known for its big, well-preserved temple. He described how the temple was made from huge blocks of stone, set out like a giant map of the sky. *Zezo* listened, wide-eyed, particularly taken by the idea of an ancient tool used to measure the Nile's water during the flood season. Together, they dreamed of traveling to Egypt to see these wonders with their own eyes.

As they talked, *Ahmed* presented a problem for *Zezo* to solve: Given n books where the $i-th$ book has a weight w_i, is it possible to select some books with total weight equals W in such a way that **exactly** one of them is of even number?

Help *Zezo* solve this problem.

Input

The first line contains integers n and W $(1 \le n \le 50)$, $(1 \le W \le 25000)$ — the number of the books, and the required total weight.

The Second line contains n integers $w_1, w_2, \dots w_n$ $(1 \le w_i \le 500)$ — w_i represents the weight of the $i-th$ book.

Output

In one line print "YES" if it's possible to select some books with total weight W. Otherwise, print "NO" — without quotes.

You can output the answer in any case (for example, the strings "yEs", "yes", "Yes", and "YES" will be recognized as a positive answer).

Example

standard input	standard output
4 50 22 50 23 5	YES

14 18

even

$\boxed{4}$

even + even

even

odd + odd

even + odd

Problem F. Fayoum

Input file: `standard input`
Output file: `standard output`
Balloon Color: `Yellow`

In the western desert of Egypt, lies the ancient city of Fayoum, where history spans not only thousands of years but millions. A city that is filled with countless natural and historical wonders such as Lake Qarun and Wadi Al-Rayyan. Of all the interesting archaeological sites, you found Wadi Al Hitan to be the most extraordinary site that reveals a rich fossil record of ancient marine life from the Eocene epoch. Among its wonders are fossilized whale skeletons and bones, revealing precious insights into the ancient ancestors of today's magnificent whales.

As you marveled at the wonders of Wadi Al Hitan with *Barakat* and *Marzooq*, you made a game for them.

But first, you gave the game referee 3 numbers l, r, and w and an array a of n elements such that a_i represents the number of balls in box_i, and asked the game referee to construct 3 new boxes box_x, box_y, box_z.

where:

- box_X has A balls where A is the minimum number in the range $[l, r]$ inclusive.
- box_Y has B balls where B is the maximum number in the range $[l, r]$ inclusive.
- box_Z has w balls.

Don't worry the referee has an **infinite** number of balls.

Marzooq and *Barakat* will play the game on the newly constructed boxes, *Barakat* starting first.

In one turn the player will perform the following operation:

- choose any non-empty box of the **three newly constructed** boxes.
- choose number T where $1 \le T \le$ (number of balls in the chosen box).
- remove T balls from that box.

The player who is unable to perform the operation loses.

Each player plays optimally.

You are given q queries. Solve each independently where for each query you are given l, r, and w.

For each query, output a single string **YES** if *Barakat* wins. Otherwise, if *Marzooq* wins print **NO**.

Input

The first line of the input contains two integer numbers, n and q ($1 \le n, q \le 2 \cdot 10^5$).

The second line of the input contains n integer numbers $a_1, a_2, \ldots, a_n$ ($1 \le a_i \le 10^9$) — the sequence itself.

Following q lines, each will contain three integers l, r, and w ($1 \le l \le r \le n$), ($0 \le w \le 10^9$).

Output

For each query, output a single string **YES** if *Barakat* wins. Otherwise, if *Marzooq* wins print **NO**.

Example

standard input	standard output
6 3 1 3 1 6 7 1 2 3 0 1 1 0 1 4 7	YES NO NO

Problem G. Giza

Input file:	`standard input`
Output file:	`standard output`
Balloon Color:	`Gold`

In ancient history, Giza city witnessed humanity's most astounding achievement—the Great Pyramids. Over 4,500 years ago, the majestic structures of Khufu, Khafre, and Menkaure rose on the Giza Plateau, Egypt. As tombs for powerful kings, these remarkable pyramids showcased ancient Egyptian ingenuity and precision. Enduring through time, these iconic monuments captivate the world, a testament to the enigmatic allure of ancient engineering prowess.

You are an archaeologist who is exploring the ancient pyramids of Egypt. You have discovered a secret chamber that contains a mysterious device that looks like an elevator. You wonder how it works and what it was used for. However, you soon realize that the device has some strange rules:

- It takes exactly ten seconds to move from one floor to another.
- It can only change directions once.

You also notice that there are N other archaeologists on different floors who have also found the device and want to use it, every floor has exactly one person to serve, and those persons call the elevator at the same time.

Print the **order of the floors** the elevator must serve to have the minimum average waiting time so that everyone can get to the chamber as soon as possible.

Note that:

- **The waiting time is the time the elevator takes to reach the floor to be served.**
- **If the average time is the same no matter the direction, the elevator goes up.**

Input

The first line contains two numbers separated by a space.

- N $(0 \le N \le 10^5)$ the number of floors to be served.
- E $(0 \le E \le 10^5)$ the floor at which the elevator starts.

The Second line contains an array of N integers $A_1, A_2, \ldots, A_N$ $(0 \le A_i \le 10^5)$ which are the floors to be served.

Output

Print the **order of the floors** the elevator must serve to have the minimum average waiting time. The output should be in one line separated by space.

Examples

standard input	standard output
`3 4` `0 1 2`	`2 1 0`
`3 4` `0 1 6`	`6 1 0`

Problem H. Heliopolis

Input file:	`standard input`
Output file:	`standard output`
Balloon Color:	`White`

Long ago, the city of Heliopolis stood tall in what is now Cairo. It had big stone pillars that reached for the sky, showing people's interest in the stars and the sun. It was a place full of learning and big ideas. Heliopolis still leaves a mark on history, inviting us all to dig deeper and learn more about the past.

As you reflect on the legacy of Heliopolis, you are faced with a new challenge.

Given a matrix with N rows and M columns containing only 1s or 0s, the task is to find the maximum area of the largest rectangular plot of land that can be formed within a submatrix of the field starting at cell $(1, 1)$ and ending at cell (X, Y). The plot of land must lie entirely within two parallel rows or columns of the same length consisting only of 1s and having the maximum combined length.

It is worth noting that the area of the rectangular plot of land is always twice the length of the longest of these two parallel rows or columns. There are Q queries, each consisting of two integers X and Y. For each query, the goal is to determine the area of the largest rectangular plot of land that can be formed within the given submatrix and within the two parallel columns or rows that consist of only 1s and have the maximum combined length.

Input

The first line contains three integers N, M, and Q, denoting the number of rows, number of columns, and number of queries, respectively ($2 \le N \le 1000, 2 \le M \le 1000, 1 \le Q \le 10^5$).

The next N lines contain M integers each, denoting the cells of the field (0 or 1).

Each of the next Q lines contains two integers X and Y ($2 \le X \le N, 2 \le Y \le M$) — denoting the submatrix for the query.

Output

For each query, output a single integer representing the area of the largest rectangular plot of land that can be formed in the given submatrix and within the two parallel columns or rows that consist of only fertile cells and have the maximum combined length.

Example

standard input	standard output
5 5 2 1 1 1 0 1 1 0 1 1 0 1 1 0 1 1 1 0 1 1 0 1 1 1 0 1 5 5 2 5	6 4

Problem I. Imhotep

Input file:	`standard input`
Output file:	`standard output`
Balloon Color:	`Silver`

Imhotep was a brilliant man who served as the chief minister, architect, and physician of King Djoser in ancient Egypt. He designed the first step pyramid at Saqqara, which was a marvel of engineering and a symbol of royal power. He was also skilled in astronomy, mathematics, and medicine, and wrote many books on various subjects. Imhotep was facing a complex problem, but unfortunately, he passed away before he could solve it. Now, it's your mission to solve Imhotep's complex problem.

Let n be an integer that represents the summation of the ASCII values of each character in a string, for example, string "abca": ASCII value of 'a' is 97, 'b' is 98, 'c' is 99. So, the value of n that represents this string is: $97 + 98 + 99 + 97 = 391$.

You are given n and a string s consisting of lowercase characters only, count how many strings can be constructed (using lowercase Latin letters only) by the given n, with the additional constraint that the constructed string shouldn't start with any character in the given s.

Input

The first line contains n where ($1 \le n \le 10^5$) — represents the summation of ASCII values of characters of any constructed string.

The second line contains the string s where ($0 \le |s| \le 26$) consists of lowercase letters only.

Output

Output the number of ways to construct the string mod M where $M = 10^9 + 7$.

Examples

standard input	standard output
97 b	1
97 a	0
195 cdef	2

Problem J. Jewels

Input file: `standard input`
Output file: `standard output`
Balloon Color: `Orange`

Diving into ancient Egypt's rich past, we've discovered so many jewels. Yet, much more is still hidden waiting for you to uncover. Take a tour on the Nile. Ancient wonders abound, a tapestry of history. Carved in stone, secrets veiled in mystery. A journey through time, on the majestic Nile.

Osama and *Mo* went shopping and they had a Coupon with a value of N. They wanted to take full advantage of this coupon, but this coupon has rules of use.

- **rule 1**: You can't buy 2 different items with the same coupon.
- **rule 2**: You can buy any quantity of an item you want but **take care of the third rule**.
- **rule 3**: If the total price is more than the coupon price you will pay the difference from your pocket.

Osama and *Mo* want to utilize the use of this coupon, they want to pay the least money from their pockets and want to use the coupon as much as they could.

Osama and *Mo* decided that paying the difference would be very difficult for them, so they would rather have any item that won't make them pay anything more than an item that would utilize the use of coupon but make them pay from their own pockets. The more they pay the difference from their pocket the less utilization of the coupon.

Also, if two items have the same utilization of the coupon but one is more expensive they will choose it over the cheaper one.

The owner of the shop will give them the price list of all the items in the following form $[price_1, price_2, ...]$ where item i has $price_i$. Also, he will give them q operations each operation is identified with a number.

- **Operation 1:** Print the item that has the best use of the coupon in the list. If there are no items, print '-1'.
- **Operation 2:** Given an extra number X, which is a new item's price, you're required to add it to the list. If this item is already on the list, ignore it.
- **Operation 3:** Given an extra number X, which is an item's price, you're required to remove it from the list. If the item isn't currently in the list, print "Can't Delete". Otherwise, print "Deleted Successfully" — without quotes.

Important Note : It's guaranteed that all prices in the list are unique even after adding new items. And that the size of the list won't exceed 10^6.

Input

The first line of the input contains integer numbers N, s, and q ($1 \leq N, s, q \leq 2 \cdot 10^5$).

The second line of the input contains s integer numbers $price_1, price_2, \ldots, price_s$ ($1 \leq price_i \leq 10^9$) — the sequence itself.

Following q lines each will contain either only an *operation number* where ($op = 1$) or *operation number* and X where ($2 \leq op \leq 3$), ($1 \leq X \leq 10^9$).

Output

For each $operation_1$, output the price of the item that causes the best use of the coupon currently in the list.

For each $operation_3$, output "Deleted Successfully" if the given item X is in the list. Otherwise, output "Can't Delete" — without quotes.

After you answer all the queries, output the remaining items (if any exist) from best to worst item separated by spaces and followed by a new line.

Examples

standard input	standard output
20 8 4 20 15 1 8 3 21 30 40 1 3 3 3 2 2 2	20 Deleted Successfully Can't Delete 20 2 1 8 15 21 30 40
10 1 4 2 1 3 10 3 2 1	2 Can't Delete Deleted Successfully -1

Problem K. Karnak

Input file:	`standard input`
Output file:	`standard output`
Balloon Color:	`Dark Blue`

In the ancient ruins of Karnak City near Luxor, *Hamada* was amazed. The magnificent Karnak Temple Complex, stood before him. The grand temples and statues showcased the brilliance of ancient Egyptian architecture. As he explored the sacred site, *Hamada* imagined the elaborate religious ceremonies that once made Karnak the spiritual heart of Egypt. The whispers of history enveloped him, urging him to uncover the secrets of this timeless place.

Inspired by the ingenuity and dedication of the ancient Egyptians, *Hamada* turned his attention to a new challenge:

Hamada wants to travel straight from the origin $(0, 0)$ to point (x, y).

Because He isn't good at planning, he needs your help to know how many lattice points he will pass by when he travels to point (x, y) from the origin.

Lost in thought, *Hamada* wondered what other secrets lay hidden within Karnak's walls. Can you help him solve the problem and unravel the mysteries of the ancient world?

Input

The first line contains a single integer t $(1 \le t \le 500)$ — the number of test cases. The description of each test case is as follows.

Each test case's first and only line contains two integers x and y $(1 \le x, y \le 10^9)$ — the coordinates of the lattice point where *Hamada* is heading to.

Output

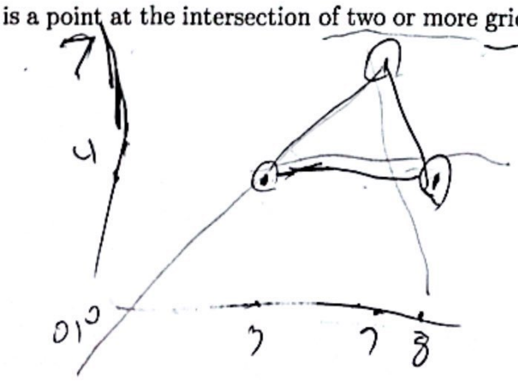
For each test case, print the number of lattice points *Hamada* will visit in order to reach his destination.

Example

standard input	standard output
3 3 4 8 4 7 7	0 3 6

Note

A lattice point is a point at the intersection of two or more grid lines in a regularly spaced array of points.



Problem L. Luxor

Input file:	`standard input`
Output file:	`standard output`
Balloon Color:	`Bronze`

Luxor, the "Hundred Gates City boasts the iconic Luxor Temple at its heart. Dedicated to the Theban triad of Amun, Mut, and Khonsu, this ancient masterpiece showcases grand statues, towering obelisks, and intricate hieroglyphics. A living relic of human civilization, the temple exudes an ethereal charm, especially when illuminated by the setting sun.

One day, our friend $Moaz$ was driving his car on his way to visit Luxor Temple when he noticed something strange. There is a workshop where people can fix their cars in m minutes per car. It can only fix one car at a time and there is no time taken between cars. Can you tell $Moaz$ the total number of minutes to fix n cars?

Input

The first and only line of input consists of two integers n and m $(0 \leq n, m \leq 100)$

Output

The total number of minutes to fix n cars.

Examples

standard input	standard output
2 2	4
7 5	35

Problem M. Memphis

Input file:	`standard input`
Output file:	`standard output`
Balloon Color:	`Pink`

In ancient Egypt, Memphis stood as the first capital of the unified country, founded around 3100 BC by King Menes. This influential city witnessed the rise and fall of many dynasties and kings, flourishing as a cultural and artistic hub. Grand monuments, temples, and statues adorned its landscape. However, Memphis's significance waned with the ascent of Thebes and Alexandria, eventually leading to its abandonment and burial under the sand.

The people at Memphis are initially given a list of n orders, each row of the list consists of the following: the ID of the customer who ordered, the time he arrived (t), the excepted time duration of the order (d), and finally the class of the customer (C) (lower number means higher class) which is the most important in the row because we live in a class society -of course-.

Memphis people would do anything to serve the class society system, to the extent of cutting less-class people's orders because higher-class people just arrived. Also, Memphis people are lazy, at any point in time, if they faced multiple orders from customers with the same class, they simply just start with the one that was mentioned earliest in the original list.

Now your task is to try to mimic their job. let's test how well you will perform.

Input

The first line of the input consists of one integer n ($1 \le n \le 10^5$) — the number of the new orders that you are asked to plan.

The next n lines consist of the following:

Each line contains four numbers ID, t, d, and C ($1 \le ID \le n, 0 \le t, C \le 10^9, 1 \le d \le 10^9$), the ID of the customer, the time he has arrived, the excepted time duration of the order, and the class of the customer (lower number means higher class).

Output

Output multiple lines, every line refers to a single activity against time (Note that the activity could be the execution of a whole order or a fraction of it).

Each activity consists of two numbers: $startTime$ and $endTime$ of the activity.

Example

standard input	standard output
3 1 0 8 3 2 2 9 2 3 3 5 10	0 2 2 11 11 17 17 22

Note

We can observe in the example the following:

- Customer 1 arrives at t = 0 and demands an order that takes 8-time units. the first 2 seconds of orders will be executed until t = 2.
- Customer 2 arrives at t = 2 and because of his higher priority compared to customer 1. the execution of the order of customer 1 will be interrupted by customer 2, his order (customer 2) will be executed without any interruption as it has the highest priority among the 3 customers.

- At $t = 11$, customer 1 will continue the rest 6 unit times of his order, that happened because of his higher priority compared to customer 3.
- Finally at $t = 17$, after both customer 1 and customer 2 have finished their order completely. customer 3 who has the lowest priority among the 3 customers will be served his 5 unit times.

Problem N. Nefertiti

Input file:	`standard input`
Output file:	`standard output`
Balloon Color:	`Light Green`

Enthralled by Nefertiti, an ancient Egyptian queen, you found inspiration in her influential role during the Amarna Period. As King Akhenaten's principal wife, she played a powerful role in state affairs and became one of ancient Egypt's most visible queens. Her captivating bust, unearthed in 1912, stands as a timeless masterpiece of ancient Egyptian art. However, many aspects of Nefertiti's later life and fate continue to intrigue historians and scholars, and of course, you.

Being curious about the mysteries surrounding Nefertiti's fate, and where her tomb is located, you decide to join a team of bounty hunters who are searching for clues about Nefertiti's tomb in an abandoned temple.

The Bounty Hunters group consists of n people. You found the abandoned temple. However, as soon as you enter the temple, you hear a weird noise behind you and see a timer appear on the door with an initial value of T. You realize that the door will close forever after the timer expires and that the group has to act quickly to get out with some items.

The goal of the group is to maximize the value of the items they can get without any of them getting stuck. So, they have to choose which one to get which item.

All of the group are initially at the door. There are exactly n items, and each person can only carry one item. People chosen must run to the items at the same time, each person grabs an item, and then run back before the timer expires. Also, some people may not enter at all, but each one runs for an item **can only do it once** - i.e. Any person who grabbed an item and returned can not go back into the temple-.

Here is the information you need to know:

- The $i-th$ item has a weight w_i.
- The $i-th$ person has an original time t_i that he needs to run from the door to any item.
- The time that the $i-th$ person needs to get back to the door after grabbing the $j-th$ item is calculated as follows:

Time taken carrying $j-th$ item = Original Time (t_i) + Item Weight (w_j) / k (integer division)

Where k is a constant also given in the input.

Can you help the group and tell them the maximum total values of items they can collect?

Input

The first line will have three integers n, T, and k. ($1 \le n \le 10^6$), ($1 \le T \le 10^9$), ($1 \le k \le 100$) — The number of people in the group (it's the same as the number of items), the time on the timer, and the constant affects the time of running back to the door while carrying an item, respectively.

The second line will have n integers $t_1, t_2, \ldots, t_n$ ($1 \le t_i \le 10^9$) — t_i represents the original time needed by the $i-th$ person to reach any item.

The third line will have n integers $v_1, v_2, \ldots, v_n$ ($1 \le v_i \le 10^9$) — the value of each item.

The last line will have n integers $w_1, w_2, \ldots, w_n$ ($1 \le w_i \le 10^9$) — the weight of each item.

Output

Output one integer, the maximum value that can be got.

Problem O. Oases

Input file:	`standard input`
Output file:	`standard output`
Balloon Color:	`Light Purple`

In the vast expanse of Egypt's desert, secret havens, Siwa, Kharga, Dakhla, and Bahariya, emerged as mirage-shrouded oases. Each, home to the resilient Date Palm, stood as a beacon of life amidst aridity. These sanctuaries provided refuge to weary travelers, their rejuvenating powers etched in age-old legends. Wrapped in an enigma, they awaited a deserving wanderer to discover their true potential. And so unfurls the tale of Egypt's elusive oases, each a mystery nestled in the desert's heart.

While you were reading about the beauty of the oases in Egypt, you were faced with a new challenge:

Given an array A consisting of n elements, and an index i $(1 \leq i \leq n)$.

Print the GCD of all the elements of the array: GCD $(A_1, A_2, ..., A_n)$, excluding A_i.

Note that the GCD of elements $(A_1, A_2, ..., A_n)$ indicates the greatest common divisor which divides all of them.

Input

The first line contains a single integer n $(2 \leq n \leq 10^5)$ — the size of the array.

The second line contains n integers A_1, A_2, ..., A_n $(1 \leq A_i \leq 10^9)$ — elements of the array A.

The third line contains an integer q $(1 \leq q \leq 10^5)$ — the number of queries.

Each of the next q lines contains an integer i $(1 \leq i \leq n)$ the index that should be excluded when calculating the GCD.

Output

For each query, output the answer to that query on its own line in the order the queries were made.

Example

standard input	standard output
5 2 3 6 15 3 2 1 5	3 1

Note

In the first test case: the output of the first query is GCD(3, 6, 15, 3) which equals 3. And the output of the second query is GCD(2, 3, 6, 15) which equals 1.