Code To Save A Mother

RUET Analytical Programming Lab (RAPL)

https://toph.co/c/code-to-save-a-mother

Schedule

The contest will run for **4h0m0s**.

Authors

The authors of this contest are adnan_toky, ashraful9194, faria_efa, freedom_seeker, Humayra_037, masood, Nahid001, OnikJahanSagor, rifatrraazz, Sanjida1903070, santo_ruet, sayeef006, and Tonoy1010x.

Rules

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

You can use Bash 5.0, Brainf*ck, C# Mono 6.0, C++11 GCC 7.4, C++14 GCC 8.3, C++17 GCC 9.2, C11 GCC 9.2, Common Lisp SBCL 2.0, Erlang 22.3, Free Pascal 3.0, Go 1.13, Haskell 8.6, Java 1.8, Kotlin 1.1, Node.js 10.16, Perl 5.30, PHP 7.2, PyPy 7.1 (2.7), PyPy 7.1 (3.6), Python 2.7, Python 3.7, Python 3.8, Ruby 2.6, 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 10 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.

A. Welcome to Code to Save a Mother Contest!

Greetings and **Welcome** to the “Code to Save a Mother” contest. This contest is created to contribute towards the medical expenditure of our beloved friend Muntasir Fahim’s mother. With all of your help, we have reached our milestone and already handed over the money to him. Thank you to everyone for your participation. We showed everyone the strength of our competitive programming community.

Print anything wishing his mother a safe and healthy recovery.

Input

There is no input.

Output

Print anything wishing Muntasir Fahim’s mother a safe and healthy recovery.

B. Adorable String <3

There is a tie for 1st position in **House Cup Tournament** between house **Gryffindor** & **Slytherin.** The House Cup Tournament is a competition between the Houses of **Hogwarts.** The house which will win the cup, can contribute more to **Code To Save A Mother** event. **Professor Severus Snape** has given a task to **Harry** (Gryffindor House) & **Malfoy** (Slytherin House), who will solve the task first, his house will win the cup & the members of that house can contribute more in the event. The task is hard for Harry. So he asked your help to solve the task. Can you solve the task for Harry?

Let's call a string adorable if its number of consonant(s) is 1 more then its number of vowel(s).

The letters ***"a, e, i, o, u"*** are vowels & others are consonant.

For more clarification, The string **'abc'** is adorable because it has 2 consonants & 1 vowel and **'b'** is also an adorable string as it has 1 consonant & 0 vowel.

You're given a string. Your task is to count the number of adorable substring(s).

Input

Input starts with an integer T $(1 \leq T \leq 500)$, denoting the number of test cases.

The first line of each test case contains the length of string n, $(1 \leq n \leq 5000000)$.

It is guaranteed that the sum of n does not exceed 5000000

The second line of each test case contains the string with only lower case letters.

Output

For each case, Output one integer denoting the number of adorable substring(s).

Samples

Input	Output
3 4 abcd 8 abcdefgh 10 aabcdeefgh	4 11 17

C. Love for Mother

Muntasir is a university student. His mother is struggling with 3 blocks in her heart and her condition is very serious. He lost his father long ago. For the treatment of his mother, he needs a big amount of money. That's why he does a lot of tuitions and besides tuitions, he also teaches at a coaching center. However, Muntasir loves teaching. He always tries to encourage his students to study.

He teaches math at coaching. Sometimes he gives some interesting maths to his students. Now, we will talk about one of these maths. The math statement is like that "You have 3 types of chocolates and a, b, c numbers of each type. You want to distribute the chocolates among the **maximum** number of children in such a way that after distributing each type of chocolates **equally** there will be x, y, z numbers of chocolates left of each type."

For more clarification "If you distribute the chocolates among n children and give p, q, r numbers of each type to each child then $n \times p + x = a$, $n \times q + y = b$ and $n \times r + z = c$. Now, you have to maximize n."

Note: After distributing, each child must get at least one chocolate of each type.

As a programmer, can you find the answer to the math before Muntasir's students?

Input

Input starts with an integer $T(1 \leq T \leq 100)$, denoting the number of test cases.

The first line of each test case contains three integers $a, b, c(1 \leq a, b, c \leq 10^{18})$.

The second line of each test case also contains three integers $x(0 \leq x \leq a), y(0 \leq y \leq b), z(0 \leq z \leq c)$.

Output

For each test case, If it's impossible to fulfill the requirements, print **"Impossible"**(without quotes). Otherwise, print one integer, the **maximum** number of children who will get chocolates.

Samples

Input	Output
2 29 50 75 1 2 3 1 1 1 1 1 1	4 Impossible

D. Convincing Customers

As a good friend of Muntasir, Evan was finding a way to help him bear the cost of his mother's treatment. So today, he has opened a charity bag shop.

Now, he has n bags to sell. You are given an array a of n integers, where a_i is the size of the i-th bag. He has somehow managed to know that there will be **exactly** q customers today and the i-th of them will search for a bag of size s_i and will donate c_i amount of money if he buys the bag.

But Evan is amazing at convincing people! (just like you when it comes to programming!) As people are not so easy going nowadays, he has already calculated the convincing value (x_i) for every i-th customer, which means that he will be able to convince the i-th customer to buy a bag of size p if $max(1, s_i - x_i) \leq p \leq s_i + x_i$. He can also ignore any customer whether convincing that customer is possible or not. **No customer will buy more than one bag and Evan can't sell a bag twice or more.**

What is the maximum amount of money Evan will be able to collect at the end of the day, if he sells bags optimally?

Input

The first line contains a single integer $n(1 \leq n \leq 200)$, the number of bags available in the shop.
The second line contains n integers $a_1, a_2, \ldots, a_n (1 \leq a_i \leq 10^9)$ - the sizes of each available bag.

The next line contains one integer $q(1 \leq q \leq 200)$, the number of customers.

The following q lines contain three integers each: the preferred size of the i-th customer $s_i (1 \leq s_i \leq 10^9)$, the amount of money $c_i (1 \leq c_i \leq 10^9)$ he will donate if he buys the bag and his convincing value $x_i (0 \leq x_i \leq 10^9)$.

Output

Output one integer, the maximum amount of money Evan can collect today to help Muntasir.

Samples

Input	Output
3 5 10 15 3 5 3 0 7 10 5 19 5 4	18

E. Area 51

Toky and Efa are playing an interesting game. A brief description of the game is given below.

Suppose h, w and a are the *height*, *width* and *area* of a rectangle respectively. So $a = h * w$. Efa has kept the *area* secret and Toky has to find it by guessing *height* and *width*. Don't worry, Efa also gives the maximum possible value of h and w. That means, Efa gives Toky an integer n and he has to guess two integers h and w between 1 and n $(1 \leq h, w \leq n)$ such that $h * w = a$. But Toky doesn't have much time to finish the game because he wants to participate in *Code To Save A Mother* and Toky also doesn't want to make Efa sad. So, he wants you to play with her as you are the only one here without them.

Now, here is the interesting part. You don't have to find the area by one chance. In reply of your guessing, Efa will tell your guessed area is *equal* or *less* or *greater* than the hidden area. But she doesn't like to talk much. So, she will reply your guess not more than 20 times. Can you find out the area?

Input

The first line of the input is an integer, t $(1 \leq t \leq 1000)$ denoting the number of test cases means how many times Efa wants to play.

Each test case has an integer, n $(1 \leq n \leq 1000)$ denoting the maximum possible value of h, w.

You have to start guessing after receiving n. To inform your guess, you have to print in the following way (without quotes).

"? h w"

After printing, you will have to read one character based on the following incidents.

- If $h * w > a$, you will receive a character "$>$" (without quotes)
- If $h * w < a$, you will receive a character "$<$" (without quotes)
- If $h * w = a$, you will receive a character "$=$" (without quotes). After receiving this, you have to print "$= a$" (without quotes) and exit from current test case immediately.

- If you try to guess more than 20 times or after receiving "$=$" or print invalid characters (any character without $?$ and $=$), invalid numbers (non-positive or $h > n$ or $w > n$), you will receive a character "!" (without quotes) and you must terminate your program after receiving this. If you terminate, you will receive *Wrong Answer* otherwise you will get any arbitrary *non-accepted verdict*.

This is an interactive problem. So, You have to flush output after printing each line. In C++, you should use **endl** or **fflush(stdout)** after printing each line. For other languages, see the language documentation.

Output

Let, $t = 1$ and $n = 6$, here is a sample interaction for $a = 16$.

```
1
6
      ? 1 2
<
      ? 5 6
>
      ? 4 4
=
      = 16
```

F. Sum & Multiple

RAPL community is raising money for the treatment of Muntasir's mother. So we have a list to do this charitable work. But we find it difficult. Help us to do it perfectly.

You are given the list of the number of donors from different institutions. There are n institutions and i^{th} institution has a_i donors (where $1 \leq i \leq n$). Institutions are ordered alphabetically in the list. We hope that every institution has at least one more donor than its previous institution on the list.

Muntasir has a lucky number k. We hope that the total number of donors in the list will be a **multiple** of k. You can **increase** the number of donors of any institution. So your task is to find the **minimum** number of **additional** donors we need to match the criteria and print the final list.

Input

The first line contains two integers $n(1 \leq n \leq 10^5)$ and $k(1 \leq k \leq 10^5)$ representing the number of institutions and Muntasir's lucky number respectively. The next line contains n space-separated integers $a_i(0 \leq a_i \leq 10^5)$ representing number of donors from i^{th} institution.

Output

In the first line, print a single integer representing the number of minimum **additional** donors we need. In the next line, you should print the final list that matches the given criteria. If there are multiple such lists, print any of them.

Samples

Input	Output
5 40 2 4 1 3 9	21 4 6 8 10 12
NOTE: Here, our aim is to make the sum of all elements a multiple of 40. The sum of our output array (4+6+8+10+12=40) is a multiple of 40. Moreover we have at least one more donor from every institution than the previous one.	

G. Rational Monkey

There lived a wise monkey in the forest of Amazon. One day he started searching the jungle for bananas as he was very hungry. Suddenly he saw a line of n **banana trees** (numbered from 1 to n). He was very wise that's true but he was a bit lazy as well. For that reason, he decided that he would collect bananas from **exactly** x banana trees and the difference of position between any two trees among these chosen x trees should be **less than** k. **He collected all bananas of those** x **banana trees.**

However, It's not necessary that the bananas will be fresh. They can be **rotten** as well. The monkey has also **c magic spells**. He can use these spells **not more than c times**. He can convert any tree containing rotten bananas to a tree of fresh bananas using just one magic spell.

Let, the monkey collected f fresh bananas and r rotten bananas. What is the **maximum** value of $(f - r)$ the monkey could get?

Note: See the input-output section & explanation of sample output for more clarification.

Input

The first line contains four integers n, k, x, c $(1 \leq x \leq k \leq n \leq 10^5, 0 \leq c \leq 10^5)$, the number of banana trees, the range, the number of trees the monkey decided to choose and the number of magic spells, respectively. The following line contains n integers $a_1, a_2, a_3 ..., a_n$ $(-10^6 \leq a_i \leq 10^6)$ the bananas in the i^{th} tree (Here, $(a_i < 0)$ means that the i^{th} tree contains $|a_i|$ rotten bananas. And $(a_i \geq 0)$ means that the i^{th} tree contains $|a_i|$ fresh bananas).

Output

Print the desired answer in a single line.

Samples

Input	Output
5 3 2 0 -1 -9 -3 -1 -2	-3

Input	Output
5 3 2 1 -3 -6 3 -8 10	18

Explanation of sample output 01: In the range $[3, 5]$ the monkey can choose trees at the positions 4 and 5. These two trees both contain rotten bananas and he cannot convert them into fresh bananas as he has 0 magic spells. So,total rotten bananas, $r = 1 + 2 = 3$, total fresh bananas, $f = 0$. The value of $(f - r) = 0 - 3 = -3$. We can show that this is the maximum value of $(f - r)$.

Explanation of sample output 02: In the range $[3, 5]$ the monkey can choose trees at the positions 4 and 5. The tree at position 4 has rotten bananas while the tree at position 5 has fresh bananas. The monkey can spend his one spell in the tree at position 4 to convert them into fresh bananas. So, total fresh bananas, $f = 8 + 10 = 18$, total rotten bananas, $r = 0$. The value of $(f - r) = 18 - 0 = 18$.We can show that this is the maximum value of $(f - r)$.

H. Protik and Hill Climbing

Who doesn't want to save a mother? Like everyone, Protik was finding a way to earn some money (he has no money right now) so that he can donate. He found an event of hill-climbing which says that the person who will be able to climb to the peak of "Wakanda" hill, will be rewarded with prize money. Protik likes hill-climbing very much and he is on a vacation too. So Protik decided to participate and donate the money which he might get from that event.

The hill is a steep hill and there are N rocks (numbered from 1 to N) and to climb to the peak of the hill Protik has to use some or all of those rocks. The heights of all the rocks **from the ground** are given. The rocks with maximum height among them are at the peak of the hill which means Protik has to reach any of the rocks with maximum height. Protik can grab any rock if the difference between that rock's height and Protik's height from the ground is at most M and he will always climb upwards. Protik has an initial weight of W kg. Each rock has a maximum weight capacity that it can hold which means if Protik's weight is greater than the rock's weight capacity, then he can not use that rock to climb. Each rock also has a magical power b_i. After reaching i^{th} $(1 \leq i \leq N)$ rock, his weight will become $max(1, W - b_i)$, (b_i can be both negative and positive integer) and he will continue with this weight. After this weight change, if his weight becomes greater than that rock's weight capacity, he will still be able to stay on that rock. **Initially, Protik is on the ground and when he is on any rock, the rock's height from the ground will be his height from the ground.**

Protik doesn't have much time. He needs to be sure that he can climb to the peak of the hill otherwise he will need to find another way of earning some money. So he needs your help because you are a good programmer. You have to tell him whether he can climb to the peak of the hill or not.

Input

The first line of the input consists an integer $T(1 \leq T \leq 10^4)$, the number of test cases.

First line of each test case contains 3 space separated integers N, M and W $(1 \leq N \leq 10^6, 1 \leq M, W \leq 10^9)$.

The next 3 lines contains N space separated integers.

1. $h_1 \leq h_2 \leq \cdots \leq h_N (1 \leq h_i \leq 10^9)$, height of each rock from ground.

2. $a_1, a_2, \ldots, a_N (0 \leq a_i \leq 10^9)$, maximum weight capacity of each rock.

3. $b_1, b_2, \ldots, b_N (-10^9 \leq b_i \leq 10^9)$, magical power of each rock.

It is guaranteed that sum of N over all test cases will not exceed 10^6.

Output

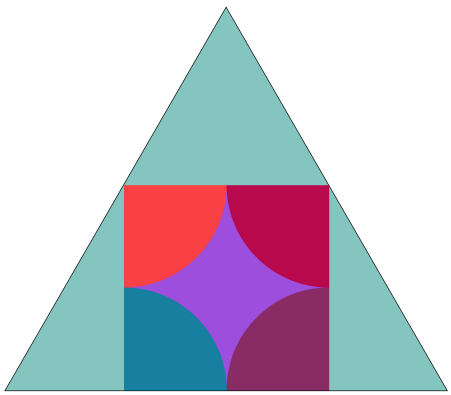
Print *YES* if Protik can win the event otherwise print *NO*. You can print *YES* / *NO* in any case. For example, "Yes", "yes", "YES", etc are equivalent.

Samples

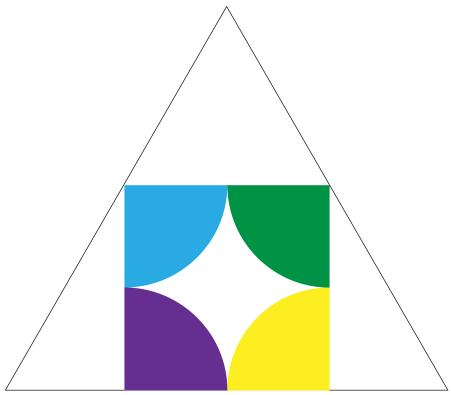
Input	Output
2 3 5 10 3 6 10 12 10 5 3 2 1 4 5 10 3 8 13 17 5 5 4 3 -5 2 -2 2	YES NO

I. Coloring

RAPL had arranged an abstract art competition before this contest to raise funds for Fahim's Mother. In that competition, equilateral triangle shaped papers were given. An artist named Robo drew a square having **maximum area**. Then he drew four **equal** segments of a circle inside that square and **colored them with blue, green, purple and yellow color** and submit to the judging panel. Robo's friend named Hobo also drew the same art but colored every parts of the art. Since judges like too much colored area, Hobo became the champion.



Hobo's Drawing



Robo's Drawing

Now RAPL has arranged a programming contest. They decided to give the **perimeter** of those papers. As a contestant in this contest, you have to find how much **extra area is colored** in Hobo's drawing than Robo's.

Note: Robo use only blue, green, purple and yellow color (ignore the border :v) .

Input

The first line contains a single integer, t ($1 \leq t \leq 10^5$) — the number of test cases.

Each test case contains a single integer, x ($1 \leq x \leq 10^9$) — the perimeter of those papers.

Output

For each test case, output one line containing the resulting area.

Your answer is considered correct if its absolute or relative error does not exceed 10^{-6}.

Formally, let your answer be 'a' and the jury's answer be 'b'. Your answer is accepted if and only if $\frac{|a-b|}{max(1,|b|)} \leq 10^{-6}$

Samples

Input	Output
2 1 100000	0.0293161721 293161720.7264060531

Assume, $\pi = cos^{-1}(-1)$

J. Battle of FurfuriNagar

The country **FurfuriNagar** is famous for the cartoon series **Motu Patlu**. The peace and furfuriness of the country were unbearable for the world's most terrifying gang **Jingalala**. Therefore they have decided to take control over the country.

There are n cities numbered from 1 to n in FurfuriNagar. Each pair of cities are connected with exactly one simple path. More formally the cities maintain a tree structure. The gang Jingalala consists of x terrorists. They want to start attacking from any arbitrary city of FurfuriNagar and take control over as many cities as possible. The i-th city has a_i armies to protect the city. When the gang attacks a city, if the number of armies of that city is greater than the number of terrorists who attacked the city, then all the terrorists are killed and they can't win that city. Otherwise, they kill half of the armies and the remaining armies join their gang. (Formally, they kill $\lceil \frac{a_i}{2} \rceil$ armies of the i-th city). Thus they win a city. After winning each city, the gang members of that city get divided into groups and attack the adjacent unattacked cities. The gang is not allowed to visit a city twice.

They know the map of FurfuriNagar and the number of armies of each city. Now it's time to find the optimal way to divide the gang members after winning each city so that they can win as many cities as possible. Now for each city, they want to know the maximum number of cities they can win, if they start attacking from that city. As you are a good programmer, they kidnapped you to solve the task.

Input

The first line of the input contains two integers $n(1 \leq n \leq 100)$ and $(1 \leq x \leq 10^9)$, denoting the number of cities and the number of terrorists.

The next line contains n space-separated integers $a_1, a_2, \ldots, a_n (1 \leq a_i \leq 10^9)$ denoting the number of armies of each city.

Each of the following $n-1$ lines contains two integers u and v $(1 \leq u, v \leq n, u \neq v)$ denoting a road between the city u and v.

Output

Print n space-separated integers. The i-th integer should be the maximum number of cities they can win if they start attacking from the i-th city.

Samples

Input	Output
7 3 3 2 4 4 5 1 2 1 2 2 3 2 4 1 5 5 6 5 7	3 5 0 0 0 1 1

Input	Output
5 3 4 1 2 5 3 1 2 1 3 1 4 1 5	0 1 4 0 4