EtCPC 2024 Kickoff Division 2

A. ALxU

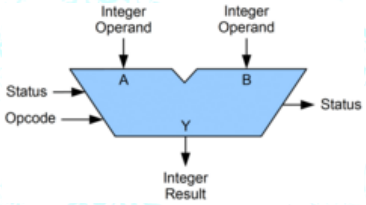
1 second, 256 megabytes

In computers, addition is performed by the Arithmetic Logic Unit (ALU). The two numbers you add together will be stored in two separate registers. The ALU will add these two numbers and store the result in another record. Depending on the size of the registers, this addition can be performed on **8-bit, 16-bit, 32-bit** or **64-bit** numbers.

Here's how the ALU performs addition:

1. The CPU acquires operands, often from a storage register.
2. The CPU routes the operands to the ALU's input registers.
3. The CU creates opcode input, which says "perform addition".
4. The ALU adds the first operand to the second operand using sequences of OR, AND, and XOR gates.
5. The ALU stores the result in another record.

The ALU includes storage places for input operands, operands that are being added, the accumulated result, and shifted results.



ALU

The downside to this operation is overflows. That is when adding two very large **16-bit** numbers, the result will a **32-bit**, if we add two very large **32-bit** numbers the result will be **64-bit**, and if we add two very large **64-bit**, the result will be **128-bits** and so on, so on…

That is why ALX introduced the new CPU with a new ALU called ALxU(Arithmetic Logic excellent Unit), which prevents the addition process from overflowing. This is done by using the modulo operation which is efficiently implemented and integrated to the CPU and ALxU unit. In simple terms given two integers a and b, it returns $((a\%M) + (b\%M))\%M$, where $M = 1,000,000,007$.

Note: If you are using C++, make sure to declare the variables a and b as long long instead of int otherwise, it would result in an overflow error.

Input

The only line in the input is two numbers a and b, ($0 \le a, b \le 2^{62}$)

Output

The sum of the two numbers, with the modulo operation as defined above.

input
6 2

output
8

input
8917861915551 59962623793830

output
485227221

If you are using C++, make sure to declare the variables a and b as long long instead of int otherwise it would result in an overflow error.

B. Buna Adegegna

1 second, 256 megabytes

The highly anticipated football match, awaited for so long by passionate fans, finally unfolded with an electrifying atmosphere and intense moments on the field.

The long-awaited football match between Buna and Barcelona concluded with a spectacular and tremendous victory for Buna. The pitch echoed with cheers and celebration as Buna's players showcased their exceptional skills, scoring numerous goals against Barcelona's defence. The crowd erupted in joy with each goal, witnessing a dominant performance that exceeded all expectations. Buna emerged triumphant, leaving an indelible mark on the match and solidifying their place in the hearts of their ecstatic fans.

While this is fantastic news for Buna F.C fans, unfortunately, Alex missed the chance to watch or attend the match, leaving him with a sense of disappointment. Fortunately, his friend Alem attended the match and she kept a record of the scores on a sheet and gave it to him.

The record is a list of numbers a_i, the minute where the ith goal was scored.

For example: $[12, 30, 40]$ indicates the first goal was in the $12th$ minute, the second in the 30th minute and the last goal was in the $40th$ minute.

But Alex doesn't want to watch the whole game as he is busy working on his project. He only wants to watch some of the goals that he heard were spectacular. He knows what number goals are they, but he doesn't know at what minute they occurred.

So given the list of number goals he wants to watch, write a program that returns the minute it occurred.

Input

The first line of input consists $n(1 \le n \le 10^5)$, indicating number of goals scored. Following line contains n numbers $a_i(1 <= a_i <= 1000)$, where a_i indicates the minute the ith goal was scored.

The third line input consist q ($q < n$). Queries, number of goals Alex wants towatch. Which is followed by q lines, each line consits a number q_i indicate the number goal Alex wants to watch.

Output

For each query return the minute, that goal was scored.

input
4 11 43 49 49 4 2 1 4 3

output
43 11 49 49

input
8 16 16 21 22 24 30 30 62 5 1 2 5 2 3

output
16 16 24 16 21

C. The Marathon

1 second, 256 megabytes

Alex is a beginner marathon athlete. As he is starting his career, he is not performing well in the contest and he gets anxious about his rank while on a contest. Luck for him, the contest provides real-time data of the distance covered by each participant through a smartwatch given to every contestant. However, it is hard for him to tell how many participants are in front of him by looking at all the given data, which might be distracting. So, he came to you to write him a piece of code that tells how many participants are in front of him given the distance covered by every participant.

Input

The first line indicates the number of participants $n(2 \le n \le 10^6)$. The second line contains n integer $a_i(0 \le a_i \le 500000)$ the distance covered by each participant.

The first distance is Alex's distance.

Output

Print a single integer indicating the number of participants in front of Alex.

input
3 231646 398487 454559
output
2

input
7 174431 24501 392367 27916 207993 233601 163515
output
5

D. Plindromes in disguise

1 second, 256 megabytes

Alex is a writer and he loves palindromes. One day he decided to see if what he had written can be turned into a palindrome by removing spaces from his sentences and reordering the letters. But as his paragraph can be long it appears it would be way to difficult to do it by hand. So, he comes to you for help. Your task is given a list of words to return "Yes" if it can form a palindrome otherwise "No" if it cannot.

Input

The first line contains a single integer $t(1 \le t \le 100)$, denoting the number of test cases. Each test case consists of an integer $n(1 \le n \le 1000)$, number of words and on the following line contains n words, where each word $|s_i| \le 1000$ which means the length of the word does not exceed 1000.

Output

"Yes" without the quotes if it is possible otherwise "No" without the quotes.

input
3 2 tikus biskut 3 this is the 4 here we go again
output
Yes Yes No

E. Kind of decimal kind of binary

1 second, 256 megabytes

Let's call a number as both decimal and binary, if it is digits are made of a 1 and 0 only. For instance, 100101, 1111, 1000, 0, and 1 are all considered both decimal and binary while 1123, 9783, 99 and 30 are not.

Your task is given a number k, write it as a sum of decimal binary numbers(the numbers might not be distinct). From such representation return the smallest number of binary decimals required to do that.

Input

The first line contains a single integer $t(1 \le t \le 1000)$, denoting the number of test cases. The only line of each test case contains a single integer $n(1 \le n \le 10^9)$, denoting the number to be represented.

Output

For each test case, output the smallest number of binary decimals required to represent n as a sum.

input
3 312 4 10010000
output
3 4 1

F. Show me the pairs

1 second, 256 megabytes

Given a list a of size n, find how many ordered pair of (i, j) which satisfy the following conditions

- $i \neq j$
- $\frac{a(i)-a(j)}{i-j} = 1$

Input

The first line of input consists $n(1 \le n \le 10^5)$, indicating the size of the list.

The second line n elements of the array. $a_i(0 <= a_i < n)$

Output

Print the number of ordered pairs that satisfy the above condition

input
5 3 1 3 2 4
output
2

G. Admission Calculator

3 seconds, 256 megabytes

In the renowned Adama Science and Technology University (ASTU), a unique and rigorous selection process is established to identify the brightest minds for admission. This process combines students' performance in the national exam, accounting for 40 The evaluation of each candidate's scores is meticulously done to normalize the results. The scores in the Mathematics section of the entrance exam are scaled to a maximum of 60 points, and those in Physics to 45 points. This normalization ensures that the total score from the five subjects aligns with the 60 For this task, you are provided with the variables S (student name), X (national exam score), M (Mathematics score), P (Physics score), C (Chemistry score), B (Biology score), and E (English score). The goal is to process the data for a given number N of students, where each line of input contains these variables. The national exam score X ranges from 0 to 700, and the entrance exam scores (M, P, C, B, E) each range from 0 to 30. The name of the student is maximum 10 upper case English letters.

Input

The input begins with an integer $N(1 \le N \le 10^5)$, representing the number of students. Following this, each of the next N lines contains the variables S, X, M, P, C, B, and E, detailing the student's name and their scores in the national and entrance exams. $0 \le X \le 700$, $0 \le M, P, C, B, E \le 30, 1 \le |S| \le 10$.

Output

Print N lines, each containing the name of the student and their final score out of 100, calculated to three decimal places. The scores should be listed in descending order, from the highest-scoring student to the lowest. In the event of tie-in scores, students should be ordered alphabetically by name.

input
5 NAILA 650 24 25 17 29 30 YAFET 590 23 28 27 29 26 ABDI 650 24 25 20 29 27 JOHN 600 21 23 10 13 16 NAHOM 540 30 22 18 19 20

output
ABDI 86.835 NAILA 86.835 YAFET 86.022 NAHOM 77.011 JOHN 69.824

H. Locker Game

1 second, 256 megabytes

In the hallowed halls of the CSEC club, a unique tradition unfolds each day. The club, home to 'n' students, each with their locker, follows a ritual steeped in numerical mystique. These lockers, lined in a row, bear the unique ID of each student, ranging from 1 to n. As dawn breaks, all locker doors stand closed, awaiting the day's curious ritual.

The ritual revolves around the concept of mentorship embedded in the locker numbers. Each student, upon arrival at the club, holds the responsibility of changing the state of their mentees' lockers, starting with their own. A mentee's locker is identified by any number that is a multiple of the mentor's ID. Thus, a student with ID x mentors the lockers numbered $2x$, $3x$, and so forth. The rule is simple yet intriguing: if a mentee's locker is open, the mentor closes it; if it's closed, the mentor opens it.

The challenge lies in deducing the number of students who attended the club on a given day, based solely on the final state of the lockers. Provided with a list of locker IDs that are open at the day's end, the task is to calculate the total count of present students.

Input

The input begins with two integers, n and k. Here, n represents the total number of students in the CSEC club, and k indicates the number of lockers that are found open at the end of the day. The constraints for these values are $1 \le k \le n \le 10^5$. The second line of the input consists of k unique positive integers, each denoted as ai. These integers represent the IDs of the open lockers. The range for each locker ID is $1 \le a_i \le n$.

Output

You are required to print a single integer, representing the total number of students who were present in the club on that day.

input
10 1 1

output
7

input
100 10 11 3 4 5 19 35 50 98 32 77

output
47