A. 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 $t(1 \leq t \leq 100)$, denoting the number of test cases. Each test case consists of an integer $n(1 \leq n \leq 1000)$, number of words and on the following line contains n words, where each word $|s_i| \leq 1000$ 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
```

B. 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 $t(1 \leq t \leq 1000)$, denoting the number of test cases. The only line of each test case contains a single integer $n(1 \leq n \leq 10^9)$, 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
```

C. 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 $n(1 \leq n \leq 10^5)$, 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
```

D. Master Method

1 second, 256 megabytes

In the world of computational wizardry, a challenge has emerged for those skilled in the art of recursive algorithms. The task revolves around the Master Theorem, a powerful tool used to decipher the time complexity of recursive functions. The theorem applies to recursions expressed in the form $T(n) = aT(n/b) + f(n)$, where $a \geq 1$, $b \geq 2$ are integers, and for simplicity, $f(n) = n^k$ with k being a positive integer.

The Master Theorem presents three distinct cases to determine the time complexity of such a recursion:

Case 1. The time complexity is $n^{log_b^a}$ if $n^{log_b^a} > f(n)$

Case 2. The time complexity is $n^{log_a^b} log(n)$ if $n^{log_a^b} = f(n)$

Case 3. The time complexity is $f(n)$ if $n^{log_a^b} < f(n)$

Given the input values of a, b, and k, they must determine which of these cases applies. Print "Case 1", "Case 2", or "Case 3", corresponding to the appropriate scenario based on the provided formulas.

Input

Input begins with an integer $1 \leq t \leq 10^5$, indicating the number of test cases that follow. Each of the next t lines contains three integer values for $a, b,$ and k. $1 \leq a \leq 10^{18}, 2 \leq b \leq 32, 1 \leq k \leq 12.$

Output

For each test case, output "Case 1", "Case 2", or "Case 3", corresponding to the appropriate scenario based on the provided formulas for the inputs.

input

```
3
8 2 1
4 3 2
4 2 2
```

output

```
Case 1
Case 3
Case 2
```

E. 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 $N(1 \leq N \leq 10^5)$, 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. $0 \leq X \leq 700$, $0 \leq M, P, C, B, E \leq 30, 1 \leq |S| \leq 10$.

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
```

F. 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 $1 \leq k \leq n \leq 10^5$. 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 $1 \leq a_i \leq n$.

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
```

G. Master of Numbers

1 second, 256 megabytes

In the ancient and vibrant land of Ethiopia, a legendary figure known as Ahmed Hibet, renowned for his wisdom and mathematical prowess, has issued a challenge to the brightest minds at the Ethiopian Collegiate Programming Contest (EtCPC). Ahmed Hibet, a sage who has spent years studying the mystical properties of numbers, seeks help in his latest endeavor. He believes that certain numbers, which he calls "Light of Sheba" numbers, are the key to unlocking a historical secret that dates back to the times of the Queen of Sheba. According to his research, these special numbers are defined by a unique characteristic: they have an odd number of divisors.

The challenge laid out by Ahmed Hibet is as follows: Contestants are tasked with finding the sum of all "Light of Sheba" numbers that are less than or equal to a given number N. Specifically, participants must calculate the summation of all integers up to N, where each integer has an odd number of divisors. Ahmed Hibet awaits the solutions, eager to see if anyone can uncover the secrets held by these enigmatic numbers and prove themselves worthy of the title of "Master of Numbers" in the EtCPC.

Input

Input begins with an integer $1 \leq t \leq 10^5$, indicating the number of test cases that follow. Each of the next t lines contains exactly one integer value for N in the range $1 \leq N \leq 10^{12}$.

Output

For each test case, output a line containing an integer representing the sum of all integers up to N, where each integer has an odd number of divisors.

input

```
4
3
10
30
100
```

output

```
1
14
55
385
```

H. The legendary Sage Algebrus

2 seconds, 256 megabytes

In the enigmatic land of Numeria, where mathematics weaves into the essence of existence, an old scroll has emerged, stirring intrigue among the realm's finest scholars. Authored by the legendary sage Algebrus, this scroll presents an extraordinary mathematical challenge:

$$(ax^2 + bx + c)^{dx^2+ex+f} = 1$$

This equation, a masterpiece of numerical artistry, is rumored to hold the secret to an ancient mystery. The sages of Numeria have thus summoned mathematicians far and wide to unravel this puzzle. Their quest is to find all integer solutions to this equation, given the variables a, b, c, d, e, and f. More than just a mathematical riddle, each integer solution is believed to be a clue leading to a fabled treasure, a legacy left by Algebrus himself. The adventurers are tasked with calculating the sum of these integer solutions. Should they find no integer solutions exist, they must declare "No Solution" an admission that the fabled treasure, if real, remains elusive.

Input

Input begins with an integer $1 \le t \le 10^5$, indicating the number of test cases that follow. Each of the next t lines contains six integer values for a, b, c, d, e, and f in the range $-1000 \le a, b, d, e \le 1000, -10^{10} \le c, f \le 10^{10}$, a and d are non-zero integers.

Output

For each test case, output a line containing the summation of all integer solutions to this equation. If there is no integer solution print "No Solution".

input

```
4
1 0 -8 1 0 -4
1 5 5 1 -21 10
3 5 -21 5 2 7
2 15 3 3 10 5
```

output

```
0
-10
2
No Solution
```

Remember to use `long long` if you are using `c++` or `Long` for `java` to avoid overflow.