Problem A. Base Conversion 1

Time limit 1000 ms
Mem limit 262144 kB

After seeing the "ALL YOUR BASE ARE BELONG TO US" meme for the first time, numbers X and Y realised that they have different bases, which complicated their relations.

You're given a number X represented in base b_x and a number Y represented in base b_y. Compare those two numbers.

Input

The first line of the input contains two space-separated integers n and b_x ($1 \le n \le 10$, $2 \le b_x \le 40$), where n is the number of digits in the b_x-based representation of X.

The second line contains n space-separated integers $x_1, x_2, \ldots, x_n$ ($0 \le x_i < b_x$) — the digits of X. They are given in the order from the most significant digit to the least significant one.

The following two lines describe Y in the same way: the third line contains two space-separated integers m and b_y ($1 \le m \le 10$, $2 \le b_y \le 40$, $b_x \ne b_y$), where m is the number of digits in the b_y-based representation of Y, and the fourth line contains m space-separated integers $y_1, y_2, \ldots, y_m$ ($0 \le y_i < b_y$) — the digits of Y.

There will be no leading zeroes. Both X and Y will be positive. All digits of both numbers are given in the standard decimal numeral system.

Output

Output a single character (quotes for clarity):

- '<' if $X < Y$
- '>' if $X > Y$
- '=' if $X = Y$

Examples

Input	Output
6 2 1 0 1 1 1 1 2 10 4 7	=

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

Input	Output
7 16 15 15 4 0 0 7 10 7 9 4 8 0 3 1 5 0	>

Note

In the first sample, $X = 101111_2 = 47_{10} = Y$.

In the second sample, $X = 102_3 = 21_5$ and $Y = 24_5 = 112_3$, thus $X < Y$.

In the third sample, $X = \mathrm{FF4007A}_{16}$ and $Y = 4803150_9$. We may notice that X starts with much larger digits and b_x is much larger than b_y, so X is clearly larger than Y.

Problem B. Base Conversion 2

Time limit	1000 ms
Mem limit	65536 kB
Input file	stdin
Output file	stdout

Little Petya likes numbers a lot. He found that number 123 in base 16 consists of two digits: the first is 7 and the second is 11. So the sum of digits of 123 in base 16 is equal to 18.

Now he wonders what is an average value of sum of digits of the number A written in all bases from 2 to A - 1.

Note that all computations should be done in base 10. You should find the result as an irreducible fraction, written in base 10.

Input

Input contains one integer number A ($3 \leq A \leq 1000$).

Output

Output should contain required average value in format «`X/Y`», where `X` is the numerator and `Y` is the denominator.

Examples

Input	Output
5	7/3

Input	Output
3	2/1

Note

In the first sample number 5 written in all bases from 2 to 4 looks so: 101, 12, 11. Sums of digits are 2, 3 and 2, respectively.

Problem C. Base Conversion 3

Time limit 1000 ms
Mem limit 262144 kB

Alexander is learning how to convert numbers from the decimal system to any other, however, he doesn't know English letters, so he writes any number only as a decimal number, it means that instead of the letter A he will write the number 10. Thus, by converting the number 475 from decimal to hexadecimal system, he gets 11311 $(475 = 1 \cdot 16^2 + 13 \cdot 16^1 + 11 \cdot 16^0)$. Alexander lived calmly until he tried to convert the number back to the decimal number system.

Alexander remembers that he worked with little numbers so he asks to find the minimum decimal number so that by converting it to the system with the base n he will get the number k.

Input

The first line contains the integer n $(2 \le n \le 10^9)$. The second line contains the integer k $(0 \le k < 10^{60})$, it is guaranteed that the number k contains no more than 60 symbols. All digits in the second line are strictly less than n.

Alexander guarantees that the answer exists and does not exceed 10^{18}.

The number k doesn't contain leading zeros.

Output

Print the number x $(0 \le x \le 10^{18})$ — the answer to the problem.

Examples

Input	Output
13 12	12

Input	Output
16 11311	475

Input	Output
20 999	3789

Input	Output
17 2016	594

Note

In the first example 12 could be obtained by converting two numbers to the system with base 13: $12 = 12 \cdot 13^0$ or $15 = 1 \cdot 13^1 + 2 \cdot 13^0$.

Problem D. Binary Number 1

Time limit 1000 ms
Mem limit 262144 kB
Input file stdin
Output file stdout

Little walrus Fangy loves math very much. That's why when he is bored he plays with a number performing some operations.

Fangy takes some positive integer x and wants to get a number one from it. While x is not equal to 1, Fangy repeats the following action: if x is odd, then he adds 1 to it, otherwise he divides x by 2. Fangy knows that for any positive integer number the process ends in finite time.

How many actions should Fangy perform to get a number one from number x?

Input

The first line contains a positive integer x in a **binary system**. It is guaranteed that the first digit of x is different from a zero and the number of its digits does not exceed 10^6.

Output

Print the required number of actions.

Examples

Input	Output
1	0

Input	Output
1001001	12

Input	Output
101110	8

Note

Let's consider the third sample. Number 101110 is even, which means that we should divide it by 2. After the dividing Fangy gets an odd number 10111 and adds one to it. Number 11000 can be divided by 2 three times in a row and get number 11. All that's left is to increase the number by one (we get 100), and then divide it by 2 two times in a row. As a result, we get 1.

Problem E. Binary Number 2

Time limit 1000 ms
Mem limit 65536 kB

This is the tale of Zephyr, the greatest time traveler the world will never know. Even those who are aware of Zephyr's existence know very little about her. For example, no one has any clue as to which time-period she is originally from.

But we do know the story of the first time she set out to chart her own path in the time stream. Zephyr had just finished building her time machine which she named - "Dokhina Batash". She was making the final adjustments for her first trip when she noticed that a vital program was not working properly. The program was supposed to take a number **N** and find what Zephyr called its Onoroy value.

The Onoroy value of an integer **N** is the number of ones in its binary representation. For example, the number 13 (1101_2) has an Onoroy value of 3. Needless to say, this was an easy problem for the great mind of Zephyr. She solved it quickly and was on her way.

You are now given a similar task. Find the first number after **N** which has the same Onoroy value as **N**.

Input

Input starts with an integer **T (≤ 65)**, denoting the number of test cases.

Each case begins with an integer **N** $(1 \le N \le 10^9)$.

Output

For each case of input you have to print the case number and the desired result.

Sample

Input	Output
5 23 14232 391 7 8	Case 1: 27 Case 2: 14241 Case 3: 395 Case 4: 11 Case 5: 16

Problem F. Binary Number 3

Time limit 1000 ms
Mem limit 262144 kB

You are given a number in binary representation consisting of exactly n bits, possibly, with leading zeroes. For example, for $n = 5$ the number 6 will be given as 00110, and for $n = 4$ the number 9 will be given as 1001.

Let's fix some integer i such that $1 \le i \le n$. In one operation you can swap any two adjacent bits in the binary representation. Your goal is to find the smallest number of operations you are required to perform to make the number divisible by 2^i, or say that it is impossible.

Please note that for each $1 \le i \le n$ you are solving the problem independently.

Input

Each test contains multiple test cases. The first line contains the number of test cases t ($1 \le t \le 10^4$). The description of the test cases follows.

The first line of each test case contains one integer n ($1 \le n \le 10^5$) — the length of the binary representation of the number.

The second line of each test case contains a string of length n, consisting of 0 and 1, — the binary representation of the number.

It is guaranteed that the sum of n over all test cases does not exceed $2 \cdot 10^5$.

Output

For each test case, for each $1 \le i \le n$ output the smallest number of operations required to make the number divisible by 2^i, or output -1 if it is impossible.

Examples

Input	Output
6 1 1 2 01 3 010 5 10101 7 0000111 12 001011000110	-1 1 -1 0 1 -1 1 3 -1 -1 -1 3 6 9 12 -1 -1 -1 0 2 4 6 10 15 20 -1 -1 -1 -1 -1

Note

In the first test case, we cannot swap any elements, and the number 1 is not divisible by 2.

In the second test case, the initial number is 1. It is not divisible by 2, but if we perform the operation, then we obtain the number with binary representation 10, which is equal to 2 in decimal representation, thus, it is divisible by 2. But this number is not divisible by 4 and we cannot obtain any other number using the operations.

In the third test case, the initial number is 2. For $i = 1$ we do not have to perform any operations since the initial number is divisible by 2. For $i = 2$ we can perform one operation swapping the first two bits (we would obtain 100 in binary representation, which corresponds to number 4). There is no answer for $i = 3$.

Problem G. Binary Number 4

Time limit 1000 ms
Mem limit 131072 kB

Carl is right now the happiest child in the world: he has just learned this morning what the binary system is. He learned, for instance, that the binary representation of a positive integer k is a string $a_n a_{n-1} \ldots a_1 a_0$ where each a_i is a binary digit 0 or 1, starting with $a_n = 1$, and such that

$$k = \sum_{i=0}^{n} a_i \times 2^i$$

It is really nice to see him turning decimal numbers into binary numbers, and then adding and even multiplying them.

Caesar is Carl's older brother, and he just can't stand to see his little brother so happy. So he has prepared a challenge: "Look Carl, I have an easy question for you: I will give you two integers a and b, and you have to tell me how many 1's there are in the binary representation of all the integers from a to b inclusive. Get ready". Carl agreed to the challenge. After a few minutes, he came back with a list of the binary representation of all the integers from 1 to 100. "Caesar, I'm ready". Caesar smiled and said: "Well, let me see, I choose $a = 10^{15}$ and $b = 10^{16}$. Your list will not be useful".

Carl hates loosing to his brother so he needs a better solution fast. Can you help him?

Input

A single line that contains two integers a and b $(1 \leq a \leq b \leq 10^{16})$.

Output

Output a line with an integer representing the total number of digits 1 in the binary representation of all the integers from a to b inclusive.

Examples

Input	Output
2 12	21

Problem H. All Possible Subset 1

Time limit 1000 ms
Mem limit 524288 kB

There are n apples with known weights. Your task is to divide the apples into two groups so that the difference between the weights of the groups is minimal.

Input

The first input line has an integer n: the number of apples.

The next line has n integers $p_1, p_2, \ldots, p_n$: the weight of each apple.

Output

Print one integer: the minimum difference between the weights of the groups.

Constraints

- $1 \le n \le 20$
- $1 \le p_i \le 10^9$

Example

Input	Output
5 3 2 7 4 1	1

Explanation: Group 1 has weights 2, 3 and 4 (total weight 9), and group 2 has weights 1 and 7 (total weight 8).

Problem I. All Possible Subset 2

Time limit 2000 ms
Mem limit 262144 kB

Problem Statement

You are given a string S consisting of digits between `1` and `9`, inclusive. You can insert the letter `+` into some of the positions (possibly none) between two letters in this string. Here, `+` must not occur consecutively after insertion.

All strings that can be obtained in this way can be evaluated as formulas.

Evaluate all possible formulas, and print the sum of the results.

Constraints

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

Input

The input is given from Standard Input in the following format:

S

Output

Print the sum of the evaluated value over all possible formulas.

Sample 1

Input	Output
125	176

There are 4 formulas that can be obtained: `125`, `1+25`, `12+5` and `1+2+5`. When each formula is evaluated,

- 125
- $1 + 25 = 26$
- $12 + 5 = 17$
- $1 + 2 + 5 = 8$

Thus, the sum is $125 + 26 + 17 + 8 = 176$.

Sample 2

Input	Output
9999999999	12656242944

Problem J. All Possible Subset 3

Time limit	2000 ms
Mem limit	262144 kB
Input file	stdin
Output file	stdout

Petya loves lucky numbers. Everybody knows that positive integers are lucky if their decimal representation doesn't contain digits other than 4 and 7. For example, numbers 47, 744, 4 are lucky and 5, 17, 467 are not.

Lucky number is super lucky if it's decimal representation contains equal amount of digits 4 and 7. For example, numbers 47, 7744, 474477 are super lucky and 4, 744, 467 are not.

One day Petya came across a positive integer n. Help him to find the least super lucky number which is not less than n.

Input

The only line contains a positive integer n ($1 \le n \le 10^9$). This number doesn't have leading zeroes.

Output

Output the least super lucky number that is more than or equal to n.

Please, do not use the %lld specificator to read or write 64-bit integers in C++. It is preferred to use the cin, cout streams or the %I64d specificator.

Examples

Input	Output
4500	4747

Input	Output
47	47

Problem K. All Possible Subset 4

Time limit 2000 ms
Mem limit 262144 kB

You have n problems. You have estimated the difficulty of the i-th one as integer c_i. Now you want to prepare a problemset for a contest, using some of the problems you've made.

A problemset for the contest must consist of at least two problems. You think that the total difficulty of the problems of the contest must be at least l and at most r. Also, you think that the difference between difficulties of the easiest and the hardest of the chosen problems must be at least x.

Find the number of ways to choose a problemset for the contest.

Input

The first line contains four integers n, l, r, x ($1 \le n \le 15$, $1 \le l \le r \le 10^9$, $1 \le x \le 10^6$) — the number of problems you have, the minimum and maximum value of total difficulty of the problemset and the minimum difference in difficulty between the hardest problem in the pack and the easiest one, respectively.

The second line contains n integers $c_1, c_2, \dots, c_n$ ($1 \le c_i \le 10^6$) — the difficulty of each problem.

Output

Print the number of ways to choose a suitable problemset for the contest.

Examples

Input	Output
3 5 6 1 1 2 3	2

Input	Output
4 40 50 10 10 20 30 25	2

Input	Output
5 25 35 10 10 10 20 10 20	6

Note

In the first example two sets are suitable, one consisting of the second and third problem, another one consisting of all three problems.

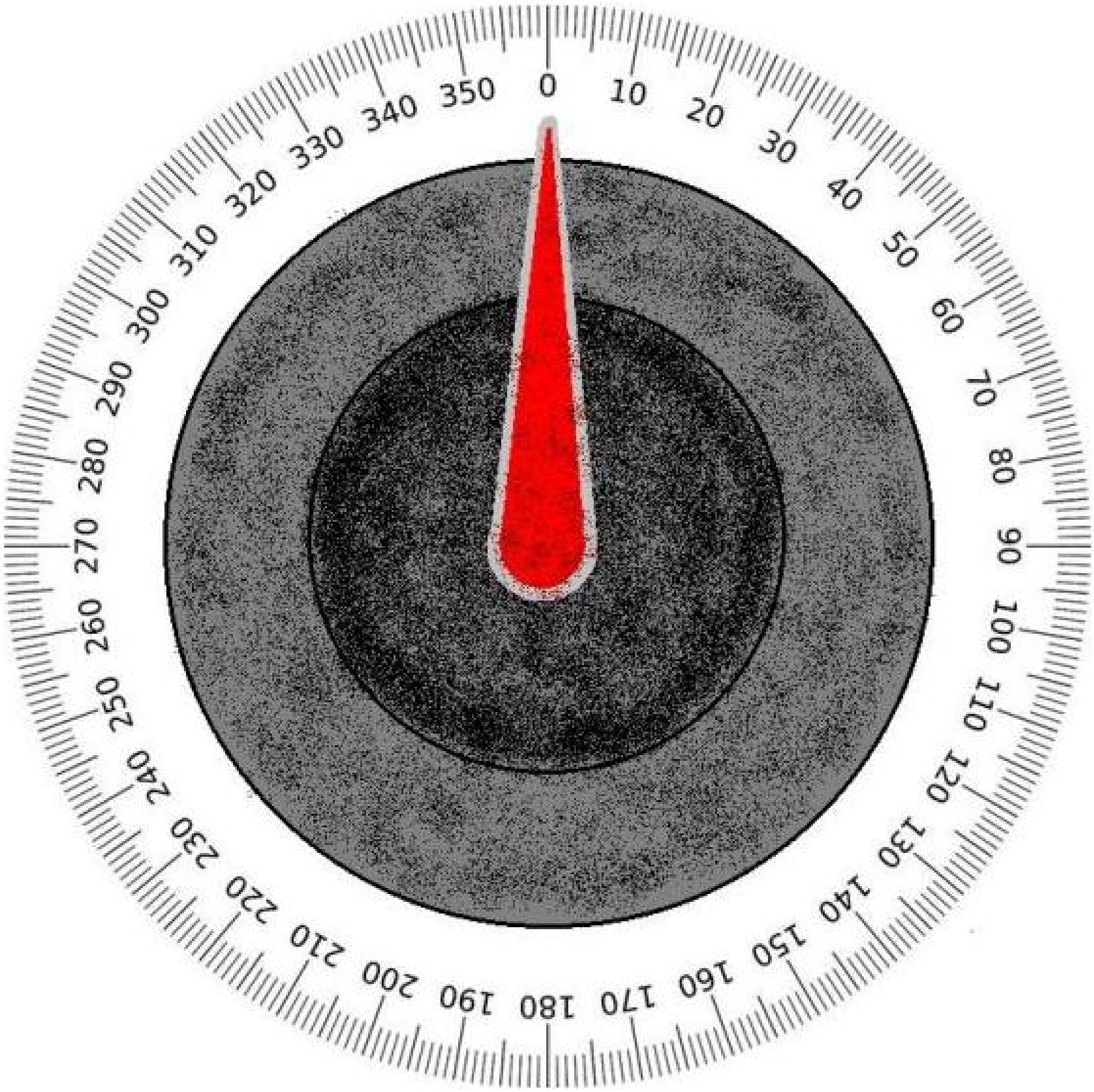
In the second example, two sets of problems are suitable — the set of problems with difficulties 10 and 30 as well as the set of problems with difficulties 20 and 30.

In the third example any set consisting of one problem of difficulty 10 and one problem of difficulty 20 is suitable.

Problem L. All Possible Subset 5

Time limit 1000 ms
Mem limit 262144 kB

Petr has just bought a new car. He's just arrived at the most known Petersburg's petrol station to refuel it when he suddenly discovered that the petrol tank is secured with a combination lock! The lock has a scale of 360 degrees and a pointer which initially points at zero:



Petr called his car dealer, who instructed him to rotate the lock's wheel exactly n times. The i-th rotation should be a_i degrees, either clockwise or counterclockwise, and after all n rotations the pointer should again point at zero.

This confused Petr a little bit as he isn't sure which rotations should be done clockwise and which should be done counterclockwise. As there are many possible ways of rotating the lock, help him and find out whether there exists at least one, such that after all n rotations the pointer will point at zero again.

Input

The first line contains one integer n ($1 \le n \le 15$) — the number of rotations.

Each of the following n lines contains one integer a_i ($1 \le a_i \le 180$) — the angle of the i-th rotation in degrees.

Output

If it is possible to do all the rotations so that the pointer will point at zero after all of them are performed, print a single word "`YES`". Otherwise, print "`NO`". Petr will probably buy a new car in this case.

You can print each letter in any case (upper or lower).

Examples

Input	Output
3 10 20 30	YES

Input	Output
3 10 10 10	NO

Input	Output
3 120 120 120	YES

Note

In the first example, we can achieve our goal by applying the first and the second rotation clockwise, and performing the third rotation counterclockwise.

In the second example, it's impossible to perform the rotations in order to make the pointer point at zero in the end.

In the third example, Petr can do all three rotations clockwise. In this case, the whole wheel will be rotated by 360 degrees clockwise and the pointer will point at zero again.

Problem M. Inclusion Exclusion Principle 1

Time limit 500 ms
Mem limit 65536 kB

IT City company developing computer games decided to upgrade its way to reward its employees. Now it looks the following way. After a new game release users start buying it actively, and the company tracks the number of sales with precision to each transaction. Every time when the next number of sales is not divisible by any number from 2 to 10 every developer of this game gets a small bonus.

A game designer Petya knows that the company is just about to release a new game that was partly developed by him. On the basis of his experience he predicts that n people will buy the game during the first month. Now Petya wants to determine how many times he will get the bonus. Help him to know it.

Input

The only line of the input contains one integer n ($1 \leq n \leq 10^{18}$) — the prediction on the number of people who will buy the game.

Output

Output one integer showing how many numbers from 1 to n are not divisible by any number from 2 to 10.

Examples

Input	Output
12	2

The Sports Association of Bangladesh is in great problem with their latest lottery 'Jodi laiga Jai'. There are so many participants this time that they cannot manage all the numbers. In an urgent meeting they have decided that they will ignore some numbers. But how they will choose those unlucky numbers!!! Mr. NondoDulal who is very interested about historic problems proposed a scheme to get free from this problem.

You may be interested to know how he has got this scheme. Recently he has read the Joseph's problem.

There are N tickets which are numbered from 1 to N. Mr. Nondo will choose M random numbers and then he will select those numbers which is divisible by at least one of those M numbers. The numbers which are not divisible by any of those M numbers will be considered for the lottery.

As you know each number is divisible by 1. So Mr. Nondo will never select 1 as one of those M numbers. Now given N, M and M random numbers, you have to find out the number of tickets which will be considered for the lottery.

Input

Each input set starts with two Integers N ($10 \le N < 2^{31}$) and M ($1 \le M \le 15$). The next line will contain M positive integers each of which is not greater than N.

Input is terminated by EOF.

Output

Just print in a line out of N tickets how many will be considered for the lottery.

Sample Input

```
10 2
2 3
20 2
2 4
```

Sample Output

```
3
10
```

Problem O. Inclusion Exclusion Principle 3

Time limit 1000 ms
Mem limit 524288 kB

You are given k distinct prime numbers $a_1, a_2, \ldots, a_k$ and an integer n.

Your task is to calculate how many of the first n positive integers are divisible by at least one of the given prime numbers.

Input

The first input line has two integers n and k.

The second line has k prime numbers $a_1, a_2, \ldots, a_k$.

Output

Print one integer: the number integers within the interval $1, 2, \ldots, n$ that are divisible by at least one of the prime numbers.

Constraints

- $1 \leq n \leq 10^{18}$
- $1 \leq k \leq 20$
- $2 \leq a_i \leq n$

Example

Input	Output
20 2 2 5	12

Explanation: the 12 numbers are $2, 4, 5, 6, 8, 10, 12, 14, 15, 16, 18, 20$.