Codeforces Round #320 (Div. 2) [Bayan Thanks-Round]

A. Raising Bacteria

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

You are a lover of bacteria. You want to raise some bacteria in a box.

Initially, the box is empty. Each morning, you can put any number of bacteria into the box. And each night, every bacterium in the box will split into two bacteria. You hope to see exactly x bacteria in the box at some moment.

What is the minimum number of bacteria you need to put into the box across those days?

Input
The only line containing one integer x ($1 \le x \le 10^9$).

Output
The only line containing one integer: the answer.

Examples

input
5
output
2

input
8
output
1

Note
For the first sample, we can add one bacterium in the box in the first day morning and at the third morning there will be 4 bacteria in the box. Now we put one more resulting 5 in the box. We added 2 bacteria in the process so the answer is 2.

For the second sample, we can put one in the first morning and in the 4-th morning there will be 8 in the box. So the answer is 1.

B. Finding Team Member

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

There is a programing contest named SnakeUp, $2n$ people want to compete for it. In order to attend this contest, people need to form teams of exactly two people. You are given the strength of each possible combination of two people. All the values of the strengths are **distinct**.

Every contestant hopes that he can find a teammate so that their team's strength is as high as possible. That is, a contestant will form a team with highest strength possible by choosing a teammate from ones who are willing to be a teammate with him/her. More formally, two people A and B may form a team if each of them is the best possible teammate (among the contestants that remain unpaired) for the other one.

Can you determine who will be each person's teammate?

Input

There are $2n$ lines in the input.

The first line contains an integer n ($1 \le n \le 400$) — the number of teams to be formed.

The i-th line ($i > 1$) contains $i - 1$ numbers $a_{i1}, a_{i2}, \ldots, a_{i(i-1)}$. Here a_{ij} ($1 \le a_{ij} \le 10^6$, all a_{ij} are distinct) denotes the strength of a team consisting of person i and person j (people are numbered starting from 1.)

Output

Output a line containing $2n$ numbers. The i-th number should represent the number of teammate of i-th person.

Examples

input
2 6 1 2 3 4 5
output
2 1 4 3

input
3 487060 3831 161856 845957 794650 976977 83847 50566 691206 498447 698377 156232 59015 382455 626960
output
6 5 4 3 2 1

Note

In the first sample, contestant 1 and 2 will be teammates and so do contestant 3 and 4, so the teammate of contestant $1, 2, 3, 4$ will be $2, 1, 4, 3$ respectively.

C. A Problem about Polyline

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

There is a polyline going through points $(0, 0) - (x, x) - (2x, 0) - (3x, x) - (4x, 0) - \ldots - (2kx, 0) - (2kx + x, x) - \ldots$.

We know that the polyline passes through the point (a, b). Find minimum positive value x such that it is true or determine that there is no such x.

Input

Only one line containing two positive integers a and b ($1 \le a, b \le 10^9$).

Output

Output the only line containing the answer. Your answer will be considered correct if its relative or absolute error doesn't exceed 10^{-9}. If there is no such x then output -1 as the answer.

Examples

input
3 1
output
1.000000000000

input
1 3
output
-1

input
4 1
output
1.250000000000

Note

You can see following graphs for sample 1 and sample 3.

D. "Or" Game

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

You are given n numbers $a_1, a_2, \ldots, a_n$. You can perform at most k operations. For each operation you can multiply one of the numbers by x. We want to make $a_1 \mid a_2 \mid \ldots \mid a_n$ as large as possible, where $\mid$ denotes the bitwise OR.

Find the maximum possible value of $a_1 \mid a_2 \mid \ldots \mid a_n$ after performing at most k operations optimally.

Input

The first line contains three integers n, k and x ($1 \le n \le 200\,000$, $1 \le k \le 10$, $2 \le x \le 8$).

The second line contains n integers $a_1, a_2, \ldots, a_n$ ($0 \le a_i \le 10^9$).

Output

Output the maximum value of a bitwise OR of sequence elements after performing operations.

Examples

input
3 1 2 1 1 1
output
3

input
4 2 3 1 2 4 8
output
79

Note

For the first sample, any possible choice of doing one operation will result the same three numbers 1, 1, 2 so the result is $1 \mid 1 \mid 2 = 3$.

For the second sample if we multiply 8 by 3 two times we'll get 72. In this case the numbers will become 1, 2, 4, 72 so the OR value will be 79 and is the largest possible result.

E. Weakness and Poorness

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

You are given a sequence of n integers $a_1, a_2, \ldots, a_n$.

Determine a real number x such that the *weakness* of the sequence $a_1 - x, a_2 - x, \ldots, a_n - x$ is as small as possible.

The *weakness* of a sequence is defined as the maximum value of the *poorness* over all segments (contiguous subsequences) of a sequence.

The *poorness* of a segment is defined as the absolute value of sum of the elements of segment.

Input

The first line contains one integer n ($1 \le n \le 200\,000$), the length of a sequence.

The second line contains n integers $a_1, a_2, \ldots, a_n$ ($|a_i| \le 10\,000$).

Output

Output a real number denoting the minimum possible *weakness* of $a_1 - x, a_2 - x, \ldots, a_n - x$. Your answer will be considered correct if its relative or absolute error doesn't exceed 10^{-6}.

Examples

input
3 1 2 3
output
1.000000000000000

input
4 1 2 3 4
output
2.000000000000000

input
10 1 10 2 9 3 8 4 7 5 6
output
4.500000000000000

Note

For the first case, the optimal value of x is 2 so the sequence becomes $-1, 0, 1$ and the max poorness occurs at the segment "-1" or segment "1". The poorness value (answer) equals to 1 in this case.

For the second sample the optimal value of x is 2.5 so the sequence becomes $-1.5, -0.5, 0.5, 1.5$ and the max poorness occurs on segment "-1.5 -0.5" or "0.5 1.5". The poorness value (answer) equals to 2 in this case.

F. LCS Again

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

You are given a string S of length n with each character being one of the first m lowercase English letters.

Calculate how many different strings T of length n composed from the first m lowercase English letters exist such that the length of LCS (longest common subsequence) between S and T is $n - 1$.

Recall that LCS of two strings S and T is the longest string C such that C both in S and T as a subsequence.

Input

The first line contains two numbers n and m denoting the length of string S and number of first English lowercase characters forming the character set for strings ($1 \le n \le 100\,000$, $2 \le m \le 26$).

The second line contains string S.

Output

Print the only line containing the answer.

Examples

input
3 3 aaa
output
6

input
3 3 aab
output
11

input
1 2 a
output
1

input
10 9 abacadefgh
output
789

Note

For the first sample, the 6 possible strings T are: aab, aac, aba, aca, baa, caa.

For the second sample, the 11 possible strings T are: aaa, aac, aba, abb, abc, aca, acb, baa, bab, caa, cab.

For the third sample, the only possible string T is b.