Time Limit: 1000ms
Memory Limit: 256MB

A - The Tortoise and the Hare

Submissions Submit

After being frustrated by losing against the tortoise in every race, the hare challenged the tortoise to an eating contest, the tortoise decided to cheat and gather his friends to help him in the contest, he plans to eat as much as he can, then swap with his other tortoise friend, his friend will also eat as much as he can, then he will swap with another friend and so on, until all of his friends took a turn eating.

If the hare can eat h kg of food and a single tortoise can eat t kg of food, how many friends does the tortoise need to get in order for them to eat more than the hare?

Input

A single line containing $h, t (1 \leq h, t \leq 100)$, the amount of food the hare can eat and the amount of food a single tortoise can eat.

Output

Print a single line containing the amount of friends the tortoise needs to get in order to eat more than the hare

Notes

In the first example, a single tortoise can eat 2 kg of food, if he brings 2 other tortoise with him, overall they will be able to eat 6 kg of food, which is more than the hare can eat.

Samples

Input	Output
4 2	2
4 3	1

Time Limit: 2000ms
Memory Limit: 256MB

B - Find a Friend

Submissions Submit

As a way to increase collaboration between employees of different ranks, Manly Software Solutions decided to create friendships inside the company.

The company has n employees, each having an id number from 1 to n, every employee in the company has a manager, except of the CEO who has an id of 1, so the structure of the employees looks like a tree, also every employee has a value p_i representing his personality.

The company wants every employee to make a friendship with a junior of his that is exactly 4 ranks below him(meaning he is the manager of the manager of the manager of the manager of that employee).

Since employees are busy working, they want you to find for every employee the best candidate for creating a friendship with, the best candidate is the employee with the closest personality value(that is the difference between their personality values is minimum), if there are multiple such employees then the one with the lowest id value among them is considered the best candidate.

Given the tree structure of the employees and their personality values, print for each employee the id of his best candidate for friendship making, or state that no such candidate exists.

Input

The first line contains a single integer n($1 \leq n \leq 2 \cdot 10^5$), the number of employees in the company.

The second line contains $n-1$ integers $m_2, m_3, \ldots, m_n$($1 \leq m_i < i$), the manager of the ith employee.

The third line contains n integers $p_1, p_2, \ldots, p_n$($1 \leq p_i \leq 10^9$), the personality of the ith employee.

Output

Print a single line containing n integers, the ith of them being the id of the best candidate for the ith employee, or 0 if there is no such candidate.

Notes

In the first example, the CEO($id = 1$) has two juniors that are exactly four ranks lower than him, employee 5 and 9, the difference of personality with employee 5 is $|3 - 6| = 3$, and with employee 9 is $|3 - 10| = 7$, so employee 5 is his best candidate.

The second employee has two juniors, employee 6 and 10, 10 is the best candidate.

None of the other employees have a junior that is exactly 4 ranks lower than them.

Samples

Input	Output
10 1 2 3 4 5 2 7 8 9 3 20 5 3 6 13 9 3 10 15	5 10 0 0 0 0 0 0 0 0

Time Limit: 2000ms
Memory Limit: 256MB

C - Slavery

Submissions Submit

Have you ever heard the quote "Everyone is a slave to something" before?

It goes like: In the grand scheme of things, most people tend to have desires that rule their life, shape their goals, and control their actions.. Whether it's the desire for money, fame, power, or even virtue; we end up doing what flows with that desire, what we originally tied ourselves to.

You might agree, you might not.. But nevertheless, dear participant, the one thing that we can know for sure at this point of time, is that $\mathbf{you}$ sir (or madam), are the slave in this problem.

In this problem we have designed here, we will be providing you with an array a of non-negative integers of length n, and will be asking you to perform a series of operations on it, fifteen precise steps, if you may; in order to test your resolve, evaluate your patience, and most importantly.. Waste your time.

Without further ado, please, perform these fifteen steps on the provided array:

1. Append the integer 0 to the end of the array.
2. Replace every integer in the array with the maximum between it and the next number in the array (if present). In other words, set $a_i = max(a_i, a_{i+1})$
3. If the size of the array is even:
   - Split the array in half
   - Reverse each of the two sides individually
   - Merge them back into a single array in the same order
4. If the size of the array is odd:
   - Remove the longest suffix of the array that has a sum of less than or equal to n (the original size of the array).
   - Jump back to step **3**
5. Append these three integers to the end of the array: $[3, 2, 1]$
6. Go over the array from left to right, if a_i is greater than the last element in the array, swap them.
7. Change the value of the first element in the array to be equal to 1031.
8. Replace every integer in the array with its largest prime factor (0 and 1 stay the same).
9. **Sort the array in a descending order.** ~~Replace the entire array with only the unique values present in it, sorted in a descending order.~~
10. Replace every element in the array with its value modulo 16384.
11. Go over the array from left to right, if the current element is equal to 6, and the next two elements are equal to 7 and 9 respectively, swap the current element with the one before it in the array (if present).
12. Multiply every integer in the array by the number of its bits.
13. If the size of the array is odd, jump back to step 5.
14. If either the first element or the last element in the array is equal to zero, replace the whole array with: $[1, 2, 3, 4, 5]$
15. Output the sum of every element multiplied by its position in the array (1-indexed). In other words: $\sum_{i=1}^{m} a_i * i$

Input

The first line of input will contain an integer n $(1 \leq n \leq 10^5)$ --- the length of the original array.

The second line will contain n space-separated integers $(0 \leq a_i \leq n)$ --- the values of the array a.

Output

Output one integer, the value described in step 15, after carefully executing the given procedure.

Samples

Input	Output

Input	Output
1 1	55
8 4 7 8 2 7 2 3 3	12834

Time Limit: 1000ms
Memory Limit: 256MB

D - Scientific Study

Submissions Submit

A scientist wants to do a study on students to see how they would choose candy from a set. He has n students and candies. The candies are sorted by deliciousness increasingly. The candies from 1 till k have deliciousness of 1 the candies from $k+1$ till $2*k$ have deliciousness of 2, the candies from $2*k+1$ till $3*k$ have deliciousness of 3,..... the candies from $x*k$ till n have deliciousness of x, the last group can have a size less than k.

Students will pick the candy with the highest deliciousness, if multiple candies have the same value then the student can pick any. Starting with the student numbered 1, take turns choosing candy, the ith student can choose candy from the candies that lie in the range $[max(1,i-e),min(n,i+e)]$ each candy can be picked by at most one student.

The scientist needs to know if there exists a way such that each student takes exactly one candy.

Input

The input is made up of 3 integers on a single line $n,e,k(1 \leq n,k \leq 10^9,0 \leq e \leq 10^9)$, the number of student and candies, the value that defines the range the student can see, and the size of a single group of candies, respectively.

Output

Output "Yes"(without quotes) if there exists a way, otherwise output "No"(without quotes).

Notes

1^{st} test case:

The first student has to pick the last candy.

The second student can pick either the first or the second.

The third can pick whatever is left since he can see the whole array.

2^{nd} test case:

The first student has to pick the second candy.

The second student has to pick the first candy.

Samples

Input	Output
3 2 2	Yes
2 1 2	Yes

Time Limit: 3000ms
Memory Limit: 256MB

E - Scientific Study Again!!

Submissions Submit

Now the scientist wants you to count the number of ways each student can pick a candy, but this time he ordered the candies by deliciousness decreasingly.

The candies are sorted by deliciousness decreasingly. The candies from 1 till k have deliciousness of 10^{18}, the candies from $k+1$ till $2*k$ have deliciousness of $10^{18}-1$, the candies from $2*k+1$ till $3*k$ have deliciousness of $10^{18}-2$,..... the candies from $x*k+1$ till n have deliciousness of $10^{18}-x$.

Each student i chooses one candy from the candies that lie in the range $[max(1,i-e),min(n,i+e)]$,borders included.

Going from left to right, each student will pick the candy with the highest deliciousness if there are multiple ones with the same value the student picks any candy.

When a student picks a candy it disappears and its place becomes empty.

The scientist needs to know the number of ways each student can pick a candy such that each student takes exactly one candy.

You need to answer q queries.

All queries have the same value of k while only the value of n changes per query.

print the answer for each query $\mod 10^9+7$.

Input

The first line of the input is made up of 3 integers $k,e,q(1 \leq k \leq 10^{18}, 1 \leq e \leq 4, 1 \leq q \leq 2000)$, the size of a single group of candies, the value that defines the range the student can see, and the number of queries, respectively.

The next q lines each contain a single integer $n(1 \leq n \leq 10^{18})$ the number of students and candies in the query.

Output

Output q lines.

The integer in i_{th} line represents the answer for the i_{th} query.

Samples

Input	Output
2 1 1 2	2
4 2 5 100 12 13 54 1	517593878 2744 2744 535396285 1
3 2 4 10 5 3 7	216 12 6 36

Input	Output
1 2 1 2	1

Time Limit: 1000ms
Memory Limit: 256MB

F - Ruler

Submissions Submit

Iyas likes measuring things, for that he wants to create his own special ruler, he has a wooden stick of length L and wants to draw dashes on the stick so that it works as a ruler, the ruler will also have a precision P.

The ruler will consist of multiple lines, there are $L+1$ main lines, the ith of them will have P dashes and number $i-1$ next to the dashes, between those will be other lines consisting of $P-1$ dashes and so on until lines with a single dash are reached (see examples for a clearer idea).

Given L and P, print the drawing that should be on the ruler.

Input

A single line consisting of $L, P (1 \leq L, P \leq 10)$, the length of the ruler and its precision.

Output

Output the drawing that should be on the ruler.

Samples

Input	Output
3 4	----0 - -- - --- - -- - ----1 - -- - --- - -- - ----2 - -- - --- - -- - ----3

Input	Output
1 5	-----0 - -- - --- - -- - ---- - -- - --- - -- - -----1

Time Limit: 1000ms
Memory Limit: 256MB

G - Legendary

Submissions Submit

Essa has presented you with one of his 3 legendary problems, you'll be given an array A of n non-negative integers, an integer d, and he asks you to find m, defined as:

The size of the largest subset of elements S you can pick from A such that the following three conditions are satisfied:

- The bitwise AND of all of the elements in the chosen set S is equal to d $(S_1 \ \& \ S_2 \ \& \ S_3 \ \& \ \ldots \ \& \ S_m = d)$
- The bitwise OR of all of the elements in the chosen set S is equal to d $(S_1 \mid S_2 \mid S_3 \mid \ \ldots \ \mid S_m = d)$
- The bitwise XOR of all of the elements in the chosen set S is equal to d $(S_1 \ \oplus \ S_2 \ \oplus \ S_3 \ \oplus \ \ \ldots \ \ \oplus \ S_m = d)$

Input

The first line will contain two integers n and d $(1 \leq n \leq 2 * 10^5, 0 \leq d \leq 10^9)$.

The second line will contain n space-separated integers, the contents of the array $(0 \leq A_i \leq 10^9)$.

Output

If there's no such subset of elements, print the integer 0. Otherwise, print the maximum possible size of a subset S that satisfies the given conditions.

Samples

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

Time Limit: 2000ms
Memory Limit: 256MB

H - Bracket Sequence

Submissions Submit

Given an integer n, find the number of balanced bracket sequences of length n, that have the sequence "((()))" as a subarray.

A sequence of brackets is called balanced if one can turn it into a valid math expression by adding characters '+' and '1'. For example, sequences "(())()", "()", and "((()(())))" are balanced, while ")(", "((", and "(()))(" are not.

Print the answer $(mod\ 10^9 + 7)$.

Input

The input is made up of one line containing an integer n $(1 \leq n \leq 1000)$.

Output

Print a single integer the answer to the problem.

Samples

Input	Output
6	1
8	4

Time Limit: 1000ms
Memory Limit: 256MB

I - Binary Circle

Submissions Submit

There are n numbers in a circle **where n is odd**, each number is either 0 or 1, in one operation you can select an index i, and the value at that index and at the two adjacent indices facing opposite of index i on that circle are flipped(if the value was 1 it becomes 0, and if it was 0 it becomes 1). More formally selecting index i will flip the value at indices $i, (i + \lfloor \frac{n}{2} \rfloor - 1 \mod n) + 1, (i - \lfloor \frac{n}{2} \rfloor - 1 \mod n) + 1$.

For example if the numbers are 0010110 and you do an operation on the 6th index, values at indices $6, 2, 3$ will be flipped, and the numbers will become 0100100.

Given the n numbers, print a list of operations that will make all the numbers in the circle equal to 1 using at most $7n$ operations, or state that it's impossible.

Input

The first line contains $n (3 \leq n \leq 9999)$, number of numbers in the circle, n **is odd**.

The second line contains the n numbers, each number is either 0 or 1.

Output

If there is no possible solution using at most $7n$ operations then print "-1" in a single line. Otherwise, let the number of operations be x, print x in the first line and x integers $o_1, o_2, \ldots, o_x$ on the second line, where o_i is the index in which operation i was done on.

Notes

This is how the numbers change in the first example: 01110 $\rightarrow$ 10100 $\rightarrow$ 11111

Samples

Input	Output
5 01110	2 4 2
3 100	-1

Time Limit: 1500ms
Memory Limit: 256MB

J - Two Teams

Submissions Submit

After getting many years of experience in the industry, you're now back to coach students of your past university in both Problem Solving and Cyber Security.

Being a master of these fields yourself, let alone your amazing teaching skills, your students turned out to be pretty good too!

There are two upcoming competitions, one in Problem Solving, the other in Cyber Security. But there's one issue, the contests dates coincide, so each student will have to compete in just one of the two.

In order to get the best total results, you have quantified a numerical skill level for each student in both topics, and you want to distribute your students into two equally sized groups, such that the summation of skills of both resulting teams is the maximum possible (you'd only count the skill level of the student that corresponds to the team you placed him in).

It is surely not an issue for you to solve this problem, but how fast can you do it?

Input

The first line of input will contain an integer n $(1 \leq n \leq 2 \cdot 10^5)$ --- the number of students. It is guaranteed that n is even (divisible by two).

The next n lines will contain two integers each p_i and c_i $(0 \leq p_i, c_i \leq 10^9)$ --- the i_{th} student's skill in Problem Solving and Cyber Security respectively.

Output

Output one integer, the maximum summation of skills you can get if you distribute the n students into two equally sized groups optimally.

Samples

Input	Output
2 4 2 5 6	10
4 5 6 7 2 0 5 4 7	24