A. Restoring Three Numbers

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

Polycarp has guessed three positive integers a, b and c. He keeps these numbers in secret, but he writes down four numbers on a board in arbitrary order — their pairwise sums (three numbers) and sum of all three numbers (one number). So, there are four numbers on a board in random order: $a+b$, $a+c$, $b+c$ and $a+b+c$.

You have to guess three numbers a, b and c using given numbers. Print three guessed integers in any order.

Pay attention that some given numbers a, b and c can be equal (it is also possible that $a=b=c$).

Input

The only line of the input contains four positive integers x_1, x_2, x_3, x_4 ($2 \le x_i \le 10^9$) — numbers written on a board in random order. It is guaranteed that the answer exists for the given number x_1, x_2, x_3, x_4.

Output

Print such positive integers a, b and c that four numbers written on a board are values $a+b$, $a+c$, $b+c$ and $a+b+c$ written in some order. Print a, b and c in any order. If there are several answers, you can print any. It is guaranteed that the answer exists.

Examples

input
3 6 5 4
output
2 1 3

input
40 40 40 60
output
20 20 20

input
201 101 101 200
output
1 100 100

B. Remove Duplicates

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

Petya has an array a consisting of n integers. He wants to remove duplicate (equal) elements.

Petya wants to leave only the rightmost entry (occurrence) for each element of the array. The relative order of the remaining unique elements should not be changed.

Input

The first line contains a single integer n ($1 \le n \le 50$) — the number of elements in Petya's array.

The following line contains a sequence $a_1, a_2, \ldots, a_n$ ($1 \le a_i \le 1\,000$) — the Petya's array.

Output

In the first line print integer x — the number of elements which will be left in Petya's array after he removed the duplicates.

In the second line print x integers separated with a space — Petya's array after he removed the duplicates. For each unique element only the rightmost entry should be left.

Examples

input

6 1 5 5 1 6 1
output
3 5 6 1

input
5 2 4 2 4 4
output
2 2 4

input
5 6 6 6 6 6
output
1 6

Note

In the first example you should remove two integers 1, which are in the positions 1 and 4. Also you should remove the integer 5, which is in the position 2.

In the second example you should remove integer 2, which is in the position 1, and two integers 4, which are in the positions 2 and 4.

In the third example you should remove four integers 6, which are in the positions 1, 2, 3 and 4.

C. File Name

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

You can not just take the file and send it. When Polycarp trying to send a file in the social network "Codehorses", he encountered an unexpected problem. If the name of the file contains three or more "x" (lowercase Latin letters "x") in a row, the system considers that the file content does not correspond to the social network topic. In this case, the file is not sent and an error message is displayed.

Determine the minimum number of characters to remove from the file name so after that the name does not contain "xxx" as a substring. Print `0` if the file name does not initially contain a forbidden substring "xxx".

You can delete characters in arbitrary positions (not necessarily consecutive). If you delete a character, then the length of a string is reduced by 1. For example, if you delete the character in the position 2 from the string "exxxii", then the resulting string is "exxii".

Input

The first line contains integer n $(3 \le n \le 100)$ — the length of the file name.

The second line contains a string of length n consisting of lowercase Latin letters only — the file name.

Output

Print the minimum number of characters to remove from the file name so after that the name does not contain "xxx" as a substring. If initially the file name dost not contain a forbidden substring "xxx", print `0`.

Examples

input
6 xxxiii
output
1

input
5 xxoxx
output
0

input

10 xxxxxxxxxx
output
8

Note
In the first example Polycarp tried to send a file with name contains number 33, written in Roman numerals. But he can not just send the file, because it name contains three letters "x" in a row. To send the file he needs to remove any one of this letters.

D. Bus Video System

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

The busses in Berland are equipped with a video surveillance system. The system records information about changes in the number of passengers in a bus after stops.

If x is the number of passengers in a bus just before the current bus stop and y is the number of passengers in the bus just after current bus stop, the system records the number $y - x$. So the system records show how number of passengers changed.

The test run was made for single bus and n bus stops. Thus, the system recorded the sequence of integers $a_1, a_2, \ldots, a_n$ (exactly one number for each bus stop), where a_i is the record for the bus stop i. The bus stops are numbered from 1 to n in chronological order.

Determine the number of possible ways how many people could be in the bus before the first bus stop, if the bus has a capacity equals to w (that is, at any time in the bus there should be from 0 to w passengers inclusive).

Input
The first line contains two integers n and w $(1 \le n \le 1\,000, 1 \le w \le 10^9)$ — the number of bus stops and the capacity of the bus.

The second line contains a sequence $a_1, a_2, \ldots, a_n$ $(-10^6 \le a_i \le 10^6)$, where a_i equals to the number, which has been recorded by the video system after the i-th bus stop.

Output
Print the number of possible ways how many people could be in the bus before the first bus stop, if the bus has a capacity equals to w. If the situation is contradictory (i.e. for any initial number of passengers there will be a contradiction), print 0.

Examples

input
3 5 2 1 -3
output
3

input
2 4 -1 1
output
4

input
4 10 2 4 1 2
output
2

Note
In the first example initially in the bus could be 0, 1 or 2 passengers.

In the second example initially in the bus could be 1, 2, 3 or 4 passengers.

In the third example initially in the bus could be 0 or 1 passenger.

E. Mentors

time limit per test: 3 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

In BerSoft n programmers work, the programmer i is characterized by a skill r_i.

A programmer a can be a mentor of a programmer b if and only if the skill of the programmer a is strictly greater than the skill of the programmer b $(r_a > r_b)$ and programmers a and b are not in a quarrel.

You are given the skills of each programmers and a list of k pairs of the programmers, which are in a quarrel (pairs are unordered). For each programmer i, find the number of programmers, for which the programmer i can be a mentor.

Input

The first line contains two integers n and k $(2 \le n \le 2 \cdot 10^5, 0 \le k \le \min(2 \cdot 10^5, \frac{n \cdot (n-1)}{2}))$ — total number of programmers and number of pairs of programmers which are in a quarrel.

The second line contains a sequence of integers $r_1, r_2, \ldots, r_n$ $(1 \le r_i \le 10^9)$, where r_i equals to the skill of the i-th programmer.

Each of the following k lines contains two distinct integers x, y $(1 \le x, y \le n, x \ne y)$ — pair of programmers in a quarrel. The pairs are unordered, it means that if x is in a quarrel with y then y is in a quarrel with x. Guaranteed, that for each pair (x, y) there are no other pairs (x, y) and (y, x) in the input.

Output

Print n integers, the i-th number should be equal to the number of programmers, for which the i-th programmer can be a mentor. Programmers are numbered in the same order that their skills are given in the input.

Examples

input
4 2 10 4 10 15 1 2 4 3
output
0 0 1 2

input
10 4 5 4 1 5 4 3 7 1 2 5 4 6 2 1 10 8 3 5
output
5 4 0 5 3 3 9 0 2 5

Note

In the first example, the first programmer can not be mentor of any other (because only the second programmer has a skill, lower than first programmer skill, but they are in a quarrel). The second programmer can not be mentor of any other programmer, because his skill is minimal among others. The third programmer can be a mentor of the second programmer. The fourth programmer can be a mentor of the first and of the second programmers. He can not be a mentor of the third programmer, because they are in a quarrel.

F1. Median on Segments (Permutations Edition)

time limit per test: 3 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

You are given a permutation $p_1, p_2, \ldots, p_n$. A permutation of length n is a sequence such that each integer between 1 and n occurs exactly once in the sequence.

Find the number of pairs of indices (l, r) $(1 \le l \le r \le n)$ such that the value of the median of $p_l, p_{l+1}, \ldots, p_r$ is exactly the given number m.

The median of a sequence is the value of the element which is in the middle of the sequence after sorting it in non-decreasing order. If the length of the sequence is even, the left of two middle elements is used.

For example, if $a = [4, 2, 7, 5]$ then its median is 4 since after sorting the sequence, it will look like $[2, 4, 5, 7]$ and the left of two middle elements is equal to 4. The median of $[7, 1, 2, 9, 6]$ equals 6 since after sorting, the value 6 will be in the middle of the sequence.

Write a program to find the number of pairs of indices (l, r) $(1 \le l \le r \le n)$ such that the value of the median of $p_l, p_{l+1}, \ldots, p_r$ is exactly the given number m.

Input

The first line contains integers n and m $(1 \le n \le 2 \cdot 10^5, 1 \le m \le n)$ — the length of the given sequence and the required value of the median.

The second line contains a permutation $p_1, p_2, \ldots, p_n$ ($1 \le p_i \le n$). Each integer between 1 and n occurs in p exactly once.

Output
Print the required number.

Examples

input
5 4 2 4 5 3 1
output
4

input
5 5 1 2 3 4 5
output
1

input
15 8 1 15 2 14 3 13 4 8 12 5 11 6 10 7 9
output
48

Note
In the first example, the suitable pairs of indices are: $(1, 3)$, $(2, 2)$, $(2, 3)$ and $(2, 4)$.

F2. Median on Segments (General Case Edition)

time limit per test: 3 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

You are given an integer sequence $a_1, a_2, \ldots, a_n$.

Find the number of pairs of indices (l, r) ($1 \le l \le r \le n$) such that the value of median of $a_l, a_{l+1}, \ldots, a_r$ is exactly the given number m.

The median of a sequence is the value of an element which is in the middle of the sequence after sorting it in non-decreasing order. If the length of the sequence is even, the left of two middle elements is used.

For example, if $a = [4, 2, 7, 5]$ then its median is 4 since after sorting the sequence, it will look like $[2, 4, 5, 7]$ and the left of two middle elements is equal to 4. The median of $[7, 1, 2, 9, 6]$ equals 6 since after sorting, the value 6 will be in the middle of the sequence.

Write a program to find the number of pairs of indices (l, r) ($1 \le l \le r \le n$) such that the value of median of $a_l, a_{l+1}, \ldots, a_r$ is exactly the given number m.

Input
The first line contains integers n and m ($1 \le n, m \le 2 \cdot 10^5$) — the length of the given sequence and the required value of the median.

The second line contains an integer sequence $a_1, a_2, \ldots, a_n$ ($1 \le a_i \le 2 \cdot 10^5$).

Output
Print the required number.

Examples

input
5 4 1 4 5 60 4
output
8

input
3 1 1 1 1
output
6

input
15 2 1 2 3 1 2 3 1 2 3 1 2 3 1 2 3
output
97

Note

In the first example, the suitable pairs of indices are: $(1, 3)$, $(1, 4)$, $(1, 5)$, $(2, 2)$, $(2, 3)$, $(2, 5)$, $(4, 5)$ and $(5, 5)$.