CODEFORCES β
Sponsored by Telegram

Codeforces Round #266 (Div. 2)

A. Cheap Travel

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

Ann has recently started commuting by subway. We know that a one ride subway ticket costs a rubles. Besides, Ann found out that she can buy a special ticket for m rides (she can buy it several times). It costs b rubles. Ann did the math; she will need to use subway n times. Help Ann, tell her what is the minimum sum of money she will have to spend to make n rides?

Input

The single line contains four space-separated integers n, m, a, b ($1 \le n, m, a, b \le 1000$) — the number of rides Ann has planned, the number of rides covered by the m ride ticket, the price of a one ride ticket and the price of an m ride ticket.

Output

Print a single integer — the minimum sum in rubles that Ann will need to spend.

Sample test(s)

input
6 2 1 2
output
6

input
5 2 2 3
output
8

Note

In the first sample one of the optimal solutions is: each time buy a one ride ticket. There are other optimal solutions. For example, buy three m ride tickets.

B. Wonder Room

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

The start of the new academic year brought about the problem of accommodation students into dormitories. One of such dormitories has a $a \times b$ square meter wonder room. The caretaker wants to accommodate exactly n students there. But the law says that there must be at least 6 square meters per student in a room (that is, the room for n students must have the area of at least $6n$ square meters). The caretaker can enlarge any (possibly both) side of the room by an arbitrary positive integer of meters. Help him change the room so as all n students could live in it and the total area of the room was as small as possible.

Input

The first line contains three space-separated integers n, a and b ($1 \le n, a, b \le 10^9$) — the number of students and the sizes of the room.

Output

Print three integers s, a_1 and b_1 ($a \le a_1; b \le b_1$) — the final area of the room and its sizes. If there are multiple optimal solutions, print any of them.

Sample test(s)

input
3 3 5
output
18 3 6

input
2 4 4
output
16 4 4

C. Number of Ways

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

You've got array $a[1], a[2], ..., a[n]$, consisting of n integers. Count the number of ways to split all the elements of the array into three contiguous parts so that the sum of elements in each part is the same.

More formally, you need to find the number of such pairs of indices i, j $(2 \le i \le j \le n - 1)$, that $\sum_{k=1}^{i-1} a_k = \sum_{k=i}^{j} a_k = \sum_{k=j+1}^{n} a_k$.

Input

The first line contains integer n $(1 \le n \le 5 \cdot 10^5)$, showing how many numbers are in the array. The second line contains n integers $a[1], a[2], ..., a[n]$ $(|a[i]| \le 10^9)$ — the elements of array a.

Output

Print a single integer — the number of ways to split the array into three parts with the same sum.

Sample test(s)

input
5 1 2 3 0 3
output
2

input
4 0 1 -1 0
output
1

input
2 4 1
output
0

D. Increase Sequence

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

Peter has a sequence of integers $a_1, a_2, ..., a_n$. Peter wants all numbers in the sequence to equal h. He can perform the operation of "adding one on the segment $[l, r]$": add one to all elements of the sequence with indices from l to r (inclusive). At that, Peter never chooses any element as the beginning of the segment twice. Similarly, Peter never chooses any element as the end of the segment twice. In other words, for any two segments $[l_1, r_1]$ and $[l_2, r_2]$, where Peter added one, the following inequalities hold: $l_1 \neq l_2$ and $r_1 \neq r_2$.

How many distinct ways are there to make all numbers in the sequence equal h? Print this number of ways modulo 1000000007 $(10^9 + 7)$. Two ways are considered distinct if one of them has a segment that isn't in the other way.

Input

The first line contains two integers n, h $(1 \leq n, h \leq 2000)$. The next line contains n integers $a_1, a_2, ..., a_n$ $(0 \leq a_i \leq 2000)$.

Output

Print a single integer — the answer to the problem modulo 1000000007 $(10^9 + 7)$.

Sample test(s)

input
3 2 1 1 1
output
4

input
5 1 1 1 1 1 1
output
1

input
4 3 3 2 1 1
output
0

E. Information Graph

time limit per test: 1 second
memory limit per test: 512 megabytes
input: standard input
output: standard output

There are n employees working in company "X" (let's number them from 1 to n for convenience). Initially the employees didn't have any relationships among each other. On each of m next days one of the following events took place:

- either employee y became the boss of employee x (at that, employee x didn't have a boss before);
- or employee x gets a packet of documents and signs them; then he gives the packet to his boss. The boss signs the documents and gives them to his boss and so on (the last person to sign the documents sends them to the archive);
- or comes a request of type "determine whether employee x signs certain documents".

Your task is to write a program that will, given the events, answer the queries of the described type. At that, it is guaranteed that throughout the whole working time the company didn't have cyclic dependencies.

Input

The first line contains two integers n and m ($1 \le n, m \le 10^5$) — the number of employees and the number of events.

Each of the next m lines contains the description of one event (the events are given in the chronological order). The first number of the line determines the type of event t ($1 \le t \le 3$).

- If $t = 1$, then next follow two integers x and y ($1 \le x, y \le n$) — numbers of the company employees. It is guaranteed that employee x doesn't have the boss currently.
- If $t = 2$, then next follow integer x ($1 \le x \le n$) — the number of the employee who got a document packet.
- If $t = 3$, then next follow two integers x and i ($1 \le x \le n$; $1 \le i \le$ [number of packets that have already been given]) — the employee and the number of the document packet for which you need to find out information. The document packets are numbered started from 1 in the chronological order.

It is guaranteed that the input has at least one query of the third type.

Output

For each query of the third type print "YES" if the employee signed the document package and "NO" otherwise. Print all the words without the quotes.

Sample test(s)

input
4 9 1 4 3 2 4 3 3 1 1 2 3 2 2 3 1 2 1 3 1 2 2 3 1 3
output
YES NO YES