Doremy's Connecting Plan

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

Doremy lives in a country consisting of n cities numbered from 1 to n, with a_i people living in the i-th city. It can be modeled as an undirected graph with n nodes.

Initially, there are no edges in the graph. Now Doremy wants to make the graph *connected.*

To do this, she can add an edge between i and j if

$$\sum_{k \in S} a_k \geq i \cdot j \cdot c,$$

where S is the set of all the nodes that are currently in the same connected component of either i or j, and c is a given constant.

Can Doremy make the graph connected?

Two nodes (i, j) are in the same connected component if there exists a path from i to j. A graph is connected if all its nodes are in the same connected component.

Input

The input consists of multiple test cases. The first line contains a single integer t ($1 \leq t \leq 10^4$) — the number of test cases. The description of the test cases follows.

The first line contains two integers n, c ($2 \leq n \leq 2 \cdot 10^5$, $1 \leq c \leq 10^6$) — the number of nodes and the constant.

The second line of each test case contains n integers $a_1, a_2, \ldots, a_n$ ($0 \leq a_i \leq 10^{12}$) — the number of people living in the i-th city.

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

Output

For each test case, print "`YES`" (without quotes), if it is possible to make the graph connected, and "`NO`" (without quotes) otherwise.

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

Example

standard input	standard output
7 4 10 0 20 15 10 2 1 1 1 5 1 0 1 0 4 199 5 2 1 1 3 1 1 5 5 5 6 1 10 2 5 1000000 1000000000000 1000000000000 1000000000000 1000000000000 1000000000000 3 1 0 0 2	Yes Yes Yes No No Yes No

Note

In the first test case, Doremy can add edges in the following order:

1. *Add* $(1, 2)$. *This operation is valid because* $a_1 + a_2 = 20 \geq i \cdot j \cdot c = 20$.
2. *Add* $(1, 3)$. *This operation is valid because* $a_1 + a_2 + a_3 = 35 \geq i \cdot j \cdot c = 30$.
3. *Add* $(1, 4)$. *This operation is valid because* $a_1 + a_2 + a_3 + a_4 = 45 \geq i \cdot j \cdot c = 40$.

In the second test case, Doremy can add edge $(1, 2)$ *because* $a_1 + a_2 = 2 \geq 1 \cdot 2 \cdot 1$. *After that, the graph is connected.*

In the third test case, Doremy can add edges in the order $(5, 4)$, $(5, 3)$, $(5, 2)$ *and* $(5, 1)$.

In the fourth test case, Doremy cannot add any edge at all.