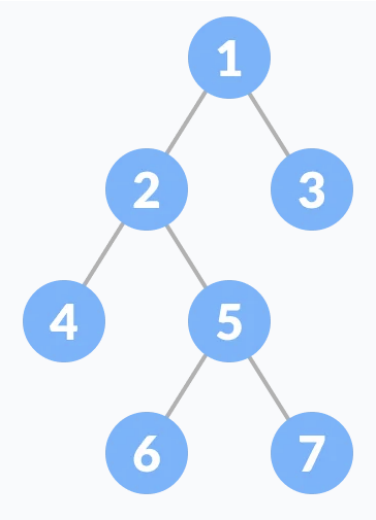
Find The Root

Problem ID: findtheroot



Full Binary Tree Example

There is a full binary tree T of size $n > 1$. Recall that a full binary tree is a binary tree in which every node has either 0 or 2 children.

Let r denote the root of T. Note that T qualifies as a full binary tree if and only if it is rooted at r. Let f be a function which takes a node u and returns the median of the sizes of the subtrees of u if T were rooted at u instead. Recall that the median of a list of even size is the mean of the 2 middle elements.

For example, in the illustrated example, $r = 1$. $f(1) = 3$, since 1 has two subtrees of sizes 1 and 5, the median of which is 3. $f(5) = 1$, since the tree rooted at node 5 has 3 subtrees, of sizes 1, 1, and 4.

A subset $S = \{u_1, u_2, \ldots, u_m\}$ of the nodes of T is chosen with $n/2 < m \leq n$. You are given the list $[f(u_1), f(u_2), \ldots, f(u_m)]$ and you have to determine whether r is in S.

Input

The first line consists of an integer m ($2 \leq m \leq 2 \cdot 10^5$), the size of S. The second line consists of m integers, the outputs of f on each node in S. It is guaranteed that a valid full binary tree T was used in generating test data for this problem.

Output

Print "yes" if r is in S and "no" if r is not in S. If you cannot determine whether r is in S from the given information, print "impossible".

Sample Input 1	Sample Output 1
2 1 2	yes

Sample Input 2	Sample Output 2
4 1 2 6 6	no