Turtle Tenacity: Continual Mods

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	256 megabytes

Given an array $a_1, a_2, \ldots, a_n$, determine whether it is possible to **rearrange its elements** into $b_1, b_2, \ldots, b_n$, such that $b_1 \bmod b_2 \bmod \ldots \bmod b_n \neq 0$.

Here $x \bmod y$ denotes the remainder from dividing x by y. Also, the modulo operations are calculated from left to right. That is, $x \bmod y \bmod z = (x \bmod y) \bmod z$. For example, $2024 \bmod 1000 \bmod 8 = (2024 \bmod 1000) \bmod 8 = 24 \bmod 8 = 0$.

Input

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

The first line of each test case contains a single integer n ($2 \leq n \leq 10^5$).

The second line of each test case contains n integers $a_1, a_2, \ldots, a_n$ ($1 \leq a_i \leq 10^9$).

The sum of n over all test cases does not exceed $2 \cdot 10^5$.

Output

For each test case, output "`YES`" if it is possible, "`NO`" otherwise.

You can output the answer in any case (upper or lower). For example, the strings "`yEs`", "`yes`", "`Yes`", and "`YES`" will be recognized as positive responses.

Example

standard input	standard output
8 6 1 2 3 4 5 6 5 3 3 3 3 3 3 2 2 3 5 1 1 2 3 7 3 1 2 2 3 1 1 2 6 5 2 10 10 10 2 4 3 6 9 3	YES NO YES NO YES NO YES NO

Note

In the first test case, rearranging the array into $b = [1, 2, 3, 4, 5, 6]$ (doing nothing) would result in $1 \bmod 2 \bmod 3 \bmod 4 \bmod 5 \bmod 6 = 1$. Hence it is possible to achieve the goal.

In the second test case, the array b must be equal to $[3, 3, 3, 3, 3]$, which would result in $3 \bmod 3 \bmod 3 \bmod 3 \bmod 3 = 0$. Hence it is impossible to achieve the goal.

In the third test case, rearranging the array into $b = [3, 2, 2]$ would result in $3 \bmod 2 \bmod 2 = 1$. Hence it is possible to achieve the goal.