Romanch and his "secret girlfriend"

HackerRank

`Time Limit: 1 second`

`Memory Limit: 256MB`

Our beloved president, Romanch bhaiya, is at a crossroads. The *hot tea* is, he (allegedly) has a secret girlfriend, and he's been consulting everyone on whether he should *finally* make the relationship public.

Being an engineer, he refuses to make a decision based on *mere feelings*. Instead, he has devised a complex, totally-not-overanalyzed test.

He has a list of n "topics of conversation" he needs to manage for the "Big Reveal." Each topic has a "chaos score" a_i. A low score is safe (e.g., "nice weather"), but a high score is *spicy* (e.g., "so, about my secret girlfriend...").

To plan the reveal, he must arrange these n topics into a sequence. To keep the conversation balanced and prevent it from exploding into drama, his plan requires a "pseudo-palindrome" structure:

- The **1st** topic he introduces must be compatible with the n**-th** (last) topic.
- The **2nd** topic must be compatible with the $(n-1)$**-th** topic.
- ...and so on, for all i from 1 to n.

Two topics x and y are "compatible" if the absolute difference of their chaos scores is no more than d (i.e., $|x-y| \leq d$).

Is it possible for Romanch to find *any* rearrangement of his n conversation topics such that for all i (from 1 to n), the i-th topic and the $(n+1-i)$-th topic are compatible?

If a low-drama arrangement exists, output 'Yes' (it's time to go public!). Otherwise, output 'No' (the secret stays safe, for now...).

Input Format

- The first line of input contains a single integer T, denoting the number of test cases.
- Each test case consists of two lines of input.
  - The first line contains two space-separated integers n and d — the length of the array (number of topics) and the maximum allowed difference.
  - The second line contains n space-separated integers $a_1, a_2, \ldots, a_n$ — the chaos scores of the topics.

Constraints

- $1 \leq T \leq 10^4$
- $1 \leq n \leq 10^5$

- $1 \leq a_i \leq 10^9$
- $0 \leq d \leq 10^9$
- The sum of n over all test cases won't exceed 10^5.

Output Format

- For each test case, output on a new line the answer: **'YES'** if a valid rearrangement exists, and **'NO'** otherwise.

Note: The output is **not case sensitive**. You can print `YES`, `yes`, `Yes`, `NO`, `no`, `nO`, etc. All will be accepted.

Sample Input 0

```
3
1 1
1
2 0
1 2
2 1000
1 2
```

Sample Output 0

```
YES
NO
YES
```

Explanation 0

Test Case 1:

- $n = 1, d = 1$, array `[1]`.
- With only 1 topic, there are no pairs to check. Since n is odd, the single element is valid on its own.
- Output: **YES**

Test Case 2:

- $n = 2, d = 0$, array `[1, 2]`.
- We must pair `(1, 2)` there are 2 possible ways `[1,2]` or `[2,1]` .
- in both the cases Difference: $|1 - 2| = 1$.
- Since $1 > d$ (which is 0), this pair is invalid.
- Output: **NO**

Test Case 3:

- $n = 2, d = 1000$, array `[1, 2]`.
- We must pair `(1, 2)` there are 2 possible ways `[1,2]` or `[2,1]`.

- in both the cases Difference: $|1 - 2| = 1$.
- Since $1 \leq 1000$, this pair is valid.
- Output: **YES**