

Chosen Pair

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

You are playing a game with Soum.

Soum gives you an array of N integers, called arr , all of which are in the range from 1 to N . From this array, Soum picks out two indices of his choosing, say i and j (Soum can pick $i = j$ or $i \neq j$). Then, Soum sets $A = arr[i]$ and $B = arr[j]$.

Your objective is to find the numbers A and B . To find these integers, you can ask Soum about a value X and its relation to A and B . To answer your question, Soum will (honestly) tell you whether $A \geq X$ and also whether $B \geq X$.

Soum allows you to ask Q questions. Can you create a strategy which is guaranteed to find the values of A and B using at most Q queries? Output “YES” if such a strategy exists, or “NO” if such a strategy does not exist.

Please note that you will ask queries one after the other (and not all at the same time), and therefore, after you ask one query, you will be able to use the answer to that query to influence your subsequent queries. You need to find whether there exists a strategy which can find both A and B in at most Q queries, regardless of what values of A and B Soum selects.

Please also note that this is **not** an interactive problem. In other words, you do **not** need to interact with Soum and find his numbers. You only need to find out whether it is possible to find both of Soum’s numbers using Q queries.

Input

The first line of input contains two space-separated integers, N and Q . N is the size of Soum’s array. Q is the number of questions Soum allows you to ask.

The second line contains N space-separated integers, which refer to the values present in Soum’s array, arr .

Output

Output “YES” if there is a strategy to find both of Soum’s numbers in Q queries, or “NO” if such a strategy does not exist.

Scoring

The input is divided into multiple subtasks. Each subtask contains a number of testcases. The score for a subtask is awarded if your code passes all the testcases within the subtask.

The score of your submission is the sum of the scores for all subtasks it solves correctly.

In all subtasks, $1 \leq N \leq 5000$ and $1 \leq Q \leq 5000$.

- Subtask 1 [7 points]: $Q = N$
- Subtask 2 [10 points]: $Q = 1$
- Subtask 3 [12 points]: $Q \leq 2$
- Subtask 4 [17 points]: $Q \leq 3, N \leq 500$
- Subtask 5 [23 points]: $Q \leq 3$
- Subtask 6 [31 points]: No additional constraints.

Examples

| standard input | standard output |
|------------------|-----------------|
| 3 2 1 3 2 | YES |
| 4 2 4 1 2 3 | NO |
| 3 3 1 1 1 | YES |
| 5 1 1 1 1 2 2 | YES |

Note

For the first sample testcase, Soum can pick A as any number between 1 and 3. Similarly, he can also pick B as any number between 1 and 3. You are allowed to ask Soum 2 queries. You can first ask him if both numbers are greater than or equal to 3. If any answer is yes, the corresponding number must be 3. If any answer is no, you can then ask Soum whether any number is greater than or equal to 2. If A or B was not greater than or equal to 3, but is greater than or equal to 2, that number must be 2. If any of A or B was not greater or equal to 3 or 2, then that number must equal 1. Therefore, with two queries, it is possible to find both A and B , regardless of what values of A or B Soum chooses.

For the second case, it can be shown that no two queries will help you find the exact values of both of Soum's numbers.

For the third case, Soum must pick both A and B as 1. Even without asking any queries you will know that both values are 1, and therefore it is possible to find both of Soum's numbers in 3 queries or fewer.

For the last case, it can be shown that you are able to use at most 1 query to find both of Soum's numbers.