# Problem 1338: Reduce Array Size to The Half

## Problem Information

**Difficulty:** Medium
**Acceptance Rate:** 69.28%
**Paid Only:** No
**Tags:** Array, Hash Table, Greedy, Sorting, Heap (Priority Queue)

## Problem Description

You are given an integer array `arr`. You can choose a set of integers and remove all the occurrences of these integers in the array.

Return _the minimum size of the set so that**at least** half of the integers of the array are removed_.

**Example 1:**

**Input:** arr = [3,3,3,3,5,5,5,2,2,7] **Output:** 2 **Explanation:** Choosing {3,7} will make the new array [5,5,5,2,2] which has size 5 (i.e equal to half of the size of the old array). Possible sets of size 2 are {3,5},{3,2},{5,2}. Choosing set {2,7} is not possible as it will make the new array [3,3,3,3,5,5,5] which has a size greater than half of the size of the old array.

**Example 2:**

**Input:** arr = [7,7,7,7,7,7] **Output:** 1 **Explanation:** The only possible set you can choose is {7}. This will make the new array empty.

**Constraints:**

* `2 <= arr.length <= 105` * `arr.length` is even. * `1 <= arr[i] <= 105`

## Code Snippets

**C++:**

```
class Solution {
public:
int minSetSize(vector<int>& arr) {

}
};
```

**Java:**

```
class Solution {
public int minSetSize(int[] arr) {

}
}
```

**Python3:**

```
class Solution:
def minSetSize(self, arr: List[int]) -> int:
```