

## 217. Contains Duplicate

Given an integer array of nums, return true if any value appears at least twice in the array. False otherwise.

Ex.]

Input: [ 1 2 3 1 ]

Output: True

Input: [ 1 2 3 4 ]

Output: False **No Duplicates**

Approaches:

- 1.) Sort array
- 2.) Enumerate over array to  $\text{len}(\text{array}) - 1$
- 3.) For each iteration, if  $\text{array}[i] == \text{array}[i+1]$ , return true.
- 4.) If no iteration comparisons were true, return False

Time:  $O(n \log n)$

Space:  $O(n)$

- 1.) Create a set of numbers
- 2.) Iterate over each value in array.
- 3.) If set contains value, return True. Otherwise insert value in s.
- 4.) Return False.

Time:  $O(n)$

Space:  $O(n)$

```
auto containsDuplicate(Container && container) {  
    using Type = typename Container::value_type;  
    std::unordered_set<Type> s;  
    for (const auto& value: container) {  
        if (s.contains(value)) {  
            return true;  
        }  
        s.insert(value);  
    }  
    return false;  
}
```