2870. Minimum Number of Operations to Make Array Empty

Given an array of numbers, you can perform one of two operations.

- Choose 2 numbers of equal value and delete them.
- Choose 3 numbers of equal value and delete them.

Return minimum # of operations to make array empty. -1 if not possible.

Approach #1

1.) Sort the array

2.) Find first count. Subtract 3 until

val = 4 => 2

val = 3 => 1

val = 2 => 1

3.) If value is 1 return -1

I'm not actually positive this will work for all conditions. Probably will. Regardless, we sort which is O(nlogn)

1.) Count quantity in map.

2.) For each count in map, do exactly step 2.
We actually only have to check for count = 1 and decrement by two until count % 3 = 0. Else return -1.

```cpp
int  minOperations( const  std::vector<int>& nums ) {
    std::unordered_map<int, int>  m;
    for ( const  auto  num :  nums ) {
        m[ num ] += 1;
    }

    int  result {};
    for ( auto [ key, value ] :  m ) {
        while( value % 3 != 0  && value > 1) {
            value -= 2;
            ++result;
        }
        if (value == 1) {
            return -1;
        }
        result += value / 3;
    }

    return result;
}
```