APPRENTICE TECHNICAL LOG DOC

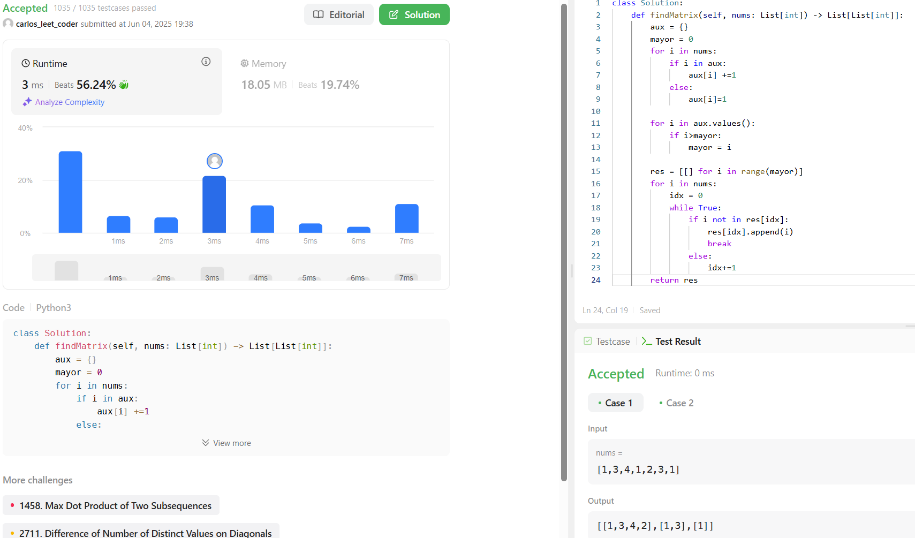
CONVERT AN ARRAY INTO A 2D ARRAY WITH CONDITIONS

Leonardo Trevizo Herrera

OVERVIEW – In this document we will explore the process and challenges I had to overcome while working on the first breakable toy for the spark program  
  
CONTEXT – We want to get the permutations from a given array of numbers  
<https://github.com/technologic-technologic/LeetCode.git>  
  
SOLUTION (YOUR APPROACH) – I think this problem was easier than the permutations one. My first thought was to count how many times every number is repeated within the input array. The largest number obtained will be number of arrays (this way we create the minimum number of list).

Once I know how many subarrays I need, I can loop through every element in the input array.

During this loop, I use an auxiliary variable to keep track of the current subarray index, so that if the current number already exists in a subarray, I can place it in the next one.

After processing all elements, I return the list of subarrays.in a O(n), to be specific the time complexity in the worst-case scenario is nxm where m is the number of arrays.  
  


ALTERNATIVE SOLUTIONS – To be honest, I didn't think of any other approach, but I'm thinking of two nested for loops. The first loop would iterate over the elements of the input array. The second loop, combined with an if condition, would check whether the element already exists in the current subarray. If it doesn't, we append it to that subarray; otherwise, we check the next subarray. When there are no more subarrays to check, create a new subarray, and append the value to it. Something like that hehe