

Problem Statements: List

- 1. Rotate a list by n places:**
Input: lst = [1, 2, 3, 4, 5] , n = 2
Output: [4, 5, 1, 2, 3]
- 2. Find all unique triplets in the list that sum up to zero:**
Input: lst = [-1, 0, 1, 2, -1, -4]
Output: [[-1, 0, 1], [-1, -1, 2]]
- 3. Find the longest consecutive subsequence in a list:**
Input: lst = [100, 4, 200, 1, 3, 2]
Output: 4 (The longest consecutive sequence is [1, 2, 3, 4])
- 4. Product of every other element:**
Input: lst = [1, 2, 3, 4, 5]
Output: 15 (The product of elements at indices 0, 2, 4 is 1*3*5)
- 5. Find the most frequent element in a list:**
Input: lst = [1, 3, 1, 3, 2, 1]
Output: 1
- 6. Check if a list is a palindrome:**
Input: lst = [1, 2, 3, 2, 1]
Output: True
- 7. Find all pairs in a list whose sum is equal to a given number:**
Input: lst = [1, 2, 3, 4, 3, 6] , sum = 6
Output: [(2, 4), (3, 3)]
- 8. Flatten a nested list:**
Input: lst = [[1, 2, 3], [4, 5], [6, [7, 8]]]
Output: [1, 2, 3, 4, 5, 6, 7, 8]
- 9. Find the maximum product of two integers in a list:**
Input: lst = [1, 20, 30, 4, 5]
Output: 600 (Product of 20 and 30)
- 10. Rearrange the list such that positive numbers come before negative numbers:**
Input: lst = [-1, 2, -3, 4, 5, -6]
Output: [2, 4, 5, -1, -3, -6]
- 11. Find the smallest positive integer missing from a list:**
Input: lst = [3, 4, -1, 1]
Output: 2
- 12. Find the subarray with the maximum sum:**
Input: lst = [-2, 1, -3, 4, -1, 2, 1, -5, 4]
Output: 6 (The subarray [4, -1, 2, 1] has the maximum sum)
- 13. Find the intersection of two lists:**
Input: lst1 = [1, 2, 3, 4] , lst2 = [3, 4, 5, 6]
Output: [3, 4]
- 14. Group elements of the list based on their parity:**
Input: lst = [1, 2, 3, 4, 5, 6]
Output: [[2, 4, 6], [1, 3, 5]] (Even numbers first, followed by odd numbers)
- 15. Partition the list into sublists of length n:**
Input: lst = [1, 2, 3, 4, 5, 6, 7, 8, 9] , n = 3
Output: [[1, 2, 3], [4, 5, 6], [7, 8, 9]]
- 16. Find the majority element in a list:**
Input: lst = [1, 2, 3, 1, 1]
Output: 1
- 17. Replace all occurrences of a given value with another value:**
Input: lst = [1, 2, 3, 1, 2, 3] , old_value = 2 , new_value = 4
Output: [1, 4, 3, 1, 4, 3]
- 18. Count the number of sublists in a list:**
Input: lst = [1, [2, 3], 4, [5, 6], 7, [8, 9]]
Output: 3
- 19. Find the first missing positive integer in a list:**
Input: lst = [3, 4, -1, 1]
Output: 2
- 20. Generate all permutations of a list:**
Input: lst = [1, 2, 3]
Output: [[1, 2, 3], [1, 3, 2], [2, 1, 3], [2, 3, 1], [3, 1, 2], [3, 2, 1]]