1.Remove Element:

def remove \_ element(nums, val):

k = 0

for num in nums:

if num != val:

nums[k] = num

k += 1

return k

nums = [3, 2, 2, 3]

val = 3

k = remove\_element(nums, val)

print(f"Output: {k}, nums = {nums[:k]}")

2.LENGTH OF LAST WORD:

def length\_of\_last\_word(s):

words = s.split()

if len(words) == 0:

return 0

return len(words[-1])

s = "Hello World"

print(length\_of\_last\_word(s))

3. max subarray:

def max\_subarray\_sum(nums):

max\_sum = float('-inf')

for i in range(len(nums)):

current\_sum = 0

for j in range(i, len(nums)):

current\_sum += nums[j]

max\_sum = max(max\_sum, current\_sum)

return max\_sum

nums = [-2, 1, -3, 4, -1, 2, 1, -5, 4]

print(max\_subarray\_sum(nums))

4. Permutation Sequence:

import math

def getPermutation(n, k):

numbers = list(range(1, n+1))

k -= 1

factorial = math.factorial(n)

result = []

for i in range(n, 0, -1):

factorial //= i

index = k // factorial

k %= factorial

result.append(str(numbers[index]))

numbers.pop(index)

return "".join(result)

n = 3

k = 3

output = getPermutation(n, k)

print(output)

5. permutation II:

from itertools import permutations

def unique\_permutations(nums):

return list(set(permutations(nums)))

# Test the function with the given example

nums = [1, 1, 2]

print(unique\_permutations(nums))