Question 1:

Given a string s consisting of words and spaces, return the length of the last word in the string. A word is a maximal substring consisting of non-space characters only. def lengthOfLastWord(s): # Split the string by spaces and remove empty strings words = s.split()

# Check if there are words in the listif words: # Get the last word and return its length return len(words[-1])else: # If there are no words, return 0 return 0s1 = "Hello World" print(lengthOfLastWord(s1)) # Output: 5s2 = " fly me to the moon " print(lengthOfLastWord(s2)) # Output: 4

Question 2:

Given an array nums of n integers, return an array of all the unique quadruplets [nums[a], nums[b], nums[c], nums[d]] such that: def fourSum(nums, target): nums.sort() quadruplets = [] n = len(nums)

for i in range(n - 3): if i > 0 and nums[i] == nums[i - 1]: continue for j in range(i + 1, n - 2): if j > i + 1 and nums[j] == nums[j - 1]: continue left = j + 1 right = n - 1 while left < right: current\_sum = nums[i] + nums[j] + nums[left] + nums[right] if current\_sum == target: quadruplets.append([nums[i], nums[j], nums[left], nums[right]]) while left < right and nums[left] == nums[left + 1]: left += 1 while left < right and nums[right] == nums[right - 1]: right -= 1 left += 1 right -= 1 elif current\_sum < target: left += 1 else: right -= 1return quadrupletsnums1 = [1, 0, -1, 0, -2, 2]target1 = 0 print(fourSum(nums1, target1)) # Output: [[-2, -1, 1, 2], [-2, 0, 0, 2], [-1, 0, 0, 1]]nums2 = [2, 2, 2, 2, 2] target2 = 8 print(fourSum(nums2, target2)) # Output: [[2, 2, 2, 2]]