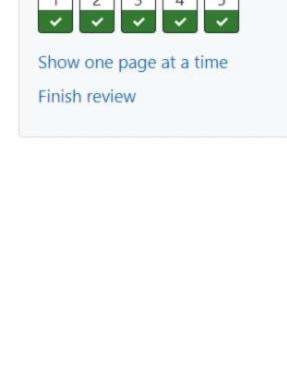
ANJALI G S 2022-BIOMED-A A2 V REC-PS GE19211 / GE23233 / GE23231 - PSPP/PUP

To find the frequency of numbers in a list and display in sorted order.



Quiz navigation

Dashboard / My courses / PSPP/PUP / Searching techniques: Linear and Binary / Week10\_Coding

Question 1

Started on Saturday, 25 May 2024, 8:31 AM

Completed on Saturday, 25 May 2024, 8:53 AM

Grade 100.00 out of 100.00

State Finished

Time taken 22 mins 15 secs

Marks 5.00/5.00

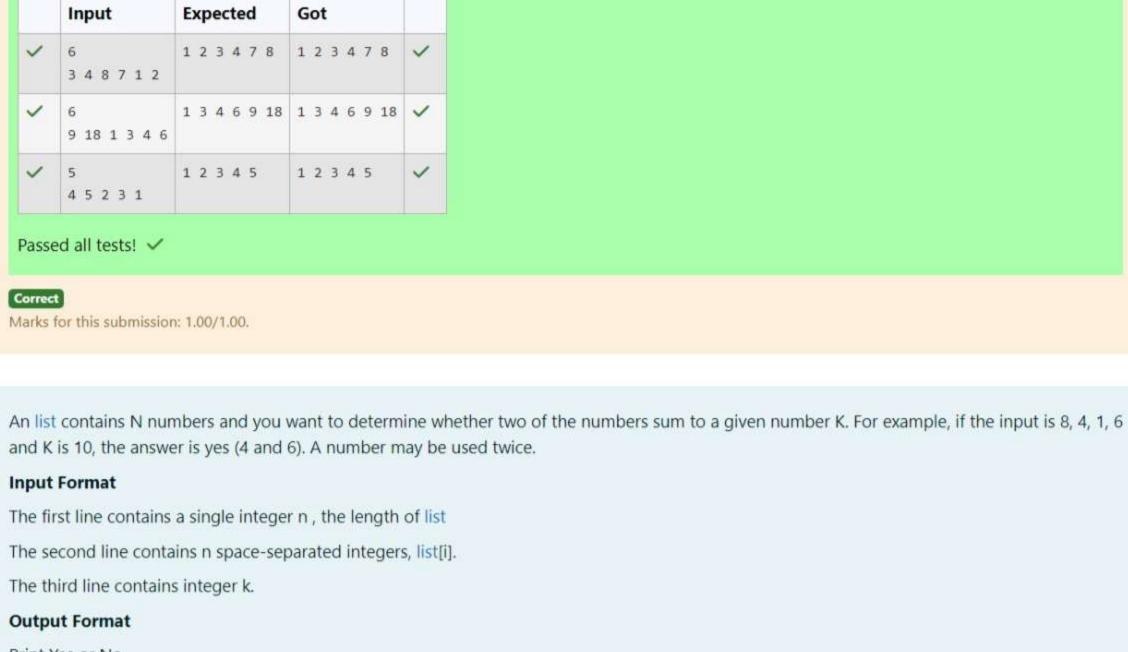
Correct Constraints: Mark 1.00 out of 1.00 1<=n, arr[i]<=100 F Flag question Input: 1 68 79 4 90 68 1 4 5 output: 12 42 51 68 2 79 1 90 1 For example: Input Result

4 3 5 3 4 5 3 2 4 2 5 2 Answer: (penalty regime: 0 %) 1 def frequency\_counter(arr): frequency\_dict = {} 3 4 for num in arr: if num in frequency\_dict: 4 1 frequency\_dict[num] += 1 5 6 1 else: frequency\_dict[num] = 1 7 sorted frequency = sorted(frequency dict.items()) 8 for num, freq in sorted\_frequency: 9 print(num, freq) 10 arr = list(map(int, input().split())) 11 frequency\_counter(arr) 12 13 14 15



Question 2 Correct Mark 1.00 out of P Flag question For example: Input Result 3 4 8 7 1 2 45231 2 3 +

123478 1 2 3 4 5 Answer: (penalty regime: 0 %) 1 def bubble\_sort(arr): n = len(arr)for i in range(n): for j in range(0, n-i-1): if arr[j] > arr[j+1]: 5 + arr[j], arr[j+1] = arr[j+1], arr[j]6 7 n = int(input()) 8 | arr = list(map(int, input().split())) 9 bubble\_sort(arr) 10 print(\*arr) 11 Got Input Expected 123478 123478 🗸



Question 3

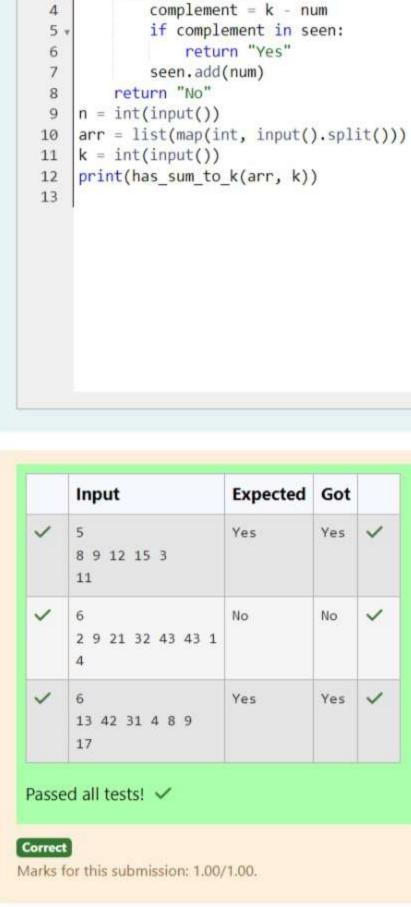
Mark 1.00 out of

F Flag question

Correct

1.00

Print Yes or No. Sample Input 0124653 **Sample Output** Yes For example: Input Result Yes 8 9 12 15 3 11 2 9 21 32 43 43 1 Answer: (penalty regime: 0 %) 1 def has\_sum\_to\_k(arr, k): seen = set() 2 3 + for num in arr:



Question 4

Mark 1.00 out of

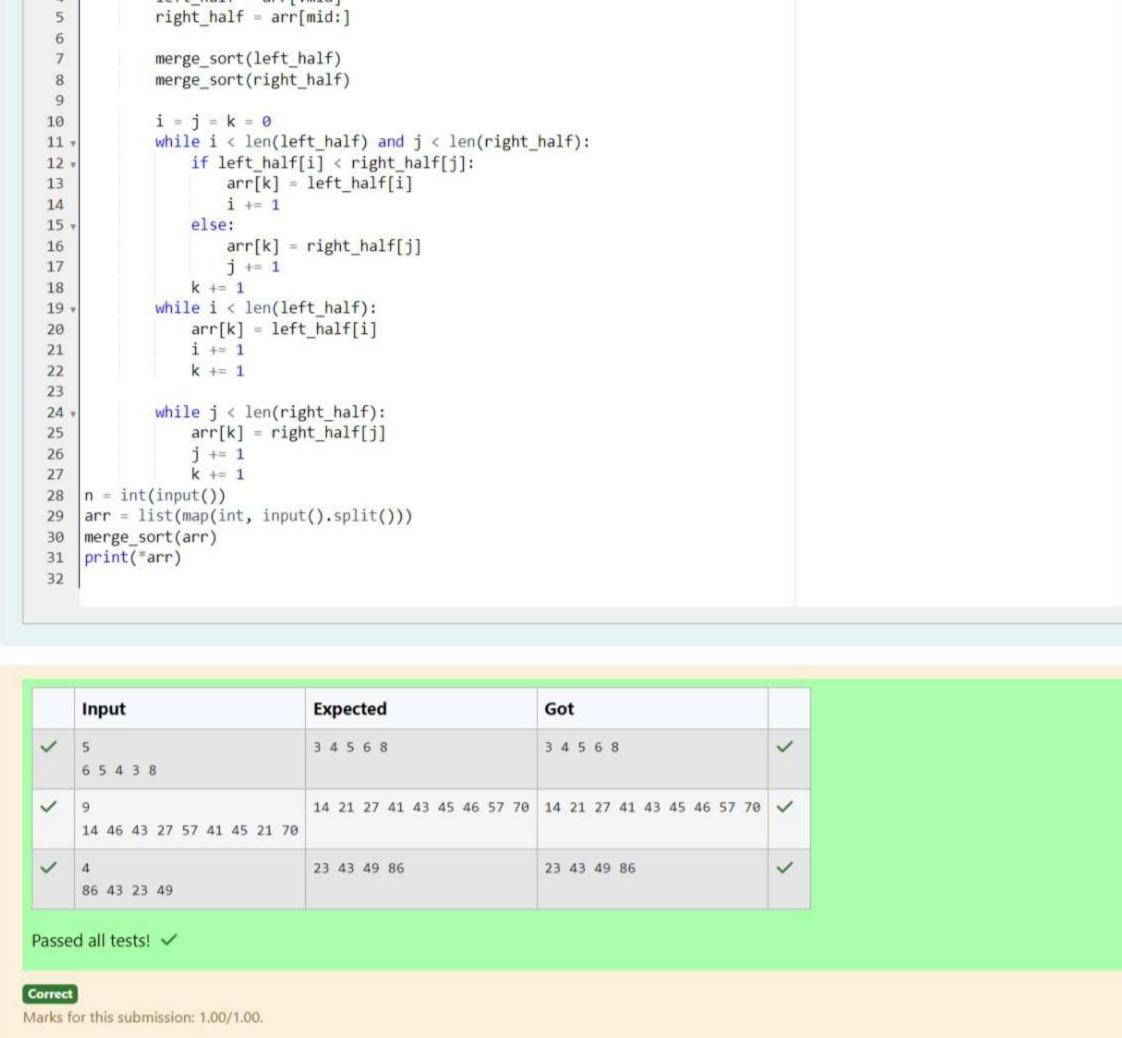
F Flag question

Correct

Expected Got

Yes 🗸

Yes Yes 🗸 Marks for this submission: 1.00/1.00. Write a Python program to sort a list of elements using the merge sort algorithm. For example: Input Result 3 4 5 6 8 6 5 4 3 8 Answer: (penalty regime: 0 %) 1 def merge\_sort(arr): if len(arr) > 1: mid = len(arr) // 2left\_half = arr[:mid] right\_half = arr[mid:] merge\_sort(left\_half) 8 merge\_sort(right\_half) 9 i = j = k = 010 while i < len(left\_half) and j < len(right\_half): 11 if left\_half[i] < right\_half[j]:</pre> 12 arr[k] = left\_half[i] 13 i += 1 14 15 else: arr[k] = right\_half[j] 16 17 j += 1 18 k += 119 while i < len(left\_half):</pre> arr[k] = left\_half[i] 20 i += 1 21 22 k += 123 while j < len(right\_half):</pre> 24 arr[k] = right\_half[j] 25 j += 1 26

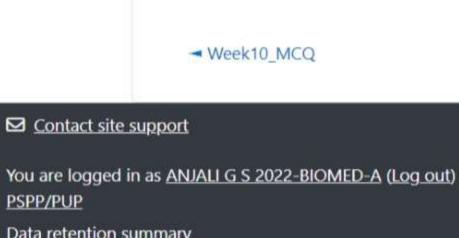


Question **5** Given an list, find peak element in it. A peak element is an element that is greater than its neighbors. Correct An element a[i] is a peak element if Mark 1.00 out of  $A[i-1] \le A[i] \ge a[i+1]$  for middle elements.  $[0 \le i \le n-1]$ Flag question  $A[i-1] \le A[i]$  for last element [i=n-1]A[i] > = A[i+1] for first element [i=0] **Input Format** The first line contains a single integer n, the length of A. The second line contains n space-separated integers, A[i]. **Output Format** 

Print peak numbers separated by space. Sample Input 891026 Sample Output 10 6 For example: Input Result 12 8 12 3 6 8 Answer: (penalty regime: 0 %) 1 - def find\_peak(arr): peak\_elements = [] if arr[0] >= arr[1]: 3 + peak\_elements.append(arr[0]) 4 for i in range(1, len(arr) - 1): 5 1 if arr[i - 1] <= arr[i] >= arr[i + 1]: 6 4 peak\_elements.append(arr[i]) 8 , if arr[-1] >= arr[-2]: peak\_elements.append(arr[-1]) 9 10 return peak\_elements 11 n = int(input()) 12 arr = list(map(int, input().split())) 13 peak\_elements = find\_peak(arr) 14 print(\*peak\_elements) 15 16

Finish review

Sorting -



Input

12 3 6 8

Passed all tests! <

Correct

15 7 10 8 9 4 6

Marks for this submission: 1.00/1.00.

**Expected Got** 

12 8

15 10 9 6 15 10 9 6 🗸

12 8

Jump to...

Data retention summary

PSPP/PUP