**Task 5 a:**

**program**

def sortedSquares(nums):

n = len(nums)

res = [0] \* n

I,r = 0, n-1

pos=n-1 # Fill from the end

while I <= r:

if abs(nums[1]) > abs(nums[r]):

res[pos] = nums[1] \* nums[1]

I+=1

else:

res[pos] = nums[r] \* nums[r]

r-=1

pos -= 1

return res

#-------- Function Calling & Printing\_\_\_\_\_\_\_\_\_\_\_

nums1 = [-4, -1, 0, 3, 10]

nums2 = [-7, -3, 2, 3, 11]

print("Input:", nums1)

print("Output:", sortedSquares (nums1))

print("\nInput:", nums2)

print("Output:", sortedSquares(nums2))

**Example 1:**

**Input:** nums = [-4,-1,0,3,10]

A screenshot of a math problem

AI-generated content may be incorrect.

**Output:** [0,1,9,16,100]

**Task 5 b:**

**program**

A screenshot of a computer program

AI-generated content may be incorrect.

**Input:**

Enter number of elements: 5

Enter the elements: 64 34 25 12 22

**Output:**

Sorted List: [12, 22, 25, 34, 64]

**Task 5 c:**

**program**

A screenshot of a graph

AI-generated content may be incorrect.

**Sample I/O**

Enter number of elements: 6

Enter elements: 34 12 45 2 18 7

Original List: [34, 12, 45, 2, 18, 7]

Sorted List: [2, 7, 12, 18, 34, 4]

**Task 5 d:**

**program**

A screenshot of a computer code

AI-generated content may be incorrect.

**Sample I/O**

Enter sorted elements: -10 -3 0 5 9 12

Enter target element: 9

Target found at index 4

**Task 5 e:**

**program**

A screenshot of a computer program

AI-generated content may be incorrect.

**Sample I/O**

Enter number of elements: 7

Enter elements: 1 3 2 5 7 6 4

Peak elements: 3 7