1. **Same or Not**

n1 = int(input())

n2 = int(input())

f = True

f = n1==n2

s1 = 0

s2 = 0

for i in range(n1):

h = int(input())

s1 += h

for i in range(n2):

h = int(input())

s2 += h

f &= s1==s2

if f:

print('Same')

else:

print('Not Same')

1. **Count Distinct elements**

n = int(input().strip())

arr = [int(input().strip()) for i in range(n)]

distinct\_elements = []

for i in range(n):

if arr[i] not in distinct\_elements:

distinct\_elements.append(arr[i])

print("There are", len(distinct\_elements), "distinct element in the array.")

1. **Compatible Array**

n1 = int(input().strip())

arr1 = [int(input().strip()) for i in range(n1)]

n2 = int(input().strip())

arr2 = [int(input().strip()) for i in range(n2)]

if n1 != n2:

print("Incompatible")

else:

for i in range(n1):

if arr1[i] < arr2[i]:

print("Incompatible")

break

else:

print("Compatible")

1. **Sum of even numbers and odd numbers**

n = int(input().strip())

arr = [int(input().strip()) for i in range(n)]

even\_sum = 0

odd\_sum = 0

for i in range(n):

if arr[i] % 2 == 0:

even\_sum += arr[i]

else:

odd\_sum += arr[i]

print("The sum of the even numbers in the array is", even\_sum)

print("The sum of the odd numbers in the array is", odd\_sum)

1. **Ascending Order**

n = int(input().strip())

arr = [int(input().strip()) for i in range(n)]

arr.sort()

print("The Sorted array is:")

for i in range(n):

print(arr[i])

1. **Queue**

# Get the inputs

n, m = map(int, input().split())

a = list(map(int, input().split()))

# Initialize the bus count and the remaining seats

bus\_count = 0

remaining\_seats = m

# Loop through the groups

for i in range(n):

# If the current group is larger than the remaining seats

if a[i] > remaining\_seats:

# Increase the bus count and reset the remaining seats

bus\_count += 1

remaining\_seats = m

# Deduct the number of people in the current group from the remaining seats

remaining\_seats -= a[i]

# Check if there are still people left after the loop

if remaining\_seats < m:

# Increase the bus count

bus\_count += 1

# Print the bus count

print(bus\_count)

1. **Array Insertion**

n = int(input())

arr = [int(input()) for i in range(n)]

pos = int(input())

if pos > n:

print("Invalid Input")

else:

ele = int(input())

print("Array after insertion is")

for i in range(n):

if i == pos - 1:

print(ele)

print(arr[i])

1. **Remove Duplicate Elements**

n = int(input())

a = [int(input()) for i in range(n)]

flag = 0

for i in range(n):

flag = 0

for j in range(i, -1, -1):

if a[i] == a[j]:

flag += 1

if flag == 1:

print(a[i])

1. **Online Game**

n = int(input().strip())

a = list(map(int, input().strip().split()))[:n]

p1 = 0

p2 = n - 1

while p1 < p2:

while a[p1] % 2 == 0 and p1 < p2:

p1 += 1

while a[p2] % 2 != 0 and p1 < p2:

p2 -= 1

if p1 < p2:

a[p1], a[p2] = a[p2], a[p1]

p1 += 1

p2 -= 1

print("Array after Segregation")

print(\*a)

1. **Toyland**

num = int(input().strip())

houses = []

for i in range(num):

houseNum, pos = map(int, input().strip().split())

houses.append((houseNum, pos))

houses = sorted(houses, key=lambda x: x[1])

max\_distance = 0

house1, house2 = 0, 0

for i in range(1, num):

distance = houses[i][1] - houses[i-1][1]

if distance > max\_distance:

max\_distance = distance

house1, house2 = houses[i-1][0], houses[i][0]

if house1 > house2:

house1, house2 = house2, house1

print(house1, house2)

1. **Pair the container**

numContainers = int(input())

containers = list(map(int, input().split()))

containers.sort(reverse=True)

left = 0

right = numContainers - 1

while left < right:

print(containers[left], containers[right])

left += 1

right -= 1

if numContainers % 2 != 0:

print(containers[left], 0)

1. **Smallest Positive Missing Number**

n = int(input())

a = [int(x) for x in input().split()]

a.sort()

missing = 1

for i in range(n):

if a[i] <= 0:

continue

if a[i] == missing:

missing += 1

elif a[i] > missing:

break

print(missing)