#### **Ex. No. : 6.1** Element Insertion

**Program:**

x=[] for i in range(0,11): b=int(input()) x.append(b) #a.sort() print("ITEM to be inserted:",x[-1],sep='') x.sort() print("After insertion array is:") for i in x:

print(i)

**Ex. No. : 6.2** Anagram

**Program:** def index\_mapping(A, B):

index\_map = {num: i for i, num in enumerate(B)} return ' '.join(str(index\_map[num]) for num in A) n=int(input())

A = list(map(int, input().split())) B = list(map(int, input().split())) print(index\_mapping(A, B))

#### **Ex. No. : 6.3** Merge Two Sorted Arrays WithoutDuplication

**Program:**

n1=int(input()) l1=[] for i in range(0,n1): a=int(input()) l1.append(a) n2=int(input()) l2=[] for i in range(0,n2): a=int(input()) l2.append(a) l3=[] l3.extend(l1) l3.extend(l2) a=list(set(l3)) a.sort() for i in a: print(i,end=' ') n1=int(input()) l1=[] for i in range(0,n1): a=int(input()) l1.append(a) n2=int(input()) l2=[] for i in range(0,n2): a=int(input()) l2.append(a) l3=[] l3.extend(l1) l3.extend(l2) a=list(set(l3)) a.sort() for i in a: print(i,end=' ')

#### **Ex. No. : 6.4** Distinct Elements in an Array

**Program:**

n = int(input()) arr = [] for \_ in range(n): arr.append(int(input())) distinct\_elements = set(arr) print(\*distinct\_elements)

**Ex. No. : 6.5** The Pivot

**Program:**

a = int(input()) b= [] for i in range(a): element = int(input()) b.append(element) total= sum(b) left= 0 right = total- b[0] if left== right: print(0) exit() for i in range(1, a): left+= b[i - 1] right-= b[i] if left== right:

print(i) break

#### **Ex. No. : 6.6** Intersection of array

**Program:** t=int(input()) l1=list() while(t!=0): n1=int(input()) l1=[] l2=[] for i in range(0,n1): a=int(input()) l1.append(a) n2=int(input()) for i in range(0,n2): a=int(input()) l2.append(a) t=t-1 c=set(l1) d=set(l2) e=list(c.intersection(d)) e.sort() for i in e: print(i,end=' ') print('\n')

#### **Ex. No. : 6.7** Location

**Program:**

n = int(input())

arr = [int(input()) for \_ in range(n)] element\_to\_search = int(input()) locations = [] occurrences = 0 for i in range(len(arr)): if arr[i] == element\_to\_search:

locations.append(i + 1) occurrences +=1 if occurrences == 0:

print(f"{element\_to\_search} is not present in the array.") else:

for loc in locations:

print(f"{element\_to\_search} is present at location {loc}.") print(f"{element\_to\_search} is present {occurrences} times in the array.")

#### **Ex. No. : 6.8** Strictly increasing

**Program:** def check\_increasing\_or\_decreasing(lst):

increasing = True decreasing = True for i in range(1, len(lst)): if lst[i] > lst[i - 1]: decreasing = False elif lst[i] < lst[i - 1]: increasing = False return increasing or decreasing

def check\_strictly\_increasing\_with\_removal(lst):

for i in range(len(lst)):

temp\_lst = lst[:i] + lst[i+1:]

if check\_increasing\_or\_decreasing(temp\_lst):

returnTrue return False

n = int(input()) lst = [] for \_ in range(n):

lst.append(int(input()) if check\_increasing\_or\_decreasing(lst) or check\_strictly\_increasing\_with\_removal(lst): print("True") else: print("False")

#### **Ex. No. : 6.9** Merge List

**Program:**

m=int(input()) n=int(input()) l1=[] l2=[] c=1 for i in range(0,m\*n\*2,2): a=int(input()) b=int(input()) if c%2!=0: l1.append(a) l1.append(b) else: l2.append(a) l2.append(b) c=c+1 l3=[] l3.append(l1) l3.append(l2) print(l3)

#### **Ex. No. : 6.10** Check pair with difference k

**Program:**

t=int(input()) for i in range(0,t): n=int(input()) l=[] for j in range(0,n): a=int(input())

l.append(a) p=int(input()) for k in range(0,n):

c=0 for m in range(i+1,n): if l[m]-l[k]==p: c=1 print('1') break

if c==1: break if c==0: print('0')