10 - Searching & Sorting

Ex. No. : 10.1 **Date: 01.06.24**

Register No.: 230701383 **Name Vinith B**

.

Bubble Sort

Bubble Sort is the simplest sorting algorithm that works by repeatedly swapping the adjacent elements if they are in wrong order. You read an list of numbers. You need to arrange the elements in ascending order and print the result. The sorting should be done using bubble sort.

Input Format: The first line reads the number of elements in the array. The second line reads the array elements one by one.

Output Format: The output should be a sorted list.

For example:

Input	Result
6 3 4 8 7 1 2	1 2 3 4 7 8
5 4 5 2 3 1	1 2 3 4 5

Program:

```
n=int(input())
k=[int(x) for x in input().split()]
k.sort()
for i in k:
    print(i,end=' ')
```


	Input	Expected	Got	
✔	6 3 4 8 7 1 2	1 2 3 4 7 8	1 2 3 4 7 8	✔
✔	6 9 18 1 3 4 6	1 3 4 6 9 18	1 3 4 6 9 18	✔
✔	5 4 5 2 3 1	1 2 3 4 5	1 2 3 4 5	✔

Ex. No. : 10.2 **Date: 01.06.24**

Register No.: 230701383 **Name Vinith B**

.

Peak Element

Given an list, find peak element in it. A peak element is an element that is greater than its neighbors.

An element a[i] is a peak element if

A[i-1] <= A[i] >=a[i+1] for middle elements. [0<i<n-1]

A[i-1] <= 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

5

8 9 10 2 6

Sample Output

10 6

For example:

Input	Result
4 12 3 6 8	12 8

Program:

```
a=int(input())
lst1=[str(x) for x in input().split(" ")]
lst2=[]
lst=[]
g=0
for i in lst1:
    if i.isdigit():
        g=int(i)
        lst.append(g)
for i in range(0,a):
    if(i==0):
        if(lst[i]>=lst[i+1]):
            lst2.append(lst[i])
    elif(i>0 and i<a-2):
        if(lst[i]>=lst[i-1] and lst[i]>=lst[i+1]):
            lst2.append(lst[i])
    elif(i==a-1):
        if(lst[i]>=lst[i-1]):
            lst2.append(lst[i])
for i in lst2:
    print(i,end=" ")
```


	Input	Expected	Got	
✔	7 15 7 10 8 9 4 6	15 10 9 6	15 10 9 6	✔
✔	4 12 3 6 8	12 8	12 8	✔

Ex. No. : 10.3 **Date: 01.06.24**

Register No.: 230701383 **Name: Vinith B**

.

Merge Sort

Write a Python program to sort a list of elements using the merge sort algorithm.

For example:

Input	Result
5 6 5 4 3 8	3 4 5 6 8

Program:

```
def merge_sort(arr):
    if len(arr) > 1:
        mid = len(arr) // 2
        left_half = arr[:mid]
        right_half = arr[mid:]
        merge_sort(left_half)
        merge_sort(right_half)
        i = j = k = 0
        while i < len(left_half) and j < len(right_half):
            if left_half[i] < right_half[j]:
                arr[k] = left_half[i]
                i += 1
            else:
```

```
            arr[k] = right_half[j]
            j += 1
        k += 1
    while i < len(left_half):
        arr[k] = left_half[i]
        i += 1
        k += 1
    while j < len(right_half):
        arr[k] = right_half[j]
        j += 1
        k += 1
def main():
    n = int(input())
    arr = list(map(int, input().split()))
    merge_sort(arr)
    for num in arr:
        print(num, end=" ")
if __name__ == "__main__":
    main()
```


	Input	Expected	Got	
✔	5 6 5 4 3 8	3 4 5 6 8	3 4 5 6 8	✔
✔	9 14 46 43 27 57 41 45 21 70	14 21 27 41 43 45 46 57 70	14 21 27 41 43 45 46 57 70	✔
✔	4 86 43 23 49	23 43 49 86	23 43 49 86	✔

Ex. No. : 10.4 **Date: 01.06.24**

Register No.: 230701383 **Name: Vinith B**

.

Sum of Two numbers

An list contains N numbers and you want to determine whether two of the numbers sum to a given number K. For example, if the input is 8, 4, 1, 6 and K is 10, the answer is yes (4 and 6). A number may be used twice.

Input Format

The first line contains a single integer n , the length of list

The second line contains n space-separated integers, list[i].

The third line contains integer k.

Output Format

Print Yes or No.

Sample Input

7

0 1 2 4 6 5 3

1

Sample Output

Yes

For example:

Input	Result
5 8 9 12 15 3 11	Yes
6 2 9 21 32 43 43 1 4	No

Program:

```
n=int(input())
a=[int(x) for x in input().split()]
k=int(input())
flag=0
if len(a)!=n:
    print("No")
    flag=1
for i in a:
    for j in a:
        if i+j==k and flag==0:
            flag=1
            print("Yes")
            break
if flag==0:
    print("No")
```

	Input	Expected	Got	
✔	5 8 9 12 15 3 11	Yes	Yes	✔
✔	6 2 9 21 32 43 43 1 4	No	No	✔
✔	6 13 42 31 4 8 9 17	Yes	Yes	✔

Ex. No. : 10.5 **Date: 01.06.24**

Register No.: 230701383 **Name: Vinith B**

.

Frequency of Elements

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

Constraints:

1<=n, arr[i]<=100

Input:

1 68 79 4 90 68 1 4 5

output:

1 2

4 2

5 1

68 2

79 1

90 1

For example:

Input	Result
4 3 5 3 4 5	3 2 4 2 5 2

Program:

```
lst5=[int(x) for x in input().split(" ")]
lst=sorted(list(set(lst5)))
c=0
for i in lst:
    c=0
    for j in lst5:
        if(i==j):
            c=c+1
    print("%d %d"%(i,c))
```

	Input	Expected	Got	
✔	4 3 5 3 4 5	3 2 4 2 5 2	3 2 4 2 5 2	✔
✔	12 4 4 4 2 3 5	2 1 3 1 4 3 5 1 12 1	2 1 3 1 4 3 5 1 12 1	✔
✔	5 4 5 4 6 5 7 3	3 1 4 2 5 3 6 1 7 1	3 1 4 2 5 3 6 1 7 1	✔