

Ex. No. : 10.1

Date:10/05/2024

Register No.: 231901004

Name: VAKASHDURAI

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

Code:

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

        while i<len(L):
            arr[k]=L[i]
            i+=1
            k+=1
        while j<len(R):
            arr[k]=R[j]
            j+=1
            k+=1
n=int(input())
arr=list(map(int,input().split()))
merge_sort(arr)
print(*arr)
```

Ex. No. : 10.2

Date:10/05/2024

Register No.: 231901004

Name: VAKASHDURAI

Bubble Sort

Given an list of integers, sort the array in ascending order using the *Bubble Sort* algorithm above. Once sorted, print the following three lines:

1. [List](#) is sorted in numSwaps swaps., where numSwaps is the number of swaps that took place.
2. First Element: firstElement, the *first* element in the sorted [list](#).
3. Last Element: lastElement, the *last* element in the sorted [list](#).

For example, given a worst-case but small array to sort: $a=[6,4,1]$. It took 3 swaps to sort the array.

Output would be

Array is sorted in 3 swaps.

First Element: 1

Last Element: 6

Input Format

The first line contains an integer, n , the size of the [list](#) a . The second line contains n , space-separated integers $a[i]$.

Constraints

- $2 \leq n \leq 600$
- $1 \leq a[i] \leq 2 \times 10^5$.

Output Format

You must print the following three lines of output:

1. [List](#) is sorted in numSwaps swaps., where numSwaps is the number of swaps that took place.
2. First Element: firstElement, the *first* element in the sorted [list](#).
3. Last Element: lastElement, the *last* element in the sorted [list](#).

Sample Input 0

3

1 2 3

SampleOutput0

[List](#) is sorted in 0 swaps.

First Element: 1

Last Element: 3

For example:

Input	Result
3 3 2 1	List is sorted in 3 swaps. First Element: 1 Last Element: 3
5 1 9 2 8 4	List is sorted in 4 swaps. First Element: 1 Last Element: 9

Code:

```
def sort(a):
    num=0
    for j in range(len(a)):
        for i in range(len(a)-1):
            if a[i]>a[i+1]:
                a[i],a[i+1]=a[i+1],a[i]
                num+=1
    print(f"List is sorted in {num} swaps.\nFirst Element: {a[0]}\nLast Element: {a[-1]}")
a=int(input())
b=input().split("")
c=[]
for i in range(len(b)):
    c.append(int(b[i]))
sort(c)
```

Ex. No. : 10.3

Date:10/05/2024

Register No.: 231901004

Name: VAKASHDURAI

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

$A[i-1] \leq A[i] \geq A[i+1]$ for middle elements. $[0 < i < n-1]$

$A[i-1] \leq A[i]$ for last element $[i=n-1]$

$A[i] \geq A[i+1]$ 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

Code:

```
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=" ")
```

Ex. No. : 10.4

Date:10/05/2024

Register No.: 231901004

Name: VAKASHDURAI

Binary Search

Write a Python program for binary search.

For example:

Input	Result
1 2 3 5 8 6	False
3 5 9 45 42 42	True

Code:

```
a = input().split(",")
```

```
b = input()
```

```
print(b in a)
```

Ex. No. : 10.5

Date:10/05/2024

Register No.: 231901004

Name: VAKASHDURAI

Frequency of Elements

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

Constraints:

$1 \leq n, \text{arr}[i] \leq 100$

Input:

1 6 8 7 9 4 9 0 6 8 1 4 5

output:

1 2

4 2

5 1

6 8 2

7 9 1

9 0 1

For example:

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

Code:

```
arr = list(map(int, input().split()))
freq_dict = {}
for num in arr:
    freq_dict[num] = freq_dict.get(num, 0) + 1
sorted_freq = sorted(freq_dict.items())
for num, freq in sorted_freq:
    print(num, freq)
```