**10- Searching&Sorting**

**Ex. No. : 10.1 Date: 5/6/24**

**Register No.: 231801171 Name: Stergio Eugin**

## MergeSort

WriteaPythonprogramtosorta listofelementsusingthemergesortalgorithm.

**For example:**

|  |  |
| --- | --- |
| **Input** | **Result** |
| 5  6 5 4 3 8 | 3 4 5 6 8 |

# Program:

a=int(input()) l=[]

l.extend(input().split()) for i in range(a-1):

for j in range(a-1): if(int(l[j])>int(l[j+1])):

t=int(l[j]) l[j]=int(l[j+1]) l[j+1]=t

for i in range(a): print(int(l[i]),end="")

# Output:



**Ex. No. : 10.2 Date: 5/6/24**

**Register No.: 231801171 Name: Stergio Eugin**

## BubbleSort

Givenanlistofintegers,sortthearrayinascendingorderusingthe *Bubble Sort* algorithm above. Once sorted, print the following three lines:

1. [List](http://118.185.187.137/moodle/mod/resource/view.php?id=1068)is sorted in numSwapsswaps., where numSwapsisthe numberof swapsthat took place.
2. FirstElement:firstElement,the*first*elementinthesorted [list](http://118.185.187.137/moodle/mod/resource/view.php?id=1068).
3. LastElement:lastElement,the *last*elementinthesorted [list](http://118.185.187.137/moodle/mod/resource/view.php?id=1068).

Forexample,givenaworst-casebutsmallarraytosort:a=[6,4,1].Ittook3swapstosort the array. Output would be

Arrayissortedin3swaps. First Element: 1

LastElement:6

### InputFormat

The first line contains an integer,n , the size of the[list](http://118.185.187.137/moodle/mod/resource/view.php?id=1068)a. The second line containsn,space-separated integers a[i].

### Constraints

· 2<=n<=600

· 1<=a[i]<=2x106.

### OutputFormat

Youmustprintthefollowingthree linesofoutput:

1. [List](http://118.185.187.137/moodle/mod/resource/view.php?id=1068)issortedinnumSwapsswaps.,where numSwapsisthe numberofswapsthat took place.
2. FirstElement:firstElement,the*first*elementinthesorted[list](http://118.185.187.137/moodle/mod/resource/view.php?id=1068).
3. LastElement:lastElement,the *last*elementinthesorted [list](http://118.185.187.137/moodle/mod/resource/view.php?id=1068).

### SampleInput0

3

123

### SampleOutput0

[List](http://118.185.187.137/moodle/mod/resource/view.php?id=1068)issortedin0swaps. First Element: 1

LastElement:3

**For example:**

|  |  |
| --- | --- |
| **Input** | **Result** |
| 3  32 1 | Listissortedin3swaps. First Element: 1  LastElement:3 |
| 5  1 9 2 8 4 | Listissortedin4swaps. First Element: 1  LastElement:9 |

# Program:

defbubble\_sort(arr): n = len(arr)swaps = 0

foriinrange(n):

forjinrange(0,n-i-1): ifarr[j] >arr[j +1]: # Swap elements

arr[j],arr[j+1]=arr[j+1],arr[j] swaps += 1

returnswaps

#Inputthesizeofthelist n = int(input())

#Inputthelistofintegers

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

#Performbubblesortandcountthenumberofswaps num\_swaps = bubble\_sort(arr)

#Printthenumberofswaps

print("Listissortedin",num\_swaps,"swaps.")

# Print the first element print("FirstElement:",arr[0])

# Print the last element print("LastElement:",arr[-1])

# Output:



**Ex. No. : 10.3 Date: 5/6/24**

**Register No.: 231801171 Name: Stergio Eugin**

## PeakElement

Givenan[list,](http://118.185.187.137/moodle/mod/resource/view.php?id=1068) findpeakelementinit. Apeakelementisan elementthatis greaterthan its neighbors.

Anelementa[i]isapeakelementif

A[i-1]<=A[i]>=a[i+1]formiddleelements.[0<i<n-1] A[i-1] <= A[i] for last element [i=n-1]

A[i]>=A[i+1]forfirstelement[i=0]

### InputFormat

Thefirstline containsasingleintegern,the lengthofA. The second line contains n space-separated integers,A[i].

### OutputFormat

**Print**peaknumbersseparatedby space.

### SampleInput

5

89102 6

### SampleOutput

106

**For example:**

|  |  |
| --- | --- |
| **Input** | **Result** |
| 4  1236 8 | 128 |

# Program:

def find\_peak(arr): peak\_elements=[]

#Checkforthefirstelement if arr[0] >= arr[1]:

peak\_elements.append(arr[0])

#Checkformiddleelements foriinrange(1,len(arr)-1):

ifarr[i-1]<=arr[i]>=arr[i+1]: peak\_elements.append(arr[i])

#Checkforthelastelement if arr[-1] >= arr[-2]:

peak\_elements.append(arr[-1])

returnpeak\_elements

#Inputthelengthofthelist n = int(input())

#Inputthelistofintegers

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

#Findpeakelementsandprinttheresult peak\_elements = find\_peak(arr) print(\*peak\_elements)

# Output:



**Ex. No. : 10.4 Date: 5/6/24**

**Register No.: 231801171 Name: Stergio Eugin**

## BinarySearch

WriteaPythonprogramforbinary search.

**For example:**

|  |  |
| --- | --- |
| **Input** | **Result** |
| 1 2 3 5 8  6 | False |
| 35 945 42  42 | True |

# Program:

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

b = input() print(bina)

# Output:



**Ex. No. : 10.5 Date: 5/6/24**

**Register No.: 231801171 Name: Stergio Eugin**

## FrequencyofElements

Tofindthefrequencyof numbersinalistanddisplayinsorted order.

### Constraints:

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

### Input:

1687949068145

### output:

12

42

51

68 2

79 1

90 1

**For example:**

|  |  |
| --- | --- |
| **Input** | **Result** |
| 4 3 5 3 4 5 | 3 2  4 2  5 2 |

# Program:

defcount\_frequency(arr): frequency = {}

#Countthefrequencyofeachnumberinthelist for num in arr:

frequency[num]=frequency.get(num,0)+1

# Sort the dictionary based on keys sorted\_frequency=sorted(frequency.items())

#Printthefrequencyofeachnumber for num, freq in sorted\_frequency:

print(num,freq)

#Inputthelistofnumbers

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

#Countthefrequencyandprinttheresult count\_frequency(arr)

Output:

