

## **06 - List in Python**



**Ex. No. : 6.1**

**Date: 04.05.24**

**Register No.:230701385**

**Name: S. Vishwak**

## **Element Insertion**

Consider a program to insert an element / item in the sorted array. Complete the logic by filling up required code in editable section. Consider an array of size 10. The eleventh item is the data is to be inserted.

Sample Test Cases

Test Case 1

Input

1

3

4

5

6

7

8

9

10

11

2

Output

ITEM to be inserted:2

After insertion array is:

1

2

3

4

5

6

7

8

9

10

11

Test Case 2

Input

11

22

33



55  
66  
77  
88  
99  
110  
120  
44  
Output  
ITEM to be inserted:44  
After insertion array is:  
11  
22  
33  
44  
55  
66  
77  
88  
99  
110  
120

**Program:**

```
t=10
list1=[]
while(t>0):
    m=int(input())
    list1.append(m)
    t-=1
n=int(input())
print("ITEM to be inserted:",n,sep="")
list1.append(n)
list1.sort()
print("After insertion array is:")
for i in list1:
    print(i,end='\n')
```



	Input	Expected	Got	
✓	1 3 4 5 6 7 8 9 10 11 2	ITEM to be inserted:2 After insertion array is: 1 2 3 4 5 6 7 8 9 10 11	ITEM to be inserted:2 After insertion array is: 1 2 3 4 5 6 7 8 9 10 11	✓
✓	11 22 33 55 66 77 88 99 110 120 44	ITEM to be inserted:44 After insertion array is: 11 22 33 44 55 66 77 88 99 110 120	ITEM to be inserted:44 After insertion array is: 11 22 33 44 55 66 77 88 99 110 120	✓



**Ex. No. : 6.2**

**Date: 04.05.24**

**Register No.:230701385**

**Name: S. Vishwak**

## **Anagram**

Given two lists A and B, and B is an anagram of A. B is an anagram of A means B is made by randomizing the order of the elements in A.

We want to find an *index mapping* P, from A to B. A mapping  $P[i] = j$  means the  $i$ th element in A appears in B at index  $j$ .

These lists A and B may contain duplicates. If there are multiple answers, output any of them.

For example, given

### **Input**

5

12 28 46 32 50

50 12 32 46 28

### **Output**

1 4 3 2 0

### **Explanation**

A = [12, 28, 46, 32, 50]

B = [50, 12, 32, 46, 28]

We should return

[1, 4, 3, 2, 0]

as  $P[0] = 1$  because the 0th element of A appears at B[1], and  $P[1] = 4$  because the 1st element of A appears at B[4], and so on.

### **Note:**

1. A, B have equal lengths in range [1, 100].
2.  $A[i]$ ,  $B[i]$  are integers in range  $[0, 10^5]$ .



**Program:**

```
n=int(input())
```

```
list1=[]
```

```
list2=[]
```

```
list1=input()
```

```
list2=input()
```

```
list1=list1.split()
```

```
list2=list2.split()
```

```
for i in list1:
```

```
    if(i in list2):
```

```
        print(list2.index(i),end=" ")
```

	Input	Expected	Got	
✓	5 12 28 46 32 50 50 12 32 46 28	1 4 3 2 0	1 4 3 2 0	✓



Ex. No. : 6.3

Date: 04.05.24

Register No.: 230701385

Name: S. Vishwak

## **Merge Two Sorted Arrays Without Duplication**

Output is a merged array without duplicates.

Input Format

N1 - no of elements in array 1

Array elements for array 1

N2 - no of elements in array 2

Array elements for array2

Output Format

Display the merged array

### **Sample Input 1**

5

1

2

3

6

9

4

2

4

5

10

### **Sample Output 1**

1 2 3 4 5 6 9 10



**Program:**

```
n1=int(input())
list1=[]
while(n1>0):
    m=int(input())
    list1.append(m)
    n1-=1
n2=int(input())
list2=[]
while(n2>0):
    m=int(input())
    list2.append(m)
    n2-=1
for i in list2:
    if(i not in list1):
        list1.append(i)
    else:
        continue
list1.sort()
for i in list1:
    print(i,end=" ")
```





	Input	Expected	Got	
✓	5 1 2 3 6 9 4 2 4 5 10	1 2 3 4 5 6 9 10	1 2 3 4 5 6 9 10	✓
✓	7 4 7 8 10 12 30 35 9 1 3 4 5 7 8 11 13 22	1 3 4 5 7 8 10 11 12 13 22 30 35	1 3 4 5 7 8 10 11 12 13 22 30 35	✓



Ex. No. : 6.4

Date: 04.05.24

Register No.: 230701385

Name: S. Vishwak

### **Distinct Elements in an Array**

Program to print all the distinct elements in an array. Distinct elements are nothing but the unique (non-duplicate) elements present in the given array.

Input Format:

First line take an Integer input from stdin which is array length n.

Second line take n Integers which is inputs of array.

Output Format:

Print the Distinct Elements in Array in single line which is space Separated

Example Input:

5

1

2

2

3

4

Output:

1 2 3 4

Example Input:

6

1

1

2

2

3

3

Output:

1 2 3



**For example:**

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

**Program:**

```
n=int(input())
```

```
list1=[]
```

```
while(n>0):
```

```
    m=int(input())
```

```
    list1.append(m)
```

```
    n-=1
```

```
list2=[]
```

```
for i in list1:
```

```
    if i in list2:
```



```
continue
```

```
else:
```

```
list2.append(i)
```

```
for i in list2:
```

```
print(i,end=" ")
```

	Input	Expected	Got	
✓	5 1 2 2 3 4	1 2 3 4	1 2 3 4	✓
✓	6 1 1 2 2 3 3	1 2 3	1 2 3	✓

**Ex. No. : 6.5**

**Date: 04.05.24**

**Register No.: 230701385**

**Name: S. Vishwak**

### **The Pivot**

Given an array of numbers, find the index of the smallest array element (the pivot), for which the sums of all elements to the left and to the right are equal. The array may not be reordered.

Example

arr=[1,2,3,4,6]

- the sum of the first three elements,  $1+2+3=6$ . The value of the last element is 6.
- Using zero based indexing, arr[3]=4 is the pivot between the two subarrays.
- The index of the pivot is 3.

Constraints

- $3 \leq n \leq 10^5$
- $1 \leq \text{arr}[i] \leq 2 \times 10^4$ , where  $0 \leq i < n$
- It is guaranteed that a solution always exists.

The first line contains an integer n, the size of the array arr.

Each of the next n lines contains an integer, arr[i], where  $0 \leq i < n$ .

Sample Case 0

Sample Input 0

4

1

2

3

3

Sample Output 0

2

Explanation 0

- The sum of the first two elements,  $1+2=3$ . The value of the last element is 3.



- Using zero based indexing, arr[2]=3 is the pivot between the two subarrays.
- The index of the pivot is 2.

Sample Case 1

Sample Input 1

3

1

2

1

Sample Output 1

1

Explanation 1

- The first and last elements are equal to 1.
- Using zero based indexing, arr[1]=2 is the pivot between the two subarrays.
- The index of the pivot is 1.

**For example:**

Input	Result
4	2
1	
2	
3	
3	
3	1
1	
2	
1	



**Program:**

```
n=int(input())
list1=[]
while(n>0):
    m=int(input())
    list1.append(m)
    n-=1
tot_sum=sum(list1)
l_sum=0
for i in range(len(list1)):
    r_sum=tot_sum-l_sum-list1[i]
    if l_sum==r_sum:
        print(i)
    l_sum+=list1[i]
```

	Input	Expected	Got	
✓	4 1 2 3 3	2	2	✓
✓	3 1 2 1	1	1	✓



**Ex. No. : 6.6**

**Date: 04.05.24**

**Register No.: 230701385**

**Name: S. Vishwak**

Determine the factors of a number (i.e., all positive integer values that evenly divide into a number) and then return the  $p^{\text{th}}$  element of the list, sorted ascending. If there is no  $p^{\text{th}}$  element, return 0.

**Example**

$n = 20$

$p = 3$

The factors of 20 in ascending order are {1, 2, 4, 5, 10, 20}. Using 1-based indexing, if  $p = 3$ , then 4 is returned. If  $p > 6$ , 0 would be returned.

**Constraints**

$1 \leq n \leq 10^{15}$

$1 \leq p \leq 10^9$

The first line contains an integer  $n$ , the number to factor.

The second line contains an integer  $p$ , the 1-based index of the factor to return.

**Sample Case 0**

**Sample Input 0**

10

3

**Sample Output 0**

5

**Explanation 0**

Factoring  $n = 10$  results in {1, 2, 5, 10}. Return the  $p = 3^{\text{rd}}$  factor, 5, as the answer.

**Sample Case 1**

**Sample Input 1**

10

5

**Sample Output 1**





0

### Explanation 1

Factoring  $n = 10$  results in  $\{1, 2, 5, 10\}$ . There are only 4 factors and  $p = 5$ , therefore 0 is returned as the answer.

### Sample Case 2

#### Sample Input 2

1

1

#### Sample Output 2

1

### Explanation 2

Factoring  $n = 1$  results in  $\{1\}$ . The  $p = 1$ st factor of 1 is returned as the answer.

### For example:

Input	Result
10 3	5
10 5	0
1 1	1

### Program:

```
n=int(input())
```

```
list1=[]
```

```
for i in range(1,n+1):
```

```
    if(n%i==0):
```



```
        list1.append(i)
list1.sort()
p=int(input())
if p<len(list1)+1:
    print(list1[p-1])
else:
    print(0)
```



**Ex. No. : 6.7**

**Date: 04.05.24**

**Register No.:230701385**

**Name: S. Vishwak**

## **Location**

Write a program to print all the locations at which a particular element (taken as input) is found in a list and also print the total number of times it occurs in the list. The location starts from 1.

For example, if there are 4 elements in the array:

5  
6  
5  
7

If the element to search is 5 then the output will be:

5 is present at location 1  
5 is present at location 3  
5 is present 2 times in the array.

### **Sample Test Cases**

#### **Test Case 1**

##### **Input**

4  
5  
6  
5  
7  
5

##### **Output**

5 is present at location 1.  
5 is present at location 3.  
5 is present 2 times in the array.



## Test Case 2

### Input

5  
67  
80  
45  
97  
100  
50

### Output

50 is not present in the array.

### Program:

```
t=int(input())
list1=[]
while(t>0):
    m=int(input())
    list1.append(m)
    t-=1
n=int(input())
l1=enumerate(list1)
if(n in list1):
    for i,j in l1:
        if(j==n):
            print(j," is present at location ",i+1,".",sep=")
            print(n,"is present",list1.count(n),"times in the array.")
else:
    print(n,"is not present in the array.")
```



	Input	Expected	Got	
✓	4 5 6 5 7 5	5 is present at location 1. 5 is present at location 3. 5 is present 2 times in the array.	5 is present at location 1. 5 is present at location 3. 5 is present 2 times in the array.	✓
✓	5 67 80 45 97 100 50	50 is not present in the array.	50 is not present in the array.	✓

**Ex. No. : 6.8**

**Date: 04.05.24**

**Register No.: 230701385**

**Name: S. Vishwak**

### **Strictly increasing**

Write a Python program to check if a given list is strictly increasing or not. Moreover, If removing only one element from the list results in a strictly increasing list, we still consider the list true

Input:

n : Number of elements

List1: List of values

Output

Print "True" if list is strictly increasing or decreasing else print "False"

Sample Test Case

Input

7

1

2

3

0

4

5

6

Output

True



**Program:**

```
n=int(input())
list1=[]
while(n>0):
    m=int(input())
    list1.append(m)
    n-=1
def st_inc(lst):
    for i in range(0,len(lst)-1):
        if lst[i]>=lst[i+1]:
            return False
    return True
def st_dec(lst):
    for i in range(0,len(lst)-1):
        if lst[i]<=lst[i+1]:
            return False
    return True
def rem_elemt(lst):
    new_list=[]
    for i in range(0,len(lst)-1):
        new_list=lst[:i]+lst[i+1:]
        if st_inc(new_list) or st_dec(new_list):
            return True
    return False
print(rem_elemt(list1))
```



	Input	Expected	Got	
✓	7 1 2 3 0 4 5 6	True	True	✓
✓	4 2 1 0 -1	True	True	✓





**Ex. No. : 6.9**

**Date: 04.05.24**

**Register No.: 230701385**

**Name: S. Vishwak**

## **Merge List**

Write a Python program to Zip two given lists of lists.

Input:

m : row size

n: column size

list1 and list 2 : Two lists

Output

Ziped List : List which combined both list1 and list2

Sample test case

Sample input

2  
2  
1  
3  
5  
7  
2  
4  
6  
8

Sample Output

[[1, 3, 2, 4], [5, 7, 6, 8]]



**Program:**

```
r=int(input())
c=int(input())
list1=[]
n=r*c
while(n>0):
    m=int(input())
    list1.append(m)
    n-=1
list2=[]
n=r*c
while(n>0):
    m=int(input())
    list2.append(m)
    n-=1
list3=[]
list4=[]
list5=[]
n=r*c
for i in range(0,n):
    if(i<r):
        list3.append(list1[i])
    else:
        list4.append(list1[i])
for i in range(0,n):
    if(i<r):
        list3.append(list2[i])
    else:
```



```

list4.append(list2[i])
print('[',end=")
print(list3,list4,sep=' ',end=")
print(']')

```

	Input	Expected	Got	
✓	2	[[1, 2, 5, 6], [3, 4, 7, 8]]	[[1, 2, 5, 6], [3, 4, 7, 8]]	✓
	2			
	1			
	2			
	3			
	4			
	5			
	6			
	7			
	8			

**Ex. No. : 6.10**

**Date: 04.05.24**

**Register No.: 230701385**

**Name S. Vishwak**

Complete the program to count frequency of each element of an array. Frequency of a particular element will be printed once.

Sample Test Cases

Test Case 1

Input

7  
23  
45  
23  
56  
45  
23  
40

Output

23 occurs 3 times  
45 occurs 2 times  
56 occurs 1 times  
40 occurs 1 times

**Program:**

```
n=int(input())  
list1=[]  
list2=[]  
while(n>0):
```



```
m=int(input())
list1.append(m)
n-=1
for i in list1:
    c=list1.count(i)
    if(i not in list2 ):
        list2.append(i)
    else:
        continue
print(i,"occurs",c,"times",sep=" ")
```

