

Ex. No. : **5.10**

Date:

Register No: **231501049**

Name: **GNAANESH B B**

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())
```

```
arr = [int(input()) for i in range(n)]
```

```
l = arr.copy()
g=0
size = len(arr)
arr_asc = sorted(arr)
arr_des = sorted(arr)[::-1]
if arr==arr_asc or arr==arr_des:
    print('True')
    g=1
else:
    for i in arr:
        l.remove(i)
        arr_asc.remove(i)
        arr_des.remove(i)
        if l==arr_asc or l==arr_des:
            print("True")
            g=1
        break
    l=arr.copy()
    arr_asc = sorted(arr)
    arr_des = sorted(arr)[::-1]
if g==0:
    print('False')
```

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

Passed all tests! ✓

06 - Strings in Python

Ex. No. : **6.1**

Date:

Register No: **231501049**

Name: **GNAANESH B B**

Count Chars

Write a python program to count all letters, digits, and special symbols respectively from a given string

For example:

Input	Result
rec@123	
3	
3	
1	

PROGRAM:

```
a=input()
c,d,s=0,0,0
for i in range(len(a)):
    if(a[i].isalpha()):
        c+=1
    elif(a[i].isdigit()):
        d+=1
    else:
        s+=1
print(c,d,s,sep="\n")
```

	Input	Expected	Got	
✓	rec@123	3 3 1	3 3 1	✓
✓	P@#yn26at^&i5ve	8 3 4	8 3 4	✓
✓	abc@12&	3 2 2	3 2 2	✓

Passed all tests! ✓

Ex. No. : **6.2**

Date:

Register No: **231501049**

Name: **GNAANESH B B**

Decompress the String

Assume that the given string has enough memory. Don't use any extra space(IN-PLACE)

Sample Input 1
a2b4c6

Sample Output 1
aabbbbcccccc

PROGRAM:

```
import re
a=input()
all=re.findall('\d+',a)
all_w=re.findall('[a-z]',a)
b=""
for i,j in zip(all,all_w):
    b+=int(i)*j
print(b)
```

	Input	Expected	Got	
✓	a2b4c6	aabbccccc	aabbccccc	✓
✓	a12b3d4	aaaaaaaaaaabbdd	aaaaaaaaaaabbdd	✓

Passed all tests! ✓

Ex. No. : **6.3**

Date:

Register No: **231501049**

Name: **GNAANESH B B**

First N Common Chars

Two string values S1, S2 are passed as the input. The program must print first N characters present in S1 which are also present in S2.

Input Format:

The first line contains S1.
The second line contains S2.
The third line contains N.

Output Format:

The first line contains the N characters present in S1 which are also present in S2.

Boundary Conditions:

$2 \leq N \leq 10$
 $2 \leq \text{Length of S1, S2} \leq 1000$

Example Input/Output 1:

Input:

abcbde
cdefghbb
3

Output:

bcd

Note:

b occurs twice in common but must be printed only once.

PROGRAM:

```
a=input()
b=input()
C=""
d=int(input())
for i in range(len(a)):
    if(len(C)-d==0):
        break
    else:
        if(a[i]in b):
            if(a[i] not in C):
                C+=a[i]
print (C)
```

	Input	Expected	Got	
✓	abcde cdefghbb 3	bcd	bcd	✓

Passed all tests! ✓

Ex. No. : **6.4**

Date:

Register No: **231501049**

Name: **GNAANESH B B**

Remove Characters

Given two Strings s1 and s2, remove all the characters from s1 which is present in s2.

Constraints

1<= string length <= 200

Sample Input 1
experience
enc

Sample Output 1
xpri

PROGRAM:

```
def remove_chars(s1, s2):
    return ''.join([char for char in s1 if char not in s2])
s1=input()
s2=input()
result = remove_chars(s1, s2)
print(result)
```

	Input	Expected	Got	
✓	experience enc	xpri	xpri	✓

Passed all tests! ✓