

# Rajalakshmi Engineering College

Name: SHIVANISREE K B  
Email: 240701501@rajalakshmi.edu.in  
Roll no: 240701501  
Phone: 7358464804  
Branch: REC  
Department: I CSE FE  
Batch: 2028  
Degree: B.E - CSE

Scan to verify results



## NeoColab\_REC\_CS23221\_Python Programming

### REC\_Python\_Week 3\_CY

Attempt : 1  
Total Mark : 30  
Marks Obtained : 30

### Section 1 : Coding

#### 1. Problem Statement

Gina is working on a data analysis task where she needs to extract sublists from a given list of integers and find the median of each sublist. For each median found, she also needs to determine its negative index in the original list.

Help Gina by writing a program that performs these tasks.

Note: The median is the middle value in the sorted list of numbers, or the first value of the two middle values if the list has an even number of elements.

Example

Input

10

1 2 3 4 5 7 8 9 10 11

3

1 5

2 6

3 10

Output

3 : -8

4 : -7

7 : -5

Explanation

For the first range (1 to 5), the sublist is [1, 2, 3, 4, 5]. The median is 3, and its negative index in the original list is -8.

For the second range (2 to 6), the sublist is [2, 3, 4, 5, 7]. The median is 4, and its negative index in the original list is -7.

For the third range (3 to 10), the sublist is [3, 4, 5, 7, 8, 9, 10, 11]. The median is 7, and its negative index in the original list is -5.

**Input Format**

The first line of input consists of an integer N, representing the number of elements in the list.

The second line consists of N space-separated integers representing the elements of the list.

The third line consists of an integer R, representing the number of ranges.

The next R lines each consist of two integers separated by space representing the start and end indices (1-based) of the ranges.

**Output Format**

The output consists of n lines, displaying "X : Y" where X is the median of the

sublist and Y is the negative index in the original list.

Refer to the sample output for the formatting specifications.

### **Sample Test Case**

Input: 10

1 2 3 4 5 7 8 9 10 11

3

1 5

2 6

3 10

Output: 3 : -8

4 : -7

7 : -5

### **Answer**

# You are using Python

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

```
og=[]
```

```
sublist=[]
```

```
# for i in range(n):
```

```
#     x=int(input())
```

```
#     og.append(x)
```

```
og=list(map(int,input().split()))
```

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

```
for i in range(m):
```

```
    start,end=map(int,input().split())
```

```
    sublist.append((start,end))
```

```
for start,end in sublist:
```

```
    sub=og[start-1:end]
```

```
    sub_sorted=sorted(sub)
```

```
    n=len(sub_sorted)
```

```
    if n%2==0:
```

```
        median = sub_sorted[n//2 -1]
```

```
    else:
```

```
        median = sub_sorted[n//2]
```

```
neg = -1 -og[::-1].index(median)
print(median,"",neg)
```

**Status :** Correct

**Marks :** 10/10

## 2. Problem Statement

Emily is a data analyst working for a company that collects feedback from customers in the form of text messages. As part of her data validation tasks, Emily needs to perform two operations on each message:

Calculate the sum of all the digits mentioned in the message. If the sum of the digits is greater than 9, check whether the sum forms a palindrome number.

Your task is to help Emily automate this process by writing a program that extracts all digits from a given message, calculates their sum, and checks if the sum is a palindrome if it is greater than 9.

### **Input Format**

The input consists of a string *s*, representing the customer message, which may contain letters, digits, spaces, and other characters.

### **Output Format**

The output prints an integer representing the sum of all digits in the string, followed by a space.

If the sum is greater than 9, print "Palindrome" if the sum is a palindrome, otherwise print "Not palindrome".

If the sum is less than or equal to 9, no palindrome check is required.

Refer to the sample output for the formatting specifications.

### Sample Test Case

Input: 12 books 4 pen

Output: 7

### Answer

```
# You are using Python
```

```
a=str(input())
```

```
sum = 0
```

```
for i in a:
```

```
    if '0'<= i <= '9':
```

```
        sum+=int(i)
```

```
# m=sum
```

```
# rev=0
```

```
# if sum>9:
```

```
    # while(sum!=0):
```

```
    #     sum=sum%10
```

```
    #     rev = (rev*10) + sum
```

```
    #     sum= sum/10
```

```
    # if(m == rev):
```

```
    #     print(m,"Palindrome")
```

```
    # else:
```

```
    #     print(m,"Not palindrome")
```

```
# else:
```

```
#     print(m)
```

```
if sum >9:
```

```
    if str(sum) == str(sum)[::-1]:
```

```
        print(sum,"Palindrome")
```

```
    else:
```

```
        print(sum,"Not palindrome")
```

```
else:
```

```
    print(sum)
```

**Status :** Correct

**Marks :** 10/10

### 3. Problem Statement

You have two strings str1 and str2, both of equal length.

Write a Python program to concatenate the two strings such that the first character of str1 is followed by the first character of str2, the second character of str1 is followed by the second character of str2, and so on.

For example, if str1 is "abc" and str2 is "def", the output should be "adbecf".

#### ***Input Format***

The input consists of two strings in each line.

#### ***Output Format***

The output displays the concatenated string in the mentioned format.

Refer to the sample output for formatting specifications.

#### ***Sample Test Case***

Input: abc

def

Output: adbecf

#### ***Answer***

```
# You are using Python
```

```
a=str(input())
```

```
b=str(input())
```

```
result=""
```

```
for i in range(len(a)):
```

```
    result+=a[i]+b[i]
```

```
print(result)
```

**Status :** Correct

**Marks :** 10/10