Rajalakshmi Engineering College

Name: n.madhu narayanan
Email: 240701295@rajalakshmi.edu.in
Roll no: 240701295
Phone: 8870065218
Branch: REC
Department: I CSE AH
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

Sarah is a technical writer who is responsible for formatting two important documents. Both documents contain a certain placeholder character that needs to be replaced with another character before they can be finalized. To ensure consistency in formatting, Sarah wants you to help her write a program that processes both documents by replacing the placeholder character with the new one.

Sarah also prefers a neat and structured output, so she wants you to ensure that both modified documents are printed in a single line, separated by a space, using the format() function.

Example

Input:

Hello

World

o

a

Output:

Hella Warld

Explanation:

Here the character 'o' is replaced with 'a' in the concatenated string.

Input Format

The first line contains string1, the first document.

The second line contains string2, the second document.

The third line contains char1, the placeholder character that needs to be replaced.

The fourth line contains char2, the new character that will replace the placeholder.

Output Format

The output displays a single line containing the modified string1 and string2, separated by a space.

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: Hello
World
o
a
Output: Hella Warld

Answer

```
# You are using Python
# Function to replace placeholder characters in the documents
def replace_placeholder(string1, string2, char1, char2):
    # Replace char1 with char2 in both strings
    modified_string1 = string1.replace(char1, char2)
    modified_string2 = string2.replace(char1, char2)

    # Print the modified strings in the required format
    print("{} {}".format(modified_string1, modified_string2))

# Input reading
string1 = input().strip()
string2 = input().strip()
char1 = input().strip()
char2 = input().strip()

# Call the function with the provided inputs
replace_placeholder(string1, string2, char1, char2)
```

Status : Correct **Marks : 10/10**

2. 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

```
def concatenate_strings(str1, str2):
    result = []
    for char1, char2 in zip(str1, str2):
        result.append(char1)
        result.append(char2)
    return ''.join(result)
str1 = input().strip()
str2 = input().strip()
output = concatenate_strings(str1, str2)
print(output)
```

Status : Correct ***Marks : 10/10***

3. 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

```
def find_median_and_negative_index(n, arr, queries):
    for start, end in queries:
        sublist = arr[start - 1:end]
        sublist_sorted = sorted(sublist)
        length = len(sublist_sorted)
        if length % 2 == 1:
            median = sublist_sorted[length // 2]
        else:
            median = sublist_sorted[length // 2 - 1]
        index = arr.index(median)
        negative_index = index - n
        print(f"{median} : {negative_index}")
n = int(input())
arr = list(map(int, input().split()))
q = int(input())
queries = [tuple(map(int, input().split())) for _ in range(q)]
find_median_and_negative_index(n, arr, queries)
```

Status : Correct ***Marks : 10/10***