Rajalakshmi Engineering College

Name: Balaji K
Email: 241501031@rajalakshmi.edu.in
Roll no: 241501031
Phone: 7339164378
Branch: REC
Department: I AIML AD
Batch: 2028
Degree: B.E - AI & ML

Scan to verify results

NeoColab_REC_CS23221_Python Programming

REC_Python_Week 5_CY

Attempt : 1
Total Mark : 40
Marks Obtained : 39

Section 1 : Coding

1. Problem Statement

Alex is tasked with managing the membership lists of several exclusive clubs. Each club has its own list of members, and Alex needs to determine the unique members who are part of exactly one club when considering all clubs together.

Your goal is to help Alex by writing a program that calculates the symmetric difference of membership lists from multiple clubs and then finds the total number of unique members.

Input Format

The first line of input consists of an integer k, representing the number of clubs.

The next k lines each contain a space-separated list of integers, where each

integer represents a member's ID.

Output Format

The first line of output displays the symmetric difference of the membership lists as a set.

The second line displays the sum of the elements in this symmetric difference.

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: 3
1 2 3
2 3 4
5 6 7

Output: {1, 4, 5, 6, 7}
23

Answer

```
k = int(input())
sets = []
for _ in range(k):
    elements = list(map(int, input().split()))
    sets.append(set(elements))
result = sets[0]
for i in range(1, k):
    result = result.symmetric_difference(sets[i])
print(result)
print(sum(result))
```

Status : Partially correct ***Marks : 9/10***

2. Problem Statement

Alex is working with grayscale pixel intensities from an old photo that has been scanned in a single row. To detect edges in the image, Alex needs to calculate the differences between each pair of consecutive pixel

intensities.

Your task is to write a program that performs this calculation and returns the result as a tuple of differences.

Input Format

The first line of input contains an integer n, representing the number of pixel intensities.

The second line contains n space-separated integers representing the pixel intensities.

Output Format

The output displays a tuple containing the absolute differences between consecutive pixel intensities.

Refer to the sample output for format specifications.

Sample Test Case

Input: 5
200 100 20 80 10
Output: (100, 80, 60, 70)

Answer

```
n = int(input())
pixels = list(map(int, input().split()))
differences = []
for i in range(n - 1):
    diff = abs(pixels[i+1] - pixels[i])
    differences.append(diff)
print(tuple(differences))
```

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

3. Problem Statement

Riley is analyzing DNA sequences and needs to determine which bases

match at the same positions in two given DNA sequences. Each DNA sequence is represented as a tuple of integers, where each integer corresponds to a DNA base.

Your task is to write a program that compares these two sequences and identifies the bases that match at the same positions and print it.

Input Format

The first line of input consists of an integer n, representing the size of the first tuple.

The second line contains n space-separated integers, representing the elements of the first DNA sequence tuple.

The third line of input consists of an integer m, representing the size of the second tuple.

The fourth line contains m space-separated integers, representing the elements of the second DNA sequence tuple.

Output Format

The output is a space-separated integer of the matching bases at the same positions in both sequences.

Refer to the sample output for format specifications.

Sample Test Case

Input: 4
5 1 8 4
4
4 1 8 2

Output: 1 8

Answer

```
n = int(input())
seq1 = tuple(map(int, input().split()))
m = int(input())
seq2 = tuple(map(int, input().split()))
```

```
length = min(n, m)
matching_bases = []
for i in range(length):
    if seq1[i] == seq2[i]:
        matching_bases.append(seq1[i])
print(*matching_bases)
```

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

4. Problem Statement

Noah, a global analyst at a demographic research firm, has been tasked with identifying which country experienced the largest population growth over a two-year period. He has a dataset where each entry consists of a country code and its population figures for two consecutive years. Noah needs to determine which country had the highest increase in population and present the result in a specific format.

Help Noah by writing a program that outputs the country code with the largest population increase, along with the increase itself.

Input Format

The first line of input consists of an integer N, representing the number of countries.

Each of the following N blocks contains three lines:

1. The first line is a country code.
2. The second line is an integer representing the population of the country in the first year.
3. The third line is an integer representing the population of the country in the second year.

Output Format

The output displays the country code and the population increase in the format {code: difference}, where code is the country code and difference is the increase in population.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 3
01
1000
1500
02
2000
2430
03
1500
3000
Output: {03:1500}

Answer

```
N = int(input())
population_diff = {}
for _ in range(N):
    country_code = input().strip()
    pop_year1 = int(input())
    pop_year2 = int(input())
    diff = pop_year2 - pop_year1
    population_diff[country_code] = diff
max_country = max(population_diff, key=population_diff.get)
max_diff = population_diff[max_country]
print("{" + max_country + ":" + str(max_diff) + "}")
```

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