Rajalakshmi Engineering College

Name: Sweta Suresh
Email: 240701549@rajalakshmi.edu.in
Roll no: 2116240701549
Phone: 9344119216
Branch: REC
Department: I CSE FF
Batch: 2028
Degree: B.E - CSE

Scan to verify results

NeoColab_REC_CS23221_Python Programming

REC_Python_Week 5_CY

Attempt : 1
Total Mark : 40
Marks Obtained : 16.5

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

```
from collections import Counter

n = int(input())
rows = []
num_in_rows = Counter()

for _ in range(n):
    row = list(map(int, input().split()))
    rows.append(row)
    for num in set(row):  # Avoid counting duplicates in the same row
        num_in_rows[num] += 1

# Get numbers that appear in only one row
result_set = {num for num, count in num_in_rows.items() if count == 1}

# Sum of those numbers only
total_sum = sum(result_set)

print(result_set)
print(total_sum)
```

Status : Partially correct ***Marks : 6.5/10***

2. 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

```
from collections import defaultdict, OrderedDict

n = int(input())
records = []
number_to_ids = defaultdict(list)

# Read records
for _ in range(n):
    id_ = input().strip()
    num1 = int(input())
    num2 = int(input())
    records.append((id_, [num1, num2]))
    number_to_ids[num1].append(id_)
    number_to_ids[num2].append(id_)

# Track which number is duplicated
duplicated_numbers = {num for num, ids in number_to_ids.items() if len(ids) > 1}

# Track result using OrderedDict to preserve input order
result = OrderedDict()

# Only include an ID if:
# - It has exactly one shared number (i.e., appears in more than one ID)
# - And that ID is not the first occurrence of that number
for id_, nums in records:
    shared = [num for num in nums if num in duplicated_numbers]
    if len(shared) == 1:
        num = shared[0]
        if number_to_ids[num][-1] == id_:  # only include the last one to have it
            result[id_] = num
```

```
# Print exactly like {02:1200}
print("{", end="")
print(", ".join([f"{k}:{v}" for k, v in result.items()]), end="")
print("}")
```

Status : Wrong ***Marks : 0/10***

3. Problem Statement

James is an engineer working on designing a new rocket propulsion system. He needs to solve a quadratic equation to determine the optimal launch trajectory. The equation is of the form ax2 +bx+c=0.

Your task is to help James find the roots of this quadratic equation. Depending on the discriminant, the roots might be real and distinct, real and equal, or complex. Implement a program to determine and display the roots of the equation based on the given coefficients.

Input Format

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

The second line contains three space-separated integers a,b, and c representing the coefficients of the quadratic equation.

Output Format

The output displays:

1. If the discriminant is positive, display the two real roots.
2. If the discriminant is zero, display the repeated real root.
3. If the discriminant is negative, display the complex roots as a tuple with real and imaginary parts.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 3
1 5 6
Output: (-2.0, -3.0)

Answer

```
import cmath

# Input number of coefficients
N = int(input())

# Input the coefficients a, b, c
if N == 3:
    a, b, c = map(int, input().split())

    # Calculate the discriminant
    D = b**2 - 4*a*c

    # Calculate the two roots using quadratic formula
    root1 = (-b + cmath.sqrt(D)) / (2*a)
    root2 = (-b - cmath.sqrt(D)) / (2*a)

    # Output formatting
    if D > 0:
        # Real and distinct roots
        roots = tuple(sorted([root1.real, root2.real], reverse=True))
        print(f"({roots[0]}, {roots[1]})")
    elif D == 0:
        # Repeated real root
        print(f"({root1.real}, {root2.real})")
    else:
        # Complex roots
        print(f"(({root1.real}, {root1.imag}), ({root2.real}, {root2.imag}))")
else:
    print("Invalid number of coefficients.")
```

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

4. Problem Statement

Samantha is working on a text analysis tool that compares two words to find common and unique letters. She wants a program that reads two words, w1, and w2, and performs the following operations:

Print the letters common to both words, in alphabetical order.Print the letters that are unique to each word, in alphabetical order.Determine if the set of letters in the first word is a superset of the letters in the second word.Check if there are no common letters between the two words and print the result as a Boolean value.

Ensure the program ignores case differences and leading/trailing spaces in the input words.

Your task is to help Samantha in implementing the same.

Input Format

The first line of input consists of a string representing the first word, w1.

The second line consists of a string representing the second word, w2.

Output Format

The first line of output should display the sorted letters common to both words, printed as a list.

The second line should display the sorted letters that are unique to each word, printed as a list.

The third line should display a Boolean value indicating if the set of letters in w1 is a superset of the set of letters in w2.

The fourth line should display a Boolean value indicating if there are no common letters between w1 and w2.

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: program
Peace
Output: ['a', 'p']
['c', 'e', 'g', 'm', 'o', 'r']
False
False

Answer

Q2
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:

The first line is a country code.
The second line is an integer representing the population of the country in the first year.
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.
Code constraints :

1 ≤ N ≤ 15

01 ≤ country code ≤ 09

1 ≤ population ≤ 10000
Sample test cases :
Input 1 :

3
01
1000
1500
02
2000
2430
03
1500
3000

Output 1 :

{03:1500}

Input 2 :

4
01
5000
5500
02
6000
7200
03
5500
6100
04
4500
5000

Output 2 :

{02:1200

Q4
Samantha is working on a text analysis tool that compares two words to find common and unique letters. She wants a program that reads two words, w1, and w2, and performs the following operations:

Print the letters common to both words, in alphabetical order.
Print the letters that are unique to each word, in alphabetical order.
Determine if the set of letters in the first word is a superset of the letters in the second word.
Check if there are no common letters between the two words and print the result as a Boolean value.

Ensure the program ignores case differences and leading/trailing spaces in the input words.

Your task is to help Samantha in implementing the same.
Input format :

The first line of input consists of a string representing the first word, w1.

The second line consists of a string representing the second word, w2.
Output format :

The first line of output should display the sorted letters common to both words, printed as a list.

The second line should display the sorted letters that are unique to each word, printed as a list.

The third line should display a Boolean value indicating if the set of letters in w1 is a superset of the set of letters in w2.

The fourth line should display a Boolean value indicating if there are no common letters between w1 and w2.

Refer to the sample output for the formatting specifications.
Code constraints :

In this scenario, the given test cases will fall under the following constraints:

Both w1 and w2 will only contain alphabetic characters and may include leading or trailing spaces.

1 ≤ Lengths of w1 and w2 ≤ 50
Sample test cases :
Input 1 :

program
Peace

Output 1 :

['a', 'p']
['c', 'e', 'g', 'm', 'o', 'r']
False
False

Input 2 :

Kindness
kind

Output 2 :

['d', 'i', 'k', 'n']
['e', 's']
True
False

Status : Wrong ***Marks : 0/10***