

# Rajalakshmi Engineering College

Name: Aldrine Linjoe.s  
Email: 240701033@rajalakshmi.edu.in  
Roll no: 240701033  
Phone: 7092049029  
Branch: REC  
Department: I CSE FA  
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 : 39

### Section 1 : Coding

#### 1. 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**

# You are using Python

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

```
v={}
```

```
k=0
```

```
m=0
```

```
l={}
```

```
for i in range(0,n):
```

```
    b=input().strip()
```

```
    f=int(input())
```

```
t=int(input())
v[b]=t-f
for i,j in v.items():
    if(j>m):
        m=j
        k=i

print(f"{{{k}:{m}}}")
```

**Status :** Correct

**Marks :** 10/10

## 2. Problem Statement

Emily is a librarian who keeps track of books borrowed and returned by her patrons. She maintains four sets of book IDs: the first set represents books borrowed, the second set represents books returned, the third set represents books added to the collection, and the fourth set represents books that are now missing. Emily wants to determine which books are still borrowed but not returned, as well as those that were added but are now missing. Finally, she needs to find all unique book IDs from both results.

Help Emily by writing a program that performs the following operations on four sets of integers:

Compute the difference between the borrowed books (first set) and the returned books (second set). Compute the difference between the added books (third set) and the missing books (fourth set). Find the union of the results from the previous two steps, and sort the final result in descending order.

### ***Input Format***

The first line of input consists of a list of integers representing borrowed books.

The second line of input consists of a list of integers representing returned books.

The third line of input consists of a list of integers representing added books.

The fourth line of input consists of a list of integers representing missing books.

### **Output Format**

The first line of output displays the difference between sets P and Q, sorted in descending order.

The second line of output displays the difference between sets R and S, sorted in descending order.

The third line of output displays the union of the differences from the previous two steps, sorted in descending order.

Refer to the sample output for the formatting specifications.

### **Sample Test Case**

Input: 1 2 3

2 3 4

5 6 7

6 7 8

Output: [1]

[5]

[5, 1]

### **Answer**

```
# You are using Python
n=set(map(int,input().split()))
t=set(map(int,input().split()))
r=set(map(int,input().split()))
s=set(map(int,input().split()))
f=n-t
o=r-s
g=f^o
h=sorted(g,reverse=True)
print(list(f))
print(list(o))
print(list(h))
```

**Status :** Correct

**Marks :** 10/10

### 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  $ax^2 + bx + c = 0$ .

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**

# You are using Python

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

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

```
a=v[0]
```

```
b=v[1]
```

```
c=v[2]
```

```
d=(b**2)-4*a*c
```

```
m=(-b+(d**(1/2)))/(2*a)
```

```
u=(-b-(d**(1/2)))/(2*a)
```

```
if(d==0):
```

```
    print((m.real,u.real))
```

```
elif(d>0):
```

```
    print((m.real,u.real))
```

```
else:
```

```
    print(((round(m.real,1),round(m.imag,1)),(round(u.real,1),round(u.imag,1))))
```

**Status :** Correct

**Marks :** 10/10

#### 4. 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**

# You are using Python

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

```
o=set()
```

```
for i in range(0,n):
```

```
    li=input().strip()
```

```
    if li:
```

```
        m=set(map(int,li.split()))
```

```
    else:
```

```
        m=set()
```

```
    o=o^m
```

```
print(o)
```

```
c=sum(o)
```

```
print(c)
```

**Status :** Partially correct

**Marks :** 9/10