

# Rajalakshmi Engineering College

Name: Kanishka S  
Email: 240701227@rajalakshmi.edu.in  
Roll no: 2116240701227  
Phone: 8825651385  
Branch: REC  
Department: I CSE AH  
Batch: 2028  
Degree: B.E - CSE

Scan to verify results



## NeoColab\_REC\_CS23221\_Python Programming

### REC\_Python\_Week1\_CY

Attempt : 1  
Total Mark : 40  
Marks Obtained : 40

### Section 1 : Coding

#### 1. Problem Statement

Mandy is working on a mathematical research project involving complex numbers. For her calculations, she often needs to swap the real and imaginary parts of two complex numbers.

Mandy needs a Python program that takes two complex numbers as input and swaps their real and imaginary values.

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

The first line of input consists of a complex number in the format  $a+bj$ , representing the first complex number.

The second line consists of a complex number in the format  $a+bj$ , representing the second complex number.

### **Output Format**

The first line of output displays "New first complex number: " followed by the swapped complex number.

The second line of output displays "New second complex number: " followed by the swapped complex number.

Refer to the sample output for the formatting specifications.

### **Sample Test Case**

Input: 10+8j

7-9j

Output: New first complex number: (8+10j)

New second complex number: (-9+7j)

### **Answer**

```
a=input()
b=input()
c1=complex(a.replace('j','j'))
c2=complex(b.replace('j','j'))
new_c1=complex(c1.imag,c1.real)
new_c2=complex(c2.imag,c2.real)
print(f"New first complex number: {new_c1}")
print(f"New second complex number: {new_c2}")
```

**Status :** Correct

**Marks :** 10/10

## **2. Problem Statement**

Nina is working on a project involving multiple sensors. Each sensor provides a data point that needs to be processed to compute an aggregated value.

Given data points from three sensors, write a program to calculate the aggregated value using specific bitwise operations and arithmetic manipulations. The final result should be the aggregated value modulo

1000.

Example:

Input:

1 //sensor 1 data

2 //sensor 2 data

3 //sensor 3 data

Output

9

Explanation

Calculate the bitwise AND of sensor 1 data and sensor 2 data: 0

Calculate the XOR of the result from step 1 and sensor 3 data: 3

Multiply the result from step 2 by 3: 9

Compute the final aggregated value by taking the result from step 3 modulo 1000: 9

So, the aggregated value is 9.

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

The first line of input consists of an integer S1, representing sensor1 data.

The second line of input consists of an integer S2, representing sensor2 data.

The third line of input consists of an integer S3, representing sensor3 data.

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

The output displays an integer representing the aggregated value.

Refer to the sample output for the formatting specifications.

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

Input: 1

2

3

Output: 9

**Answer**

# You are using Python

```
a=int(input())
```

```
b=int(input())
```

```
c=int(input())
```

```
d=(((a&b)^c)*3)%1000
```

```
print(d)
```

**Status :** Correct

**Marks : 10/10**

### 3. Problem Statement

John is developing a financial application to help users manage their investment portfolios. As part of the application, he needs to write a program that receives the portfolio's main value and the values of two specific investments as inputs. The program should then display these values in reverse order for clear visualization.

Help John achieve this functionality by writing the required program.

#### **Input Format**

The first line of input consists of a float, representing the first investment value.

The second line of input consists of a float, representing the second investment value.

The third line of input consists of an integer, representing the portfolio ID.

#### **Output Format**

The first line of output prints "The values in the reverse order:".

The second line prints the integer, representing the portfolio ID.

The third line prints the second float, representing the second investment value.

The fourth line prints the first float, representing the first investment value.

Refer to the sample output for the formatting specifications.

### **Sample Test Case**

Input: 35.29

9374.11

48

Output: The values in the reverse order:

48

9374.11

35.29

### **Answer**

# You are using Python

```
a=float(input())
```

```
b=float(input())
```

```
c=int(input())
```

```
print(f"The values in the reverse order:\n{c}\n{b}\n{a}")
```

**Status :** Correct

**Marks :** 10/10

## **4. Problem Statement**

Liam and his friends are sharing the cost of a group purchase. The total cost of the purchase is subject to a 10% discount. One of the friends receives a 35% bonus, which means they will pay a larger portion of the discounted cost. The remaining cost is then divided equally among the other friends.

Write a program to:

Calculate the total cost after applying a 10% discount. Determine the amount paid by the friend who receives a 35% bonus. Calculate the amount each of the other friends will pay.

### ***Input Format***

The first line of input consists of a float value  $f$ , representing the total cost.

The second line contains an integer value  $n$ , representing the total number of friends.

### ***Output Format***

The first line of output displays "Cost after a 10% discount: " followed by the discounted cost of the ticket package as a float value formatted to two decimal places.

The second line displays "Friend with a 35% bonus pays: " followed by the amount paid by the friend with the bonus as a float value formatted to two decimal places.

The third line displays "Each of the other friends pays: " followed by the individual share of the remaining cost as a float value formatted to two decimal places.

Refer to the sample output for the formatting specifications.

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

Input: 10000.0

5

Output: Cost after a 10% discount: 9000.00

Friend with a 35% bonus pays: 3150.00

Each of the other friends pays: 1462.50

### ***Answer***

# You are using Python

```
f=float(input())
```

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

```
discounted_cost=f*0.9
```

```
bonus_friend_pays=discounted_cost*0.35
```

```
remaining_cost=discounted_cost-bonus_friend_pays
```

```
each_other_friend_pays=remaining_cost/(n-1)
```

```
print(f"Cost after a 10% discount:{discounted_cost:.2f}")
```

```
print(f"Friend with a 35% bonus pays:{bonus_friend_pays:.2f}")
```

```
print(f"Each of the other friends pays:{each_other_friend_pays:.2f}")
```

**Status :** Correct

**Marks :** 10/10