

Rajalakshmi Engineering College

Name: Ashwin C
Email: 240701058@rajalakshmi.edu.in
Roll no:
Phone: 9940303740
Branch: REC
Department: I CSE FA
Batch: 2028
Degree: B.E - CSE

Scan to verify results



NeoColab_REC_CS23221_Python Programming

REC_Python_Week 1_MCQ

Attempt : 1
Total Mark : 15
Marks Obtained : 14

Section 1 : MCQ

1. Which of these is not a core data type?

Answer

Class

Status : Correct

Marks : 1/1

2. What is typecasting in Python?

Answer

Change data type property

Status : Correct

Marks : 1/1

3. What will be the output of the following code?

```
x = int(34.56 - 2 * 2)
print(x)
```

Answer

30

Status : Correct

Marks : 1/1

4. Evaluate the expression given below if A= 16 and B = 15

```
A % B // A
```

Answer

0

Status : Correct

Marks : 1/1

5. Which of the following functions converts a string to a float in Python?

Answer

```
float(x)
```

Status : Correct

Marks : 1/1

6. What is the return type of the function id?

Answer

int

Status : Correct

Marks : 1/1

7. Which of the following can convert the string to a float number?

Answer

```
float(str)
```

Status : Correct

Marks : 1/1

8. What will the following code output?

```
z = 3 + 4j  
print(abs(z))
```

Answer

5.0

Status : Correct

Marks : 1/1

9. Which of the following expressions results in an error?

Answer

```
int('10.8')&nbsp;
```

Status : Correct

Marks : 1/1

10. What will be the output for the below code?

```
x=15  
y=12  
print(x&y)
```

Answer

12

Status : Correct

Marks : 1/1

11. Which is the correct operator for power(xy)?

Answer

```
x**y
```

Status : Correct

Marks : 1/1

12. What is the value of the following expression?

8/4/2, 8/(4/2)

Answer

(1.0,4.0)

Status : Correct

Marks : 1/1

13. What is the value of x in the following program?

```
x = int(43.55+2/2)
print(x)
```

Answer

22

Status : Wrong

Marks : 0/1

14. What will be the value of the following Python expression?

4 + 3 % 5

Answer

7

Status : Correct

Marks : 1/1

15. What is used to concatenate two strings in Python?

Answer

+ operator

Status : Correct

Marks : 1/1

Rajalakshmi Engineering College

Name: Ashwin C
Email: 240701058@rajalakshmi.edu.in
Roll no:
Phone: 9940303740
Branch: REC
Department: I CSE FA
Batch: 2028
Degree: B.E - CSE

Scan to verify results



NeoColab_REC_CS23221_Python Programming

REC_Python_Week 1_COD

Attempt : 1
Total Mark : 5
Marks Obtained : 5

Section 1 : Coding

1. Problem Statement

Bob, the owner of a popular bakery, wants to create a special offer code for his customers. To generate the code, he plans to combine the day of the month with the number of items left in stock.

Help Bob to encode these two values into a unique offer code.

Note: Use the bitwise operator to calculate the offer code.

Example

Input:

15

9

Output:

Offer code: 6

Explanation:

Given the day of the month 15th day (binary 1111) and there are 9 items left (binary 1001), the offer code is calculated as 0110 which is 6.

Input Format

The first line of input consists of an integer D, representing the day of the month.

The second line consists of an integer S, representing the number of items left in stock.

Output Format

The output displays "Offer code: " followed by an integer representing the encoded offer code.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 15

9

Output: Offer code: 6

Answer

```
# You are using Python
d=int(input())
s=int(input())
oc=d^s
print("Offer code:",oc)
```

Status : Correct

Marks : 1/1

2. Problem Statement

A company has hired two employees, Alice and Bob. The company wants to swap the salaries of both employees. Alice's salary is an integer value and Bob's salary is a floating-point value.

Write a program to swap their salaries and print the new salary of each employee.

Input Format

The first line of input consists of an integer N, representing Alice's salary.

The second line consists of a float value F, representing Bob's salary.

Output Format

The first line of output displays "Initial salaries:"

The second line displays "Alice's salary = N", where N is Alice's salary.

The third line of output displays "Bob's salary = F", where F is Bob's salary.

After a new line space, the following line displays "New salaries after swapping:"

The next line displays "Alice's salary = X", where X is the swapped salary.

The last line displays "Bob's salary = Y", where Y is the swapped salary.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 10000
15400.55

Output: Initial salaries:
Alice's salary = 10000
Bob's salary = 15400.55

New salaries after swapping:
Alice's salary = 15400.55
Bob's salary = 10000

Answer

```
a=int(input())
b=float(input())
print("Initial salaries: ")
print("Alice's salary =",a)
print("Bob's salary =",b)
print("New salaries after swapping:")
print("Alice's salary =",b)
print("Bob's salary =",a)
```

Status : Correct**Marks : 1/1****3. Problem Statement**

In a family, two children receive allowances based on the gardening tasks they complete. The older child receives an allowance rate of Rs.5 for each task, with a base allowance of Rs.50. The younger child receives an allowance rate of Rs.3 for each task, with a base allowance of Rs.30.

Your task is to calculate and display the allowances for the older and younger children based on the number of gardening tasks they complete, along with the total allowance for both children combined.

Input Format

The first line of input consists of an integer n , representing the number of chores completed by the older child.

The second line consists of an integer m , representing the number of chores completed by the youngest child.

Output Format

The first line of output displays "Older child allowance: Rs." followed by an integer representing the allowance calculated for the older sibling.

The second line displays "Younger child allowance: Rs." followed by an integer representing the allowance calculated for the youngest sibling.

The third line displays "Total allowance: Rs." followed by an integer representing the sum of both siblings' allowances.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 10

5

Output: Older child allowance: Rs.100

Younger child allowance: Rs.45

Total allowance: Rs.145

Answer

```
n=int(input())
m=int(input())
oca=(n*5)+50
yca=(m*3)+30
tot=oca+yca
print("Older child allowance: Rs.",oca)
print("Younger child allowance: Rs.",yca)
print("Total allowance: Rs.",tot)
```

Status : Correct

Marks : 1/1

4. Problem Statement

Quentin, a mathematics enthusiast, is exploring the properties of numbers. He believes that for any set of four consecutive integers, calculating the average of their fourth powers and then subtracting the product of the first and last numbers yields a constant value.

To validate his hypothesis, check if the result is indeed constant and display.

Example:

Input:

5

Output:

Constant value: 2064.5

Explanation:

Find the Average:

Average: $(625 + 1296 + 2401 + 4096)/4 = 2104.5$

Now, we calculate the product of a and (a + 3):

Product = $5 \times (5 + 3) = 5 \times 8 = 40$

Final result: $2104.5 - 40 = 2064.5$

Input Format

The input consists of an integer a, representing the first of four consecutive integers.

Output Format

The output displays "Constant value: " followed by the computed result based on Quentin's formula.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 5

Output: Constant value: 2064.5

Answer

```
# You are using Python
a=int(input())
b1=(a+0)**4
b2=(a+1)**4
b3=(a+2)**4
b4=(a+3)**4
avg=(b1+b2+b3+b4)/4
pdt=a*(a+3)
```

```
res=avg-pdt  
print("Constant value:",res)
```

Status : Correct

Marks : 1/1

5. Problem Statement

A science experiment produces a decimal value as the result. However, the scientist needs to convert this value into an integer so that it can be used in further calculations.

Write a Python program that takes a floating-point number as input and converts it into an integer.

Input Format

The input consists of a floating point number, F.

Output Format

The output prints "The integer value of F is: {result}", followed by the integer number equivalent to the floating point number.

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: 10.36

Output: The integer value of 10.36 is: 10

Answer

```
# You are using Python  
a=float(input())  
print("The integer value of ",a,"is:",int(a))
```

Status : Correct

Marks : 1/1

Rajalakshmi Engineering College

Name: Ashwin C
Email: 240701058@rajalakshmi.edu.in
Roll no:
Phone: 9940303740
Branch: REC
Department: I CSE FA
Batch: 2028
Degree: B.E - CSE

Scan to verify results



NeoColab_REC_CS23221_Python Programming

REC_Python_Week 1_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=complex(input())
b=complex(input())
s=complex(a.imag,a.real)
p=complex(b.imag,b.real)
print("New first complex number:",s)
print("New second complex number:",p)
```

Status : Correct

Marks : 10/10

2. 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())
a=(10/100)*f
b=f-a
print(f"Cost after a 10% discount: {b:.2f}")
c=(35/100)*b
```

```
print(f"Friend with a 35% bonus pays: {c:.2f}")
d=(b-c)/(n-1)
print(f"Each of the other friends pays: {d:.2f}")
```

Status : Correct

Marks : 10/10

3. Problem Statement

Olivia is creating a wellness dashboard for her new fitness app, FitTrack. She needs a program that can capture and display key details about a user's workout. The program should read the user's full name, the total steps they ran, the energy they expended in kilojoules, and the duration of their workout in hours. After collecting this information, the program will generate a detailed summary of the user's fitness activity.

Your task is to guide Olivia through the program.

Input Format

The first line of input consists of a string, representing the user's name.

The second line consists of an integer, representing the total steps taken.

The third line consists of a float value, representing the calories burned.

The fourth line consists of a float value, representing the workout duration in hours.

Output Format

The first line of output prints "User Name: " followed by the user's name.

The second line prints "Total Steps: " followed by the total steps.

The third line prints "Calories Burned: " followed by the calories burned, rounded off to one decimal place.

The fourth line prints "Workout Duration: X hours" where X is the workout duration, rounded off to one decimal place.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: Alex

10000

350.5

1.5

Output: User Name: Alex

Total Steps: 10000

Calories Burned: 350.5

Workout Duration: 1.5 hours

Answer

```
a=str(input())
b=int(input())
c=float(input())
d=float(input())
print("User Name:",a)
print("Total Steps:",b)
print("Calories Burned:",c)
print("Workout Duration:",d," hours")
```

Status : Correct

Marks : 10/10

4. Problem Statement

Shawn is planning for his younger sister's college education and wants to ensure she has enough funds when the time comes. He starts with an initial principal amount and plans to make regular monthly contributions to a savings account that offers a fixed annual interest rate.

Shawn needs to calculate the total amount that will accumulate by the time his sister is ready for college. Your task is to write a program that calculates the final amount in the savings account based on the initial principal, monthly contributions, annual interest rate, and the number of months the money is invested.

Formula:

$$A = P \times (1 + r/n)^{(n \times t)} + C \times [(1 + r/n)^{(n \times t)} - 1] / (r/n)$$

Where:

A = Final amount after the specified time

P = Initial principal amount

C = Monthly contribution

r = Annual interest rate (as a decimal, e.g., 5% = 0.05)

n = Number of compounding periods per year (12 for monthly compounding)

t = Total time in years (months / 12)

Input Format

The first line of input consists of a float P, representing the initial principal amount.

The second line of input consists of a float R, representing the annual interest rate (in percentage).

The third line of input consists of a float C, representing the monthly contribution.

The fourth line of input consists of an integer M, representing the number of months.

Output Format

The output displays "Final amount after X months: Rs." followed by the total accumulated amount, formatted to two decimal places, where X is the number of months.

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: 10000.0

5.0

2000.0

12

Output: Final amount after 12 months: Rs.35069.33

Answer

You are using Python

P=float(input())

R=float(input())

C=float(input())

M=int(input())

t=M/12

r=R/100

n=12

A=P*(1+r/n)**(n*t)+C*(((1+r/n)**(n*t)-1)/(r/n))

print(f"Final amount after {M} months:Rs. {A:.2f}")

Status : Correct

Marks : 10/10

Rajalakshmi Engineering College

Name: Ashwin C
Email: 240701058@rajalakshmi.edu.in
Roll no:
Phone: 9940303740
Branch: REC
Department: I CSE FA
Batch: 2028
Degree: B.E - CSE

Scan to verify results



NeoColab_REC_CS23221_Python Programming

REC_Python_Week 1_PAH

Attempt : 1
Total Mark : 6
Marks Obtained : 6

Section 1 : Coding

1. Problem Statement

Liam works at a car dealership and is responsible for recording the details of cars that arrive at the showroom. To make his job easier, he wants a program that can take the car's make, model, and price, and display the information in a formatted summary.

Assist him in the program.

Input Format

The first line of input contains a string, representing the car make.

The second line contains a string, representing the car model.

The third line contains a float value, representing the car price.

Output Format

The first line of output prints "Car Make: ", followed by the car make.

The second line prints "Car Model: ", followed by the car model.

The third line prints "Price: ", followed by the car price, formatted to two decimal places.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: Toyota

Camry

23450.75

Output: Car Make: Toyota

Car Model: Camry

Price: Rs.23450.75

Answer

```
a=str(input())  
b=str(input())  
c=float(input())  
print(f"Car Make: {a}")  
print(f"Car Model: {b}")  
print(f"Price: Rs.{c:.2f}")
```

Status : Correct

Marks : 1/1

2. Problem Statement

Ella, an avid TV show enthusiast, is planning a binge-watching marathon for a new series. She has a specific routine: after watching a set number of episodes, she takes a short break.

She is provided with the following information:

Each episode of the series has a fixed duration of 45 minutes. After a

certain number of episodes, there is a break of 15 minutes.

Ella wants to know the total time she will need to watch the entire series, including the breaks. Your task is to help Ella by calculating the total viewing time.

Input Format

The first line of input consists of an integer E, representing the total number of episodes in the series.

The second line consists of an integer B, representing the number of episodes watched before taking a break.

Output Format

The output prints an integer representing the total viewing time required to watch the entire series, including the breaks.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 5

2

Output: 255 minutes

Answer

```
# You are using Python
e=int(input())
b=int(input())
tot=(45*e)+(((e-1)//b)*15)
print(f"{tot} minutes")
```

Status : Correct

Marks : 1/1

3. Problem Statement

A smart home system tracks the temperature and humidity of each room. Create a program that takes the room name (string), temperature (float),

and humidity (float).

Display the room's climate details.

Input Format

The first line of input consists of a string, representing the room name.

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

The third line consists of a float value, representing the humidity.

Output Format

The first line of output prints "Room: " followed by the room name (string).

The second line prints "Temperature: " followed by the temperature (float) formatted to two decimal places.

The third line prints "Humidity: " followed by the humidity (float) formatted to two decimal places and a percentage sign (%).

Refer to the sample output for formatting specifications.

Sample Test Case

Input: Living Room

23.45

45.78

Output: Room: Living Room

Temperature: 23.45

Humidity: 45.78%

Answer

```
# You are using Python
a=str(input())
b=float(input())
c=float(input())
print("Room:",a)
print(f"Temperature: {b:.2f}")
print(f"Humidity: {c:.2f}%")
```

Status : Correct

Marks : 1/1

4. Problem Statement

Shawn, a passionate baker, is planning to bake cookies for a large party. His original recipe makes 15 cookies, with the following ingredient quantities: 2.5 cups of flour, 1 cup of sugar, and 0.5 cups of butter.

Write a program to calculate the amounts of flour, sugar, and butter needed for a different number of cookies. Provide the ingredient quantities for a specified number of cookies, maintaining the original proportions of the recipe.

Input Format

The input consists of an integer n , representing the number of cookies.

Output Format

The first line prints "Flour: X cups" where X represents the amount of flour required for n cookies, as a double value rounded to two decimal places.

The second line prints "Sugar: Y cups" where Y represents the amount of Sugar required for n , as a double value rounded to two decimal places.

The third line prints "Butter: Z cups" where Z represents the amount of flour required for n , as a double value rounded to two decimal places.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 15

Output: Flour: 2.50 cups

Sugar: 1.00 cups

Butter: 0.50 cups

Answer

You are using Python

```
n=int(input())
a=n/6
b=n/15
c=n/30
print(f"Flour:{a:.2f}", "cups")
print(f"Sugar:{b:.2f}", "cups")
print(f"Butter:{c:.2f}", "cups")
```

Status : Correct

Marks : 1/1

5. Problem Statement

Mandy is debating with her friend Rachel about an interesting mathematical claim. Rachel asserts that for any positive integer n , the ratio of the sum of n and its triple to the integer itself is always 4. Mandy, intrigued by this statement, decides to validate it using logical operators and basic arithmetic.

She wants to confirm if the statement holds true for any positive integer n .

Input Format

The input consists of a positive integer n , representing the integer to be tested.

Output Format

The first line of output displays "Sum:" followed by an integer representing the calculated sum.

The second line displays "Rachel's statement is: " followed by a Boolean value indicating whether Rachel's statement is correct.

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: 12

Output: Sum: 48

Rachel's statement is: True

Answer

```
# You are using Python
n=int(input())
Sum=n*4
print("Sum:",Sum)
print("Rachel's statement is: True")
```

Status : Correct

Marks : 1/1

6. Problem Statement

Oliver is planning a movie night with his friends and wants to download a high-definition movie. He knows the file size of the movie in megabytes (MB) and his internet speed in megabits per second (Mbps). To ensure the movie is ready in time, Oliver needs to calculate the download time.

Your task is to write a program that calculates the download time and displays it in hours, minutes, and seconds.

Example

Input:

MB = 800

mbps = 40

Output:

Download Time: 0 hours, 2 minutes, and 40 seconds

Explanation:

Convert the file size to bits ($800 \text{ MB} \times 8 \text{ bits/byte} = 6400 \text{ megabits}$) and divide it by the download speed ($6400 \text{ Mbps} / 40 \text{ Mbps} = 160 \text{ seconds}$). Now, convert the download time in seconds to hours, minutes, and seconds: 160 seconds is equal to 2 minutes and 40 seconds. So, the download time is 0 hours, 2 minutes and 40 seconds.

Input Format

The first line of input consists of an integer N, representing the file size in megabytes (MB).

The second line consists of an integer S, representing the network speed in megabits per second(mbps).

Output Format

The output prints "Download Time: X hours, Y minutes, and Z seconds", where X, Y, and Z are integers representing the hours, minutes, and seconds respectively.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 180

3

Output: Download Time: 0 hours, 8 minutes, and 0 seconds

Answer

You are using Python

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

```
s=int(input())
```

```
a=n*8
```

```
b=a/s
```

```
c=int(b//3600)
```

```
d=int((b%3600)//60)
```

```
e=int(b%60)
```

```
print(f"Download Time: {c} hours, {d} minutes, and {e} seconds")
```

Status : Correct

Marks : 1/1

Rajalakshmi Engineering College

Name: Ashwin C
Email: 240701058@rajalakshmi.edu.in
Roll no:
Phone: 9940303740
Branch: REC
Department: I CSE FA
Batch: 2028
Degree: B.E - CSE

Scan to verify results



NeoColab_REC_CS23221_Python Programming

REC_Python_Week 2_MCQ

Attempt : 1
Total Mark : 15
Marks Obtained : 11

Section 1 : MCQ

1. What will be the output for the following code snippet?

```
i = 0  
for i in range(10):  
    break  
print(i)
```

Answer

0

Status : Correct

Marks : 1/1

2. What will be the output of the following Python code?

```
i = 1
```

```
while True:
    if i % 2 == 0:
        i += 1
        continue
    if i > 10:
        break
    print(i, end = " ")
    i += 2
```

Answer

1 3 5 7 9

Status : Correct

Marks : 1/1

3. What will be the output of the following Python code?

```
i = 5
while True:
    if i%10011 == 0:
        break
    print(i, end = " ")
    i += 1
```

Answer

5 6 7 8 9 10

Status : Wrong

Marks : 0/1

4. What will be the output of the following Python code?

```
i = 1
while False:
    if i%2 == 0:
        break
    print(i)
    i += 2
```

Answer

1 3 5 7 ...

Status : **Wrong**

Marks : 0/1

5. What will the following code output?

```
x = 0
while x < 5:
    if x == 3:
        break
    x += 1
else:
    print("Completed")
print(x)
```

Answer

3

Status : **Correct**

Marks : 1/1

6. What will be the output of the following Python code?

```
i = 1
while True:
    if i%3 == 0:
        break
    print(i)
    i += 1
```

Answer

1 2

Status : **Wrong**

Marks : 0/1

7. How many times will the inner for loop be executed in the below code?

```
i=0
while(True):
    for j in range(4,0,-2):
        print(i*j)
```

```
print("")
i=i+1
if(i%2==0):
    break
```

Answer

02

Status : Correct

Marks : 1/1

8. What is the output of the following?

```
True = False
while True:
    print(True)
    break
```

Answer

False

Status : Wrong

Marks : 0/1

9. What is the output of the following program?

```
i=1
while(i<3):
    j=0
    while(j<3):
        print(i%3,end=" ")
        j=j+1
    i=i+1
```

Answer

1 1 1 2 2 2

Status : Correct

Marks : 1/1

10. What will be the output of the following Python code?

```
i = 1
while True:
    if i%3 == 0:
        break
    print(i)
    i += 1
```

Answer

1 2

Status : Correct

Marks : 1/1

11. Which keyword is used to immediately terminate a loop?

Answer

break

Status : Correct

Marks : 1/1

12. What will be the output of the following code snippet?

```
balloon_inflated = False
while not balloon_inflated:
    if not balloon_inflated:
        balloon_inflated = True
        print("inflate-", end="")
print("done")
```

Answer

inflate-done

Status : Correct

Marks : 1/1

13. What will be the output of the following Python code?

```
i = 5
while True:
    if i%0011 == 0:
```

```
    break
print(i)
i += 1
```

Answer

5 6 7 8

Status : Correct

Marks : 1/1

14. Which keyword used in loops can skip the remaining statements for a particular iteration and start the next iteration?

Answer

continue

Status : Correct

Marks : 1/1

15. What is the output of the following?

```
for i in range(10):
    if i == 5:
        break
    else:
        print(i, end=' ')
else:
    print("Here")
```

Answer

0 1 2 3 4

Status : Correct

Marks : 1/1

Rajalakshmi Engineering College

Name: Ashwin C
Email: 240701058@rajalakshmi.edu.in
Roll no:
Phone: 9940303740
Branch: REC
Department: I CSE FA
Batch: 2028
Degree: B.E - CSE

Scan to verify results



NeoColab_REC_CS23221_Python Programming

REC_Python_Week 2_COD_Updated

Attempt : 1
Total Mark : 50
Marks Obtained : 50

Section 1 : Coding

1. Problem Statement

Ethan, a curious mathematician, is fascinated by perfect numbers. A perfect number is a number that equals the sum of its proper divisors (excluding itself). Ethan wants to identify all perfect numbers within a given range.

Help him write a program to list these numbers.

Input Format

The first line of input consists of an integer start, representing the starting number of the range.

The second line consists of an integer end, representing the ending number of the range.

Output Format

The output prints all perfect numbers in the range, separated by a space.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 1

100

Output: 6 28

Answer

```
# You are using Python
start=int(input())
end=int(input())
for i in range(start,end+1):
    divisorsum=0
    for j in range(1,i):
        if(i%j==0):
            divisorsum+=j
    if(divisorsum==i):
        print(i)
```

Status : Correct

Marks : 10/10

2. Problem Statement

Emma, a mathematics enthusiast, is exploring a range of numbers and wants to count how many of them are not Fibonacci numbers.

Help Emma determine the count of non-Fibonacci numbers within the given range [start, end] using the continue statement.

Input Format

The first line of input consists of an integer, representing the starting number of the range.

The second line consists of an integer, representing the ending number of the range.

Output Format

The output prints a single integer, representing the count of numbers in the range that are not Fibonacci numbers.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 1

10

Output: 5

Answer

```
# You are using Python
def isfibonacci(n):
    a,b=0,1
    while b<n:
        a,b=b,a+b
    return b==n
start=int(input())
end=int(input())
nonfibcount=0
for i in range(start,end+1):
    if isfibonacci(i):
        continue
    nonfibcount+=1
print(nonfibcount)
```

Status : Correct

Marks : 10/10

3. Problem Statement

You work as an instructor at a math enrichment program, and your goal is to develop a program that showcases the concept of using control statements to manipulate loops. Your task is to create a program that

takes an integer 'n' as input and prints the squares of even numbers from 1 to 'n', while skipping odd numbers.

Input Format

The input consists of a single integer, which represents the upper limit of the range.

Output Format

The output displays the square of even numbers from 1 to 'n' separated by lines.

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: 10

Output: 4

16

36

64

100

Answer

```
# You are using Python
n=int(input())
for i in range(1,n+1):
    if(i%2==0):
        print(i**2)
```

Status : Correct

Marks : 10/10

4. Problem Statement

John, a software developer, is analyzing a sequence of numbers within a given range to calculate their digit sum. However, to simplify his task, he excludes all numbers that are palindromes (numbers that read the same backward as forward).

Help John find the total sum of the digits of non-palindromic numbers in the range [start, end] (both inclusive).

Example:

Input:

10

20

Output:

55

Explanation:

Range [10, 20]: Non-palindromic numbers are 10, 12, 13, 14, 15, 16, 17, 18, 19 and 20.

Digit sums: $1+0 + 1+2 + 1+3 + 1+4 + 1+5 + 1+6 + 1+7 + 1+8 + 1+9 + 2+0 = 55$.

Output: 55

Input Format

The first line of input consists of an integer, representing the starting number of the range.

The second line of input consists of an integer, representing the ending number of the range.

Output Format

The output prints a single integer, representing the total sum of the digits of all non-palindromic numbers in the range.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 10

20

Output: 55

Answer

```
# You are using Python
def ispal(n):
    return str(n)==str(n) [::-1]
def digitsum(n):
    return sum(int(digit)for digit in str(n))
start=int(input())
end=int(input())
totalsum=0
for num in range(start,end+1):
    if ispal(num):
        continue
    totalsum+=digitsum(num)
print(totalsum)
```

Status : Correct

Marks : 10/10

5. Problem Statement

As a junior developer working on a text analysis project, your task is to create a program that displays the consonants in a sentence provided by the user, separated by spaces.

You need to implement a program that takes a sentence as input and prints the consonants while skipping vowels and non-alphabetic characters using only control statements.

Input Format

The input consists of a string representing the sentence.

Output Format

The output displays space-separated consonants present in the sentence.

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: Hello World!

Output: H l l W r l d

Answer

```
# You are using Python
str1=str(input())
cons=""
for char in str1:
    if char.isalpha() and char.lower() not in 'aeiou':
        cons+=char + " "
print(cons.strip())
```

Status : Correct

Marks : 10/10

Rajalakshmi Engineering College

Name: Ashwin C
Email: 240701058@rajalakshmi.edu.in
Roll no:
Phone: 9940303740
Branch: REC
Department: I CSE FA
Batch: 2028
Degree: B.E - CSE

Scan to verify results



NeoColab_REC_CS23221_Python Programming

REC_Python_Week 2_CY

Attempt : 1
Total Mark : 40
Marks Obtained : 40

Section 1 : Coding

1. Problem Statement

Alex is practicing programming and is curious about prime and non-prime digits. He wants to write a program that calculates the sum of the non-prime digits in a given integer using loops.

Help Alex to complete his task.

Example:

Input:

845

output:

12

Explanation:

Digits: 8 (non-prime), 4 (non-prime), 5 (prime)

The sum of Non-Prime Digits: $8 + 4 = 12$

Output: 12

Input Format

The input consists of a single integer X.

Output Format

The output prints an integer representing the sum of non-prime digits in X.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 845

Output: 12

Answer

```
# You are using Python
def sumofnonprime(n):
    nonprimedigits={0,1,4,6,8,9}
    total=0
    for digit in str(n):
        if int(digit) in nonprimedigits:
            total=total+int(digit)
    return total
n=int(input())
print(sumofnonprime(n))
```

Status : Correct

Marks : 10/10

2. Problem Statement

Max is fascinated by prime numbers and the Fibonacci sequence. He

wants to combine these two interests by creating a program that outputs the first n prime numbers within the Fibonacci sequence.

Your task is to help Max by writing a program that prints the first n prime numbers in the Fibonacci sequence using a while loop along with the break statement to achieve the desired functionality.

Input Format

The input consists of an integer n, representing the number of prime Fibonacci numbers to generate.

Output Format

The output displays space-separated first n prime numbers found in the Fibonacci sequence.

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: 5

Output: 2 3 5 13 89

Answer

```
# You are using Python
def isprime(num):
    if num<2:
        return False
    for i in range(2,int(num**0.5)+1):
        if num%i==0:
            return False
    return True
def fibprime(n):
    fib1,fib2=0,1
    count=0
    primes=[ ]
    while True:
        fib1,fib2=fib2,fib1+fib2
        if isprime(fib1):
            primes.append(fib1)
```

```
        count+=1
    if count==n:
        break
    return "\n".join(map(str,primes))
n=int(input())
print(fibprime(n))
```

Status : Correct

Marks : 10/10

3. Problem Statement

Students are allowed to work on our computer center machines only after entering the correct secret code. If the code is correct, the message "Logged In" is displayed. They are not allowed to log in to the machine until they enter the correct secret code.

Write a program to allow the student to work only if he/she enters the correct secret code.

Note: Here, secret code means the last three digits should be divisible by the first digit of the number.

Input Format

The input consists of an integer n, which represents the secret code.

Output Format

The output displays either "Logged In" or "Incorrect code" based on the given condition.

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: 2345

Output: Incorrect code

Answer

```
# You are using Python
def isvalidcode(n):
    strn=str(n)
    if len(strn)<4:
        return False
    firstdigit=int(strn[0])
    lastthreedigits=int(strn[-3:])
    return firstdigit!=0 and lastthreedigits%firstdigit==0
n=int(input())
if isvalidcode(n):
    print("Logged In")
else:
    print("Incorrect code")
```

Status : Correct

Marks : 10/10

4. Problem Statement

Nisha is a mathematics enthusiast, eager to explore the realm of twin prime numbers. The objective is to develop a program that enables the discovery and presentation of twin prime pairs.

The program should take an integer 'n' as input and generate 'n' pairs of twin primes, displaying the pairs with a difference of 2 between them.

Input Format

The input consists of a single integer, n.

Output Format

The output displays the 'n' pairs of twin primes, the pairs with a difference of 2 between them.

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: 5

Output: 3 5
5 7
11 13
17 19
29 31

Answer

```
# You are using Python
def isprime(num):
    if num<2:
        return False
    for i in range(2,int(num ** 0.5)+1):
        if num%i==0:
            return False
    return True
def findtwinprimes(n):
    count=0
    num=3
    twinprimes=[]
    while count<n:
        if isprime(num) and isprime(num+2):
            twinprimes.append(f"{num} {num+2}")
            count+=1
        num+=2
    return twinprimes
n=int(input())
for pair in findtwinprimes(n):
    print(pair)
```

Status : Correct

Marks : 10/10

Rajalakshmi Engineering College

Name: Ashwin C
Email: 240701058@rajalakshmi.edu.in
Roll no:
Phone: 9940303740
Branch: REC
Department: I CSE FA
Batch: 2028
Degree: B.E - CSE

Scan to verify results



NeoColab_REC_CS23221_Python Programming

REC_Python_Week 2_PAH_Updated

Attempt : 1
Total Mark : 60
Marks Obtained : 60

Section 1 : Coding

1. Problem Statement

As a software engineer, your goal is to develop a program that facilitates the identification of leap years in a specified range. Your task is to create a program that takes two integer inputs, representing the start and end years of the range and then prints all the leap years within that range.

Input Format

The first line of the input consists of an integer, which represents the start year.

The second line consists of an integer, which represents the end year.

Output Format

The output displays the leap years within the given range, separated by lines.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 2000

2053

Output: 2000

2004

2008

2012

2016

2020

2024

2028

2032

2036

2040

2044

2048

2052

Answer

```
def isleapyear(year):
    return(year%4==0 and year%100!=0)or(year%400==0)
def findleapyears(start,end):
    leapyears=[year for year in range(start,end+1)if isleapyear(year)]
    return leapyears
startyear=int(input())
endyear=int(input())
for year in findleapyears(startyear,endyear):
    print(year)
```

Status : Correct

Marks : 10/10

2. Problem Statement

Sophia, a primary school teacher, wants to calculate the sum of numbers within a given range, excluding those that are multiples of 3.

Write a program to help Sophia compute the sum of all numbers between start and end (inclusive) that are not divisible by 3 using the continue statement.

Input Format

The first line of input consists of an integer, representing the starting number of the range.

The second line of input consists of an integer, representing the ending number of the range.

Output Format

The output prints a single integer, representing the sum of numbers in the range that are not multiples of 3.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 1

10

Output: 37

Answer

```
# You are using Python
def sumexmulo3(start,end):
    total=0
    for num in range(start,end+1):
        if num%3==0:
            continue
        total+=num
    return total
start=int(input())
end=int(input())
print(sumexmulo3(start,end))
```

Status : Correct

Marks : 10/10

3. Problem Statement

Aarav is fascinated by the concept of summing numbers separately based on their properties. He plans to write a program that calculates the sum of even numbers and odd numbers separately from 1 to a given positive integer.

Aarav wants to input an integer value to represent the upper limit of the range. Help Aarav by developing a program that computes and displays the sum of even and odd numbers separately.

Input Format

The input consists of a single integer N, where N is the upper limit of the range.

Output Format

The output consists of two lines:

- The first line displays the sum of even numbers from 1 to N.
- The second line displays the sum of odd numbers from 1 to N.

Refer to the sample output for the exact format.

Sample Test Case

Input: 10

Output: Sum of even numbers from 1 to 10 is 30

Sum of odd numbers from 1 to 10 is 25

Answer

You are using Python

```
def sumevenodd(n):  
    evensum=sum(num for num in range(1,n+1)if num%2==0)  
    oddsum=sum(num for num in range(1,n+1)if num%2!=0)  
    return evensum,oddsum  
n=int(input())  
evensum,oddsum=sumevenodd(n)  
print(f"Sum of even numbers from 1 to {n} is {evensum}")  
print(f"Sum of odd numbers from 1 to {n} is {oddsum}")
```

Status : Correct

Marks : 10/10

4. Problem Statement

Rajesh wants to design a program that simulates a real-time scenario based on a mathematical concept known as the Collatz Conjecture. This concept involves the repeated application of rules to a given starting number until the number becomes 1. The rules are as follows:

If the number is even, divide it by 2. If the number is odd, multiply it by 3 and add 1.

Your task is to write a program that takes a positive integer as input, applies the Collatz Conjecture rules to it, counts the number of steps taken to reach 1, and provides an output accordingly. If the process exceeds 100 steps, the program should print a message indicating so and use break to exit.

Input Format

The input consists of a single integer, n.

Output Format

The output displays the total number of steps taken to reach 1 if it's under 100.

If it's more than 100, it displays "Exceeded 100 steps. Exiting...".

Refer to sample output for the formatting specifications.

Sample Test Case

Input: 6

Output: Steps taken to reach 1: 8

Answer

```
# You are using Python
def collatz_conjecture(n):
    steps=0
```

```
while n!=1:
    if steps>=100:
        print("Exceeded 100 steps. Exiting...")
        break
    if n%2==0:
        n//=2
    else:
        n=3*n+1
    steps+=1
else:
    print(f"Steps taken to reach 1: {steps}")
n=int(input())
if n>0:
    collatz_conjecture(n)
```

Status : Correct

Marks : 10/10

5. Problem Statement

Kamali recently received her electricity bill and wants to calculate the amount she needs to pay based on her usage. The electricity company charges different rates based on the number of units consumed.

For the first 100 units, there is no charge. For units consumed beyond 100 and up to 200, there is a charge of Rs. 5 per unit. For units consumed beyond 200, there is a charge of Rs. 10 per unit.

Write a program to help Kamali calculate the amount she needs to pay for her electricity bill based on the units consumed.

Input Format

The input consists of an integer, representing the number of units.

Output Format

The output prints the total amount of the electricity bill, an integer indicating the amount Kamali needs to pay in the format "Rs. amount".

Refer to the sample output for the exact format.

Sample Test Case

Input: 350

Output: Rs. 2000

Answer

```
# You are using Python
def calcbill(units):
    if units<=100:
        amt=0
    elif units<=200:
        amt=(units-100)*5
    else:
        amt=(500)+(units-200)*10
    print(f"Rs. {amt}")
units=int(input())
calcbill(units)
```

Status : Correct

Marks : 10/10

6. Problem Statement

Imagine being entrusted with the responsibility of creating a program that simulates a math workshop for students. Your task is to develop an interactive program that not only calculates but also showcases the charm of factorial values. Your program should efficiently compute and present the sum of digits for factorial values of only odd numbers within a designated range. This approach will ingeniously keep even factorials at bay, allowing students to delve into the intriguing world of mathematics with enthusiasm and clarity.

Input Format

The input consists of a single integer, n.

Output Format

The output displays the factorial and sum of digits of the factorial of odd numbers within the given range.

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: 6

Output: 1! = 1, sum of digits = 1

3! = 6, sum of digits = 6

5! = 120, sum of digits = 3

Answer

```
# You are using Python
import math
def sod(n):
    return sum(int(digit) for digit in str(n))
def facwor(n):
    for i in range(1,n+1,2):
        fact=math.factorial(i)
        ds=sod(fact)
        print(f"{i}! = {fact}, sum of digits = {ds}")
n=int(input())
facwor(n)
```

Status : Correct

Marks : 10/10

Rajalakshmi Engineering College

Name: Ashwin C
Email: 240701058@rajalakshmi.edu.in
Roll no:
Phone: 9940303740
Branch: REC
Department: I CSE FA
Batch: 2028
Degree: B.E - CSE

Scan to verify results



NeoColab_REC_CS23221_Python Programming

REC_Python_Week 3_MCQ

Attempt : 1
Total Mark : 25
Marks Obtained : 22

Section 1 : MCQ

1. What is the output of the following Python code?

```
text = " Python "  
answer = text.strip()  
print(answer)
```

Answer

Python

Status : Correct

Marks : 1/1

2. If you have a list `lst = [1, 2, 3, 4, 5, 6]`, what does the slicing operation `lst[-3:]` return?

Answer

The last three elements of the list

Status : Correct

Marks : 1/1

3. What will be the output of the following program?

```
numbers = [1, 2, 3, 4, 5]
numbers.append(6, 7)
print(numbers)
```

Answer

Compile Time Error

Status : Correct

Marks : 1/1

4. What is the output of the following Python code?

```
word = "Python"
result = word[::-1]
print(result)
```

Answer

nohtyP

Status : Correct

Marks : 1/1

5. What is the output of the following Python code?

```
word = "programming"
answer = word.index("gram")
print(answer)
```

Answer

3

Status : Correct

Marks : 1/1

6. What is the output of the following Python code?

```
text = "Python"  
result = text.center(10, "*")  
print(result)
```

Answer

```
**Python**
```

Status : Correct

Marks : 1/1

7. What does the append() method do in Python?

Answer

Adds a new element to the end of the list

Status : Correct

Marks : 1/1

8. What is the output of the following Python code?

```
txt = "My Classroom"  
print(txt.find("o"))  
print(txt.index("o"))
```

Answer

99

Status : Correct

Marks : 1/1

9. What is the output of the following code?

```
my_list = [1, 2, 3]  
my_list *= 2  
print(len(my_list))
```

Answer

3

Status : Wrong

Marks : 0/1

10. Which method is used to add multiple items to the end of a list?

Answer

`extend()`

Status : Correct

Marks : 1/1

11. What will be the output of the following code?

```
my_list = [1, 2, 2, 3]
print(my_list.count(2))
```

Answer

2

Status : Correct

Marks : 1/1

12. What will be the output of the following code?

```
numbers = [1, 2, 3, 4, 5]
numbers.remove(6)
print(numbers)
```

Answer

ValueError: list.remove(x): x not in list

Status : Correct

Marks : 1/1

13. What is the result of the slicing operation `lst[-5:-2]` on the list `lst = [1, 2, 3, 4, 5, 6]`?

Answer

[2, 3, 4]

Status : Correct

Marks : 1/1

14. Suppose `list1` is `[2, 33, 222, 14, 25]`, What is `list1[-1]`?

Answer

25

Status : Correct

Marks : 1/1

15. What does negative indexing in Python lists allow you to do?

Answer

Access elements in the list from the end

Status : Correct

Marks : 1/1

16. What is the output of the following Python code?

```
a = "Hello"  
b = "World"  
c = a + " " + b  
print(c)
```

Answer

Hello World

Status : Correct

Marks : 1/1

17. Which method in Python is used to create an empty list?

Answer

empty_list()

Status : Wrong

Marks : 0/1

18. Suppose list1 is [2, 33, 222, 14, 25], What is list1[:-1]?

Answer

[2, 33, 222, 14]

Status : Correct

Marks : 1/1

19. What is the output of the following Python code?

```
name = "John"  
age = 25  
message = "My name is %s and I am %d years old." % (name, age)  
print(message)
```

Answer

My name is John and I am 25 years old.

Status : Correct

Marks : 1/1

20. What is the output of the following Python code?

```
b = "Projects!"  
print(b[2:5])
```

Answer

oje

Status : Correct

Marks : 1/1

21. Suppose list1 is [4, 2, 2, 4, 5, 2, 1, 0], Which of the following is the correct syntax for slicing operation?

Answer

```
print(list1[2:])
```

Status : Wrong

Marks : 0/1

22. Which of the following is a valid way to use the '%' operator to concatenate strings in Python?

Answer

```
"%s %s" % (string1, string2)
```

Status : Correct

Marks : 1/1

23. What does the following code output?

```
lst = [10, 20, 30, 40, 50]  
print(lst[-4:-1])
```

Answer

[20, 30, 40]

Status : Correct

Marks : 1/1

24. What is the output of the following code?

```
my_list = [3, 6, 1, 2, 5, 4]  
print(sorted(my_list) == my_list.sort())
```

Answer

False

Status : Correct

Marks : 1/1

25. What is the output of the following Python code?

```
string1 = "Hello"  
string2 = "World"  
result = string1 + string2  
print(result)
```

Answer

HelloWorld

Status : Correct

Marks : 1/1

Rajalakshmi Engineering College

Name: Ashwin C
Email: 240701058@rajalakshmi.edu.in
Roll no:
Phone: 9940303740
Branch: REC
Department: I CSE FA
Batch: 2028
Degree: B.E - CSE

Scan to verify results



NeoColab_REC_CS23221_Python Programming

REC_Python_Week 3_COD

Attempt : 1
Total Mark : 50
Marks Obtained : 50

Section 1 : Coding

1. Problem Statement

Ram is working on a program to manipulate strings. He wants to create a program that takes two strings as input, reverses the second string, and then concatenates it with the first string.

Ram needs your help to design a program.

Input Format

The input consists of two strings in separate lines.

Output Format

The output displays a single line containing the concatenated string of the first string and the reversed second string.

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: hello
word

Output: hellodrow

Answer

```
# You are using Python
def reversestring(s):
    return s[::-1]
str1=str(input())
str2=str(input())
rstr2=reversestring(str2)
result=str1+rstr2
print(result)
```

Status : Correct

Marks : 10/10

2. Problem Statement

Alex is working on a Python program to manage a list of elements. He needs to append multiple elements to the list and then remove an element from the list at a specified index.

Your task is to create a program that helps Alex manage the list. The program should allow Alex to input a list of elements, append them to the existing list, and then remove an element at a specified index.

Input Format

The first line contains an integer n , representing the number of elements to be appended to the list.

The next n lines contain integers, representing the elements to be appended to the list.

The third line of input consists of an integer M , representing the index of the

element to be popped from the list.

Output Format

The first line of output displays the original list.

The second line of output displays the list after popping the element of the index M.

The third line of output displays the popped element.

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: 5

64

98

-1

5

26

3

Output: List after appending elements: [64, 98, -1, 5, 26]

List after popping last element: [64, 98, -1, 26]

Popped element: 5

Answer

```
elements=[]
n=int(input())
for _ in range(n):
    elem=int(input())
    elements.append(elem)
M=int(input())
print("List after appending elements:",elements)
poppedelement=elements.pop(M)
print("List after popping last element:",elements)
print("Popped element:",poppedelement)
```

Status : Correct

Marks : 10/10

3. Problem Statement

Given a list of positive and negative numbers, arrange them such that all negative integers appear before all the positive integers in the array. The order of appearance should be maintained.

Example

Input:

[12, 11, -13, -5, 6, -7, 5, -3, -6]

Output:

List = [-13, -5, -7, -3, -6, 12, 11, 6, 5]

Explanation:

The output is the arranged list where all the negative integers appear before the positive integers while maintaining the original order of appearance.

Input Format

The input consists of a single line containing a list of integers enclosed in square brackets separated by commas.

Output Format

The output displays "List = " followed by an arranged list of integers as required, separated by commas and enclosed in square brackets.

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: [12, 11, -13, -5, 6, -7, 5, -3, -6]

Output: List = [-13, -5, -7, -3, -6, 12, 11, 6, 5]

Answer

You are using Python


```
inputstr=input()
arr=list(map(int,inputstr.strip('[]').split(',')))
neg=[x for x in arr if x<0]
positives=[x for x in arr if x>=0]
result=neg+positives
print("List = ",result)
```

Status : Correct

Marks : 10/10

4. Problem Statement

You have a string containing a phone number in the format "(XXX) XXX-XXXX". You need to extract the area code from the phone number and create a new string that contains only the area code.

Write a Python program for the same.

Note

(XXX) - Area code

XXX-XXXX - Phone number

Input Format

The input consists of a string, representing the phone number in the format "(XXX) XXX-XXXX".

Output Format

The output displays "Area code: " followed by a string representing the area code for the given phone number.

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: (123) 456-7890

Output: Area code: 123

Answer

```
# You are using Python
phno=input()
areacode=phno[1:4]
print("Area code:",areacode)
```

Status : Correct

Marks : 10/10

5. Problem Statement

Dhruv wants to write a program to slice a given string based on user-defined start and end positions.

The program should check whether the provided positions are valid and then return the sliced portion of the string if the positions are within the string's length.

Input Format

The first line consists of the input string as a string.

The second line consists of the start position (0-based index) as an integer.

The third line consists of the end position (0-based index) as an integer.

Output Format

The output displays the following format:

If the start and end positions are valid, print the sliced string.

If the start and end positions are invalid, print "Invalid start and end positions".

Refer to the sample output for formatting specifications.

Sample Test Case

Input: pythonprogramming

0

5

Output: python

Answer

```
# You are using Python
inputstr=input()
start=int(input())
end=int(input())
if 0<=start<=len(inputstr) and 0<=end<=len(inputstr) and start<=end:
    sliced=inputstr[start:end+1]
    print(sliced)
else:
    print("Invalid start and end positions")
```

Status : Correct

Marks : 10/10

Rajalakshmi Engineering College

Name: Ashwin C
Email: 240701058@rajalakshmi.edu.in
Roll no:
Phone: 9940303740
Branch: REC
Department: I CSE FA
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 : 25

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 World

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 World

Answer

```
# You are using Python
string1=input()
string2=input()
char1=input()
char2=input()
mod1=string1.replace(char1,char2)
mod2=string2.replace(char1,char2)
print("{} {}".format(mod1,mod2))
```

Status : Correct

Marks : 10/10

2. Problem Statement

Raj wants to write a program that takes a list of strings as input and returns the longest word in the list. If there are multiple words with the same length, the program should return the first one encountered.

Help Raj in his task.

Input Format

The input consists of a single line of space-separated strings.

Output Format

The output prints a string representing the longest word in the given list.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: cat dog elephant lion tiger giraffe

Output: elephant

Answer

```
# You are using Python
words=input().split()
longword=words[0]
for word in words:
    if len(word)>len(longword):
```

```
longword=word
print(longword)
```

Status : Correct

Marks : 10/10

3. Problem Statement

Write a program to check if a given string is perfect.

A perfect string must satisfy the following conditions:

The string starts with a consonant. The string alternates between consonants and vowels. Each consonant appears exactly once. Vowels can occur consecutively multiple times but should not be followed immediately by a consonant.

If the string satisfies all these conditions, print "True"; otherwise, print "False".

Input Format

The input consists of a string.

Output Format

The output prints "True" if the string is perfect. Otherwise, print "False".

Refer to the sample output for formatting specifications.

Sample Test Case

Input: capacitor

Output: True

Answer

```
# You are using Python
s=input().strip()
vowels='aeiouAEIOU'
seenconstants=set()
n=len(s)
```

```

i=0
perfect=True
if n==0 or not s[0].isalpha() or s[0] in vowels:
    perfect=False
else:
    while i<n:
        if s[i].isalpha() and s[i] not in vowels:
            if s[i] in seenconstants:
                perfect=False
                break
            seenconstants.add(s[i])
            i+=1
        else:
            perfect=False
            break
        if i<n and s[i] in vowels:
            while i<n and s[i] in vowels:
                i+=1
            if i<n and s[i] not in vowels:
                perfect=False
                break
    if perfect:
        print("True")
    else:
        print("False")

```

Status : Partially correct

Marks : 5/10

Rajalakshmi Engineering College

Name: Ashwin C
Email: 240701058@rajalakshmi.edu.in
Roll no:
Phone: 9940303740
Branch: REC
Department: I CSE FA
Batch: 2028
Degree: B.E - CSE

Scan to verify results



NeoColab_REC_CS23221_Python Programming

REC_Python_Week 3_PAH

Attempt : 1
Total Mark : 60
Marks Obtained : 57.5

Section 1 : Coding

1. Problem Statement

You are tasked with writing a program that takes n integers as input from the user and stores them in a list. After this, you need to transform the list according to the following rules:

The element at index 0 should be replaced with 0. For elements at even indices (excluding index 0), replace the element with its cube. For elements at odd indices, replace the element with its square.

Additionally, you should sort the list in ascending order before applying these transformations.

Input Format

The first line of input represents the size of the list, N .

The elements of the list are represented by the next N lines.

Output Format

The first line of output displays "Original List: " followed by the original list.

The second line displays "Replaced List: " followed by the replacement list as per the given condition.

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: 5

5
1
2
3
4

Output: Original List: [1, 2, 3, 4, 5]

Replaced List: [0, 4, 27, 16, 125]

Answer

You are using Python

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

```
nums=[]
```

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

```
    val=int(input())
```

```
    nums.append(val)
```

```
nums.sort()
```

```
print("Original List:",nums)
```

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

```
    if i==0:
```

```
        nums[i]=0
```

```
    elif i%2==0:
```

```
        nums[i]=nums[i]**3
```

```
    else:
```

```
        nums[i]=nums[i]**2
```

```
print("Replaced List:",nums)
```

Status : Correct

Marks : 10/10

2. Problem Statement

Gowri was doing her homework. She needed to write a paragraph about modern history. During that time, she noticed that some words were repeated repeatedly. She started counting the number of times a particular word was repeated.

Your task is to help Gowri to write a program to get a string from the user. Count the number of times a word is repeated in the string.

Note: Case-sensitive

Input Format

The first line of input consists of a string, str1.

The second line consists of a single word that needs to be counted, str2.

Output Format

The output displays the number of times the given word is in the string.

If the second string str2 is not present in the first string str1, it prints 0.

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: I felt happy because I saw the others were happy and because I knew I should feel happy

happy

Output: 3

Answer

```
import string
str1=input()
str2=input()
translator=str.maketrans(" ",string.punctuation)
str1cleaned=str1.translate(translator)
words=str1cleaned.split()
```

```
count=words.count(str2)
print(count)
```

Status : Partially correct

Marks : 7.5/10

3. Problem Statement

Accept an unsorted list of length n with both positive and negative integers, including 0. The task is to find the smallest positive number missing from the array. Assume the n value is always greater than zero.

Input Format

The first line consists of n , which means the number of elements in the array.

The second line consists of the values in the list as space-separated integers.

Output Format

The output displays the smallest positive number, which is missing from the array.

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: 6
-5 2 0 -1 -10 2
Output: 1

Answer

```
# You are using Python
n=int(input())
nums=list(map(int,input().split()))
for i in range(n):
    if nums[i]<=0:
        nums[i]=n+1
for i in range(n):
    val=abs(nums[i])
    if 1<=val<=n:
```

```
    if nums[val-1]>0:
        nums[val-1]=-nums[val-1]
for i in range(n):
    if nums[i]>0:
        print(i+1)
        break
else:
    print(n+1)
```

Status : Correct

Marks : 10/10

4. Problem Statement

Kyara is analyzing a series of measurements taken over time. She needs to identify all the "peaks" in this list of integers.

A peak is defined as an element that is greater than its immediate neighbors. Boundary elements are considered peaks if they are greater than their single neighbor.

Your task is to find and list all such peaks using list comprehension.

Example

Input

1 3 2 4 1 5 7 6 10 2 8

Output

Peaks: [3, 4, 7, 10, 8]

Explanation

3 is a peak because it's greater than 1 and 2.

4 is a peak because it's greater than 2 and 1.

7 is a peak because it's greater than 5 and 6.

10 is a peak because it's greater than 6 and 2.

8 is a peak because it is an boundary element and it is greater than 2.

Input Format

The input consists of several integers separated by spaces, representing the measurements.

Output Format

The output displays "Peaks: " followed by a list of integers, representing the peak elements in the list.

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: 1 3 2 4 1 5 7 6 10 2 8

Output: Peaks: [3, 4, 7, 10, 8]

Answer

```
# You are using Python
nums=list(map(int,input().split()))
peaks=[nums[i] for i in range(len(nums)) if(i==0 and nums[i]>nums[i
+1])or(i==len(nums)-1 and nums[i]>nums[i-1])or(0<i<len(nums)-1 and
nums[i]>nums[i-1] and nums[i]>nums[i+1])]
print("Peaks:",peaks)
```

Status : Correct

Marks : 10/10

5. Problem Statement

Neha is learning string operations in Python and wants to practice using built-in functions. She is given a string A, and her task is to:

Find the length of the string using a built-in function.Copy the content of A into another string B using built-in functionality.

Help Neha implement a program that efficiently performs these operations.

Input Format

The input consists of a single line containing the string A (without spaces).

Output Format

The first line of output prints the length of the given string.

The second line prints the copied string without an extra newline at the end.

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: technology-23

Output: Length of the string: 13

Copied string: technology-23

Answer

```
# You are using Python
A=input()
length=len(A)
B=A[:]
print("Length of the string:",length)
print("Copied string:",B)
```

Status : Correct

Marks : 10/10

6. Problem Statement

Imagine you are developing a text analysis tool for a cybersecurity company. Your task is to analyze input strings to categorize and count the characters into four categories: uppercase letters, lowercase letters, digits, and special characters. The company needs this tool to process log files and identify potential security threats.

Input Format

The input consists of the log entry provided as a single string.

Output Format

The output consists of four lines:

The first line contains an integer representing the count of uppercase letters in the format "Uppercase letters: {uppercase count}".

The second line contains an integer representing the count of lowercase letters in the format "Lowercase letters: {lowercase count}".

The third line contains an integer representing the count of digits in the format "Digits: {digits count}".

The fourth line contains an integer representing the count of special characters in the format "Special characters: {special characters count}".

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: Hello123

Output: Uppercase letters: 1

Lowercase letters: 4

Digits: 3

Special characters: 0

Answer

You are using Python

```
logentry=input()
```

```
uppercase=0
```

```
lowercase=0
```

```
digits=0
```

```
special=0
```

```
for char in logentry:
```

```
    if char.isupper():
```

```
        uppercase+=1
```

```
    elif char.islower():
```

```
        lowercase+=1
```

```
    elif char.isdigit():
```

```
        digits+=1
```



```
    else:  
        special+=1  
print(f"Uppercase letters:",uppercase)  
print(f"Lowercase letters:",lowercase)  
print(f"Digits:",digits)  
print(f"Special characters:",special)
```

Status : Correct

Marks : 10/10

Rajalakshmi Engineering College

Name: Ashwin C
Email: 240701058@rajalakshmi.edu.in
Roll no:
Phone: 9940303740
Branch: REC
Department: I CSE FA
Batch: 2028
Degree: B.E - CSE

Scan to verify results



NeoColab_REC_CS23221_Python Programming

REC_Python_Week 4_MCQ

Attempt : 1
Total Mark : 15
Marks Obtained : 14

Section 1 : MCQ

1. What is the main advantage of using lambda functions in Python?

Answer

They allow you to write shorter code than regular functions

Status : Correct

Marks : 1/1

2. What will be the output of the following Python code?

```
def cube(x):  
    return x * x * x  
x = cube(3)  
print(x)
```

Answer

27

Status : Correct

Marks : 1/1

3. What will be the output of the following Python code?

```
multiply = lambda x, y: x * y  
print(multiply(2, 'Hello'))
```

Answer

HelloHello

Status : Correct

Marks : 1/1

4. What will be the output of the following Python code?

```
def absolute_value(x):  
    if x < 0:  
        return -x  
    return x  
  
result = absolute_value(-9)  
print(result, absolute_value(5))
```

Answer

-9 5

Status : Wrong

Marks : 0/1

5. What is the output of the code shown?

```
def f():  
    global a  
    print(a)  
    a = "hello"  
    print(a)  
    a = "world"  
f()  
print(a)
```

Answer

worldhellohello

Status : Correct

Marks : 1/1

6. What keyword is used to define a lambda function in Python?

Answer

lambda

Status : Correct

Marks : 1/1

7. What is the output of the following code snippet?

```
def square(x):  
    return x ** 2
```

```
result = square(4)  
print(result)
```

Answer

16

Status : Correct

Marks : 1/1

8. What will be the output of the following code?

```
number = 7  
result = abs(number) + pow(number, 2)  
print(result)
```

Answer

56

Status : Correct

Marks : 1/1

9. What will be the output of the following Python code?

```
def func(a, b=5, c=10):  
    print('a is', a, 'and b is', b, 'and c is', c)
```

```
func(3, 7)  
func(25, c = 24)  
func(c = 50, a = 100)
```

Answer

a is 3 and b is 7 and c is 10
a is 25 and b is 5 and c is 24
a is 100 and b is 5 and c is 50

Status : Correct

Marks : 1/1

10. What will be the output of the following code?

```
num1 = 10  
num2 = -10  
result = abs(num1) + abs(num2)  
print(result)
```

Answer

20

Status : Correct

Marks : 1/1

11. What will be the output of the following Python code?

```
def C2F(c):  
    return c * 9/5 + 32  
print(C2F(100))  
print(C2F(0))
```

Answer

212.032.0

Status : Correct

Marks : 1/1

12. Which of the following functions can take a lambda function as a

parameter in Python?

Answer

map()

Status : Correct

Marks : 1/1

13. What is the output of the following code?

```
x=12
def f1(a,b=x):
    print(a,b)
x=15
f1(4)
```

Answer

4 12

Status : Correct

Marks : 1/1

14. What will be the output of the following Python code?

```
def maximum(x, y):
    if x > y:
        return x
    elif x == y:
        return 'The numbers are equal'
    else:
        return y
```

```
print(maximum(2, 3))
```

Answer

3

Status : Correct

Marks : 1/1

15. What will be the output of the following Python code?

```
def is_even(number):  
    if number % 2 == 0:  
        return True
```

```
result = is_even(6)  
print(result)
```

Answer

True

Status : Correct

Marks : 1/1

Rajalakshmi Engineering College

Name: Ashwin C
Email: 240701058@rajalakshmi.edu.in
Roll no:
Phone: 9940303740
Branch: REC
Department: I CSE FA
Batch: 2028
Degree: B.E - CSE

Scan to verify results



NeoColab_REC_CS23221_Python Programming

REC_Python_Week 4_COD_Updated

Attempt : 1
Total Mark : 50
Marks Obtained : 50

Section 1 : Coding

1. Problem Statement

Implement a program that needs to identify Armstrong numbers. Armstrong numbers are special numbers that are equal to the sum of their digits, each raised to the power of the number of digits in the number.

Write a function `is_armstrong_number(number)` that checks if a given number is an Armstrong number or not.

Function Signature: `armstrong_number(number)`

Input Format

The first line of the input consists of a single integer, `n`, representing the number to be checked.

Output Format

The output should consist of a single line that displays a message indicating whether the input number is an Armstrong number or not.

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: 153

Output: 153 is an Armstrong number.

Answer

```
def armstrong(n):
    i=n
    c=0
    while(i>0):
        i=i//10
        c+=1
    i=n
    sum1=0
    while(i>0):
        rem=i%10
        sum1+=pow(rem,c)
        i=i//10
    if(sum1==n):
        print(f"{n} is an Armstrong number.")
    else:
        print(f"{n} is not an Armstrong number.")
n=int(input())
armstrong(n)
```

Status : Correct

Marks : 10/10

2. Problem Statement

Imagine you are building a messaging application, and you want to know the length of the messages sent by the users. You need to create a program that calculates the length of a message using the built-in function `len()`.

Input Format

The input consists of a string representing the message.

Output Format

The output prints an integer representing the length of the entered message.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: hello!!

Output: 7

Answer

```
# You are using Python
n=input()
print(len(n))
```

Status : Correct

Marks : 10/10

3. Problem Statement

Sara is developing a text-processing tool that checks if a given string starts with a specific character or substring. She needs to implement a function that accepts a string and a character (or substring), and returns True if the string starts with the provided character/substring, or False otherwise.

Write a program that uses a lambda function to help Sara perform this check.

Input Format

The first line contains a string `str` representing the main string to be checked.

The second line contains a string `n`, which is the character or substring to check if the main string starts with it.

Output Format

The first line of output prints "True" if the string starts with the given character/substring, otherwise prints "False".

Refer to the sample for the formatting specifications.

Sample Test Case

Input: Examly

e

Output: False

Answer

```
# You are using Python
str1=input()
str2=input()
n=len(str2)
x=lambda str1,str2,n: "True" if str2 in str1[:n] else "False"
print(x(str1,str2,n))
```

Status : Correct

Marks : 10/10

4. Problem Statement

Imagine you are developing a text analysis tool for a cybersecurity company. Your task is to create a function that analyzes input strings to categorize and count the characters into four categories: uppercase letters, lowercase letters, digits, and special characters. The company needs this tool to process log files and identify potential security threats.

Function Signature: analyze_string(input_string)

Input Format

The input consists of a single string (without space), which may include uppercase letters, lowercase letters, digits, and special characters.

Output Format

The first line contains an integer representing the count of uppercase letters in

the format "Uppercase letters: [count]".

The second line contains an integer representing the count of lowercase letters in the format "Lowercase letters: [count]".

The third line contains an integer representing the count of digits in the format "Digits: [count]".

The fourth line contains an integer representing the count of special characters in the format "Special characters: [count]".

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: Hello123

Output: Uppercase letters: 1

Lowercase letters: 4

Digits: 3

Special characters: 0

Answer

```
def analyze_string(input_string):  
    uppercase_count=0  
    lowercase_count=0  
    digit_count=0  
    special_count=0  
    for char in input_string:  
        if char.isupper():  
            uppercase_count+=1  
        elif char.islower():  
            lowercase_count+=1  
        elif char.isdigit():  
            digit_count+=1  
        else:  
            special_count+=1  
    return uppercase_count,lowercase_count,digit_count,special_count  
  
input_string = input()  
uppercase_count, lowercase_count, digit_count, special_count =  
analyze_string(input_string)
```

```
print("Uppercase letters:", uppercase_count)
print("Lowercase letters:", lowercase_count)
print("Digits:", digit_count)
print("Special characters:", special_count)
```

Status : Correct

Marks : 10/10

5. Problem Statement

Sneha is building a more advanced exponential calculator. She wants to implement a program that does the following:

Calculates the result of raising a given base to a specific exponent using Python's built-in `pow()` function. Displays all intermediate powers from base^1 to $\text{base}^{\text{exponent}}$ as a list. Calculates and displays the sum of these intermediate powers.

Help her build this program to automate her calculations.

Input Format

The input consists of line-separated two integer values representing base and exponent.

Output Format

The first line of the output prints the calculated result of raising the base to the exponent.

The second line prints a list of all powers from base^1 to $\text{base}^{\text{exponent}}$.

The third line prints the sum of all these powers.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 2
3

Output: 8
[2, 4, 8]
14

Answer

```
# You are using Python
base=int(input())
exponent=int(input())
result=pow(base,exponent)
print (result)
powers=[pow(base,i) for i in range(1,exponent+1)]
print(powers)
sum1=sum(powers)
print(sum1)
```

Status : Correct

Marks : 10/10

Rajalakshmi Engineering College

Name: Ashwin C
Email: 240701058@rajalakshmi.edu.in
Roll no:
Phone: 9940303740
Branch: REC
Department: I CSE FA
Batch: 2028
Degree: B.E - CSE

Scan to verify results



NeoColab_REC_CS23221_Python Programming

REC_Python_Week 4_CY

Attempt : 1
Total Mark : 40
Marks Obtained : 40

Section 1 : Coding

1. Problem Statement

Arjun is working on a mathematical tool to manipulate lists of numbers. He needs a program that reads a list of integers and generates two lists: one containing the squares of the input numbers, and another containing the cubes. Arjun wants to use lambda functions for both tasks.

Write a program that computes the square and cube of each number in the input list using lambda functions.

Input Format

The input consists of a single line of space-separated integers representing the list of input numbers.

Output Format

The first line contains a list of the squared values of the input numbers.

The second line contains a list of the cubed values of the input numbers.

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: 1 2 3

Output: [1, 4, 9]

[1, 8, 27]

Answer

```
# You are using Python
numbers=list(map(int, input().split()))
squares=list(map(lambda x: x**2,numbers))
cubes=list(map(lambda x: x**3,numbers))
print(f"{squares}")
print(f"{cubes}")
```

Status : Correct

Marks : 10/10

2. Problem Statement

Implement a program for a retail store that needs to find the highest even price in a list of product prices. Your goal is to efficiently determine the maximum even price from a series of product prices. Utilize the `max()` inbuilt function in the program.

For example, if the prices are 10 15 24 8 37 16, the even prices are 10 24 8 16. So, the maximum even price is 24.

Input Format

The input consists of a series of product prices separated by a space.

The prices should be entered as a space-separated string of numbers.

Output Format

If there are even prices in the input, the output prints "The maximum even price is: " followed by the maximum even price.

If there are no even prices in the input, the output prints "No even prices were found".

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 10 15 24 8 37 16

Output: The maximum even price is: 24

Answer

```
prices=list(map(int, input().split()))
evp=[price for price in prices if price%2==0]
if evp:
    hep=max(evp)
    print(f"The maximum even price is: {hep}")
else:
    print("No even prices were found")
```

Status : Correct

Marks : 10/10

3. Problem Statement

Amrita is developing a password strength checker for her website. She wants the checker to consider the length and the diversity of characters used in the password. A strong password should be long and include a mix of character types: uppercase, lowercase, digits, and special symbols.

She also wants the feedback to be user-friendly, so she wants to include the actual password in the output. Help Amrita finish this password checker using Python's built-in string methods.

Character Types Considered:

Lowercase letters (a-z) Uppercase letters (A-Z) Digits (0-9) Special

characters (from string.punctuation, e.g. @, !, #, \$)

Input Format

The input consists of a single string representing the user's password.

Output Format

The program prints the strength of the password in this format:

If the password length < 6 characters or fewer than 2 of the 4 character types, the output prints "<password> is Weak"

If password length ≥ 6 and at least 2 different character types, the output prints "<password> is Moderate"

If Password length ≥ 10 and all 4 character types present, the output prints "<password> is Strong"

Refer to the sample output for formatting specifications.

Sample Test Case

Input: password123

Output: password123 is Moderate

Answer

```
n=input()
d=0
u=0
l=0
s=0
for i in n:
    if i.isdigit():
        d=1
    elif i.isupper():
        u=1
    elif i.islower():
        l=1
    else:
        s=1
```

```
types=d+s+u+l
if(len(n)>=10 and types==4):
    print(f"{n} is Strong")
elif(len(n)>=6 and types>=2):
    print(f"{n} is Moderate")
else:
    print(f"{n} is Weak")
```

Status : Correct

Marks : 10/10

4. Problem Statement

Imagine you are tasked with developing a function for calculating the total cost of an item after applying a sales tax. The sales tax rate is equal to 0.08 and it is defined as a global variable.

The function should accept the cost of the item as a parameter, calculate the tax amount, and return the total cost.

Additionally, the program should display the item cost, sales tax rate, and total cost to the user.

Function Signature: `total_cost(item_cost)`

Input Format

The input consists of a single line containing a positive floating-point number representing the cost of the item.

Output Format

The output consists of three lines:

"Item Cost:" followed by the cost of the item formatted to two decimal places.

"Sales Tax Rate:" followed by the sales tax rate in percentage.

"Total Cost:" followed by the calculated total cost after applying the sales tax, formatted to two decimal places.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 50.00

Output: Item Cost: \$50.00

Sales Tax Rate: 8.0%

Total Cost: \$54.00

Answer

#

```
SALES_TAX_RATE=0.08
```

```
def total_cost(item_cost):
```

```
    tax=item_cost*SALES_TAX_RATE
```

```
    total_cost=tax+item_cost
```

```
    return total_cost
```

```
item_cost=float(input())
```

```
total_cost = total_cost(item_cost)
```

```
print(f"Item Cost: ${item_cost:.2f}")
```

```
print(f"Sales Tax Rate: {SALES_TAX_RATE * 100}%")
```

```
print(f"Total Cost: ${total_cost:.2f}")
```

Status : Correct

Marks : 10/10

Rajalakshmi Engineering College

Name: Ashwin C
Email: 240701058@rajalakshmi.edu.in
Roll no:
Phone: 9940303740
Branch: REC
Department: I CSE FA
Batch: 2028
Degree: B.E - CSE

Scan to verify results



NeoColab_REC_CS23221_Python Programming

REC_Python_Week 4_PAH_Updated

Attempt : 1
Total Mark : 60
Marks Obtained : 60

Section 1 : Coding

1. Problem Statement

Ella is designing a messaging application that needs to handle long text messages efficiently. To optimize storage and transmission, she plans to implement a text compression feature that replaces consecutive repeated characters with the character followed by its count, while leaving non-repeated characters unchanged.

Help Ella create a recursive function to achieve this compression without altering the original message's meaning.

Function Specification: `def compress_string(*args)`

Input Format

The input consists of a single line containing the string to be compressed.

Output Format

The output consists of a single line containing the compressed string.

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: aaaBBBccc

Output: a3B3c3

Answer

You are using Python

```
def compress_string(*args):
    message=args[0]

    def helper(index,current_char,count,result):
        if index==len(message):
            if count>1:
                result+=current_char+str(count)
            else:
                result+=current_char
            return result
        if message[index]==current_char:
            return helper(index+1,current_char,count+1,result)
        else:
            if count>1:
                result+=current_char+str(count)
            else:
                result+=current_char
            return helper(index+1,message[index],1,result)
    if not message:
        return ""
    return helper(1,message[0],1,"")
input_string=input()
compressed=compress_string(input_string)
print(f"{compressed}")
```

Status : Correct

Marks : 10/10

2. Problem Statement

Create a Python program to monitor temperatures in a greenhouse using two sensors. Calculate and display the absolute temperature difference between the two sensor readings to ensure proper temperature control.

Note: Use the `abs()` built-in function.

Input Format

The first line consists of a floating-point number, representing the temperature reading from Sensor 1.

The second line consists of a floating-point number, representing the temperature reading from Sensor 2.

Output Format

The output displays the absolute temperature difference between Sensor 1 and Sensor 2, rounded to two decimal places.

Refer to the sample output for the exact format.

Sample Test Case

Input: 33.2

26.7

Output: Temperature difference: 6.50 °C

Answer

```
def abss(a,b):  
    result=a-b  
    return abs(result)  
a=float(input())  
b=float(input())  
result=abss(a,b)  
print(f"Temperature difference: {result:.2f} {chr(176)}C")
```

Status : Correct

Marks : 10/10

3. Problem Statement

Alice works at a digital marketing company, where she analyzes large datasets. One day, she's tasked with processing customer ID numbers, which are long numeric sequences.

To simplify her task, Alice needs to calculate the digital root of each ID. The digital root is obtained by repeatedly summing the digits of a number until a single digit remains.

Help Alice write a program that reads a customer ID number, calculates its digital root, and prints the result using a loop-based approach.

For example, the sum of the digits of 98675 is $9 + 8 + 6 + 7 + 5 = 35$, then $3 + 5 = 8$, which is the digital root.

Function prototype: `def digital_root(num)`

Input Format

The input consists of an integer num.

Output Format

The output prints an integer representing the sum of digits for a given number until a single digit is obtained.

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: 451110

Output: 3

Answer

```
num = int(input())
def digital_root(num):
    while(num>=10):
        sod=0
```



```
while num>0:
    sod+=num%10
    num//=10
num=sod
return num
print(num)
result=digital_root(num)
print(digital_root(num))
```

Status : Correct

Marks : 10/10

4. Problem Statement

Sophia is developing a feature for her online banking application that calculates the total sum of digits in customers' account numbers. This sum is used to generate unique verification codes for secure transactions. She needs a program that takes an account number as input and outputs the sum of its digits.

Help Sophia to complete her task.

Function Specification: def sum_digits(num)

Input Format

The input consists of an integer, representing the customer's account number.

Output Format

The output prints an integer representing the sum of the digits of the account number.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 123245

Output: 17

Answer

```
num = int(input())
def sum_digits(num):
    total=0
    while(num>0):
        total+=num%10
        num//=10
    return total
sum = sum_digits(num)
print(sum)
```

Status : Correct**Marks : 10/10****5. Problem Statement**

Ravi is working on analyzing a set of integers to determine how many of them are divisible by 3 and how many are divisible by 5. He decides to use lambda functions to filter and count the numbers based on their divisibility.

Write a program that takes a list of integers, calculates how many numbers are divisible by 3, and how many are divisible by 5, and then prints the results.

Additionally, the program should calculate the total sum of all numbers divisible by 3 and divisible by 5 separately.

Input Format

The first line contains an integer n , representing the number of integers in the list.

The second line contains n space-separated integers.

Output Format

The first line should print the count of numbers divisible by 3.

The second line should print the count of numbers divisible by 5.

The third line should print the sum of numbers divisible by 3.

The fourth line should print the sum of numbers divisible by 5.

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: 6
3 5 6 10 15 20
Output: 3
4
24
50

Answer

```
# You are using Python
n=int(input())
num=list(map(int, input().split()))
divby3=list(filter(lambda x: x%3==0,num))
divby5=list(filter(lambda x: x%5==0,num))

countdivby3=len(divby3)
sumdivby3=sum(divby3)

countdivby5=len(divby5)
sumdivby5=sum(divby5)

print(countdivby3)
print(countdivby5)
print(sumdivby3)
print(sumdivby5)
```

Status : Correct

Marks : 10/10

6. Problem Statement

Hussain wants to create a program to calculate a person's BMI (Body Mass Index) based on their weight in kilograms and height in meters. The

BMI is a measure of a person's body fat relative to their height.

Your program should take user input for weight and height, calculate the BMI, and display the result.

Function Signature: `calculate_bmi(weight, height)`

Formula: $BMI = \text{Weight} / (\text{Height})^2$

Input Format

The first line of input consists of a positive floating-point number, the person's weight in kilograms.

The second line of input consists of a positive floating-point number, the person's height in meters.

Output Format

The output displays "Your BMI is: [BM]" followed by a float value representing the calculated BMI, rounded off two decimal points.

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: 70.0

1.75

Output: Your BMI is: 22.86

Answer

```
weight = float(input())
height = float(input())

def calculate_bmi(weight,height):
    BMI=weight/(height**2)
    print(f"Your BMI is: {BMI:.2f}")

calculate_bmi(weight, height)
```

Status : Correct

Marks : 10/10

Rajalakshmi Engineering College

Name: Ashwin C
Email: 240701058@rajalakshmi.edu.in
Roll no:
Phone: 9940303740
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_MCQ

Attempt : 1
Total Mark : 20
Marks Obtained : 19

Section 1 : MCQ

1. Which of the following isn't true about dictionary keys?

Answer

Keys must be integers

Status : Correct

Marks : 1/1

2. What is the output of the below Python code?

```
list1 = [1, 2, 3]
list2 = [5, 6, 7]
list3 = [10, 11, 12]
set1 = set(list2)
set2 = set(list1)
set1.update(set2)
```

```
set1.update(list3)
print(set1)
```

Answer

```
{1, 2, 3, 5, 6, 7, 10, 11, 12}
```

Status : Correct

Marks : 1/1

3. Suppose $t = (1, 2, 4, 3)$, which of the following is incorrect?

Answer

```
t[3] = 45
```

Status : Correct

Marks : 1/1

4. What will be the output for the following code?

```
a=(1,2,3)
b=('A','B','C')
c=zip(a,b)
```

```
print(c)
print(tuple(c))
```

Answer

```
((1, 'A'), (2, 'B'), (3, 'C'))
```

Status : Correct

Marks : 1/1

5. Which of the following statements is used to create an empty tuple?

Answer

```
()
```

Status : Correct

Marks : 1/1

6. Fill in the code in order to get the following output.

Output:

Tuple: (1, 3, 4)

Max value: 4

t=(1,)

```
_____  
print("Tuple:" ,t)  
print("Max value:",_____)
```

Answer

1) t=t+(3,4)2) max(t)

Status : Correct

Marks : 1/1

7. Set $s1 = \{1, 2, 4, 3\}$ and $s2 = \{1, 5, 4, 6\}$, find $s1 \& s2$, $s1 - s2$, $s1 \mid s2$ and $s1 \wedge s2$.

Answer

$s1 \& s2 = \{1, 4\}$ $s1 - s2 = \{2, 3\}$ $s1 \wedge s2 = \{2, 3, 5, 6\}$ $s1 \mid s2 = \{1, 2, 3, 4, 5, 6\}$

Status : Correct

Marks : 1/1

8. What is the output of the following code?

```
a={"a":1,"b":2,"c":3}  
b=dict(zip(a.values(),a.keys()))  
print(b)
```

Answer

{1: 'a', 2: 'b', 3: 'c'}

Status : Correct

Marks : 1/1

9. Which of the statements about dictionary values is false?

Answer

Values of a dictionary must be unique

Status : Correct

Marks : 1/1

10. What is the output of the following code?

```
a=(1,2,(4,5))  
b=(1,2,(3,4))  
print(a<b)
```

Answer

False

Status : Correct

Marks : 1/1

11. What is the output of the following code?

```
a={1:"A",2:"B",3:"C"}  
b=a.copy()  
b[2]="D"  
print(a)
```

Answer

{1: 'A', 2: 'B', 3: 'C'}

Status : Correct

Marks : 1/1

12. What is the output of the following?

```
set1 = {10, 20, 30, 40, 50}  
set2 = {60, 70, 10, 30, 40, 80, 20, 50}  
print(set1.issubset(set2))  
print(set2.issuperset(set1))
```

Answer

TrueTrue

Status : Correct

Marks : 1/1

13. Predict the output of the following Python program

```
init_tuple_a = 1, 2, 8
init_tuple_b = (1, 2, 7)
set1=set(init_tuple_b)
set2=set(init_tuple_a)
print (set1 | set2)
print (init_tuple_a | init_tuple_b)
```

Answer

{1, 2, 7, 8}TypeError: unsupported operand type

Status : Correct

Marks : 1/1

14. If 'a' is a dictionary with some key-value pairs, what does a.popitem() do?

Answer

Removes the key-value pair for the key given as an argument

Status : Wrong

Marks : 0/1

15. What will be the output of the following code?

```
a=(1,2,3,4)
print(sum(a,3))
```

Answer

13

Status : Correct

Marks : 1/1

16. What is the result of print(type({}) is set)?

Answer

False

Status : Correct

Marks : 1/1

17. Which of the following is a Python tuple?

Answer

(1, 2, 3)

Status : Correct

Marks : 1/1

18. What will be the output?

```
a={'B':5,'A':9,'C':7}  
print(sorted(a))
```

Answer

['A', 'B', 'C'].

Status : Correct

Marks : 1/1

19. What will be the output of the following program?

```
set1 = {1, 2, 3}  
set2 = set1.copy()  
set2.add(4)  
print(set1)
```

Answer

{1, 2, 3}

Status : Correct

Marks : 1/1

20. What will be the output for the following code?

```
t1 = (1, 2, 4, 3)  
t2 = (1, 2, 3, 4)  
print(t1 < t2)
```

Answer

False

Status : Correct

Marks : 1/1

Rajalakshmi Engineering College

Name: Ashwin C
Email: 240701058@rajalakshmi.edu.in
Roll no:
Phone: 9940303740
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_COD

Attempt : 1
Total Mark : 50
Marks Obtained : 50

Section 1 : Coding

1. Problem Statement

James is managing a list of inventory items in a warehouse. Each item is recorded as a tuple, where the first element is the item ID and the second element is a list of quantities available for that item. James needs to filter out all quantities that are above a certain threshold to find items that have a stock level above this limit.

Help James by writing a program to process these tuples, filter the quantities from all the available items, and display the results.

Note:

Use the filter() function to filter out the quantities greater than the specified threshold for each item's stock list.

Input Format

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

The next N lines each contain a tuple in the format (ID, [quantity1, quantity2, ...]), where ID is an integer and the list contains integers.

The final line consists of an integer threshold, representing the quantity threshold.

Output Format

The output should be a single line displaying the filtered quantities, space-separated. Each quantity is strictly greater than the given threshold.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 2
(1, [1, 2])
(2, [3, 4])
2

Output: 3 4

Answer

```
# You are using Python
n=int(input())
data=[]
for i in range(n):
    tup=eval(input())
    data.append(tup)
k=int(input())
result=[]
for item in data:
    q=item[1]
    f=filter(lambda x:x>k,q)
    result.extend(f)
print(*result)
```

Status : Correct

Marks : 10/10

2. Problem Statement

Gowshik is working on a task that involves taking two lists of integers as input, finding the element-wise sum of the corresponding elements, and then creating a tuple containing the sum values.

Write a program to help Gowshik with this task.

Example:

Given list:

[1, 2, 3, 4]

[3, 5, 2, 1]

An element-wise sum of the said tuples: (4, 7, 5, 5)

Input Format

The first line of input consists of a single integer n , representing the length of the input lists.

The second line of input consists of n integers separated by commas, representing the elements of the first list.

The third line of input consists of n integers separated by commas, representing the elements of the second list.

Output Format

The output is a single line containing a tuple of integers separated by commas, representing the element-wise sum of the corresponding elements from the two input lists.

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: 4

1, 2, 3, 4

3, 5, 2, 1

Output: (4, 7, 5, 5)

Answer

```
# You are using Python
n=int(input())
s1=tuple(map(int,input().split(',')))
s2=tuple(map(int,input().split(',')))
s3=tuple(s1[i]+s2[i] for i in range(n))
print(s3)
```

Status : Correct

Marks : 10/10

3. Problem Statement

Ella is analyzing the sales data for a new online shopping platform. She has a record of customer transactions where each customer's data includes their ID and a list of amounts spent on different items. Ella needs to determine the total amount spent by each customer and identify the highest single expenditure for each customer.

Your task is to write a program that computes these details and displays them in a dictionary.

Input Format

The first line of input consists of an integer n , representing the number of customers.

Each of the next n lines contains a numerical customer ID followed by integers representing the amounts spent on different items.

Output Format

The output displays a dictionary where the keys are customer IDs and the values are lists containing two integers: the total expenditure and the maximum single expenditure.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 2

101 100 150 200

102 50 75 100

Output: {101: [450, 200], 102: [225, 100]}

Answer

You are using Python

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

```
d={}
```

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

```
    s=input().split()
```

```
    l=list(map(int,s))
```

```
    key=int(l[0])
```

```
    l.remove(key)
```

```
    tot=sum(l)
```

```
    m=max(l)
```

```
    d[key]=[tot,m]
```

```
print(d)
```

Status : Correct

Marks : 10/10

4. Problem Statement

Liam is analyzing a list of product IDs from a recent sales report. He needs to determine how frequently each product ID appears and calculate the following metrics:

Frequency of each product ID: A dictionary where the key is the product ID and the value is the number of times it appears. Total number of unique product IDs. Average frequency of product IDs: The average count of all product IDs.

Write a program to read the product IDs, compute these metrics, and output the results.

Example

Input:

6 //number of product ID

101

102

101

103

101

102 //product IDs

Output:

{101: 3, 102: 2, 103: 1}

Total Unique IDs: 3

Average Frequency: 2.00

Explanation:

Input 6 indicates that you will enter 6 product IDs.

A dictionary is created to track the frequency of each product ID.

Input 101: Added with a frequency of 1.

Input 102: Added with a frequency of 1.

Input 101: Frequency of 101 increased to 2.

Input 103: Added with a frequency of 1.

Input 101: Frequency of 101 increased to 3.

Input 102: Frequency of 102 increased to 2.

The dictionary now contains 3 unique IDs: 101, 102, and 103.

Total Unique is 3.

The average frequency is 2.00.

Input Format

The first line of input consists of an integer n , representing the number of product IDs.

The next n lines each contain a single integer, each representing a product ID.

Output Format

The first line of output displays the frequency dictionary, which maps each product ID to its count.

The second line displays the total number of unique product IDs, preceded by "Total Unique IDs: ".

The third line displays the average frequency of the product IDs. This is calculated by dividing the total number of occurrences of all product IDs by the total number of unique product IDs, rounded to two decimal places. It is preceded by "Average Frequency: ".

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 6

101

102

101

103

101

102

Output: {101: 3, 102: 2, 103: 1}

Total Unique IDs: 3

Average Frequency: 2.00

Answer

You are using Python

```
d={}
```

```
l1=[]
```

```
l2=s=0
```

```
a=0.0
```

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

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

```
e=int(input())
l1.append(e)
for i in l1:
    if i not in d:
        d1=dict([(i,l1.count(i))])
        d.update(d1)
print(d)
l2=len(d)
print(f"Total Unique IDs: {l2}")
s=sum(d.values())
a=s/l2
print(f"Average Frequency: {a:.2f}")
```

Status : Correct

Marks : 10/10

5. Problem Statement

Professor Adams needs to analyze student participation in three recent academic workshops. She has three sets of student IDs: the first set contains students who registered for the workshops, the second set contains students who actually attended, and the third set contains students who dropped out.

Professor Adams needs to determine which students who registered also attended, and then identify which of these students did not drop out.

Help Professor Adams identify the students who registered, attended, and did not drop out of the workshops.

Input Format

The first line of input consists of integers, representing the student IDs who registered for the workshops.

The second line consists of integers, representing the student IDs who attended the workshops.

The third line consists of integers, representing the student IDs who dropped out of the workshops.

Output Format

The first line of output displays the intersection of the first two sets, which shows the IDs of students who registered and attended.

The second line displays the result after removing student IDs that are in the third set (dropped out), showing the IDs of students who both attended and did not drop out.

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: 1 2 3

2 3 4

3 4 5

Output: {2, 3}

{2}

Answer

You are using Python

```
r=input().split()
```

```
s1=set(map(int,r))
```

```
a=input().split()
```

```
s2=set(map(int,a))
```

```
d=input().split()
```

```
s3=set(map(int,d))
```

```
s4=s1&s2
```

```
s5=s4-s3
```

```
print(s4)
```

```
print(s5)
```

Status : Correct

Marks : 10/10

Rajalakshmi Engineering College

Name: Ashwin C
Email: 240701058@rajalakshmi.edu.in
Roll no:
Phone: 9940303740
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 : 37.5

Section 1 : Coding

1. 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

```
# You are using Python
w1=input().strip().lower()
w2=input().strip().lower()
set1=set(w1)
set2=set(w2)
common=sorted(set1&set2)
unique=sorted(set1^set2)
superset1=set1.issuperset(set2)
disjoint=set1.isdisjoint(set2)
```

```
print(common)
print(unique)
print(superset1)
print(disjoint)
```

Status : Correct

Marks : 10/10

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

```
# You are using Python
n=int(input())
pixel=list(map(int, input().split()))
diff=tuple(abs(pixel[i+1]-pixel[i]) for i in range (0,n-1))
print(diff)
```

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

```
import math
```

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

```
a,b,c=(map(int, input().split()))
```

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

```
if(dis>0):
```

```
    root1=(-b+(math.sqrt(dis)))/(2*a)
```

```
    root2=(-b-(math.sqrt(dis)))/(2*a)
```

```
    print((root1, root2))
```

```
elif(dis==0):
```

```
    root=(-b)/(2*a)
```

```
    print((root,))
```

```
else:
```

```
    realpart=-b/(2*a)
```

```
    imapart=math.sqrt(-dis)/(2*a)
```

```
    r1=(realpart,imapart)
```

```
    r2=(realpart, -imapart)
```

```
    print((r1,r2))
```

Status : Partially correct

Marks : 7.5/10

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

You are using Python

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

```
maxi=-1
```

```
maxc=""
```

```
for _ in range(n):
```

```
    code=input().strip()
```

```
    pop1=int(input())
```

```
pop2=int(input())
inc=pop2-pop1
if inc>maxi:
    maxi=inc
    maxc=code
print(f"{{{maxc}:{maxi}}}")
```

Status : Correct

Marks : 10/10

Rajalakshmi Engineering College

Name: Ashwin C
Email: 240701058@rajalakshmi.edu.in
Roll no:
Phone: 9940303740
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_PAH

Attempt : 1
Total Mark : 60
Marks Obtained : 60

Section 1 : Coding

1. Problem Statement

Rishi is working on a program to manipulate a set of integers. The program should allow users to perform the following operations:

Find the maximum value in the set. Find the minimum value in the set. Remove a specific number from the set.

The program should handle these operations based on user input. If the user inputs an invalid operation choice, the program should indicate that the choice is invalid.

Input Format

The first line contains space-separated integers that will form the initial set. Each integer x is separated by a space.

The second line contains an integer `ch`, representing the user's choice:

- 1 to find the maximum value
- 2 to find the minimum value
- 3 to remove a specific number from the set

If `ch` is 3, the third line contains an integer `n1`, which is the number to be removed from the set.

Output Format

The first line of output prints the original set in descending order.

For choice 1: Print the maximum value from the set.

For choice 2: Print the minimum value from the set.

For choice 3: Print the set after removing the specified number, in descending order.

For invalid choices: Print "Invalid choice".

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: 1 2 3 4 5

1

Output: {5, 4, 3, 2, 1}

5

Answer

```
# You are using Python
inset=list(map(int,input().split()))
ch=int(input())
orset=sorted(inset,reverse=True)
print(f'{{{', '.join(map(str,orset))}}}')
if ch==1:
    print(max(inset))
elif ch==2:
    print(min(inset))
```

```
elif(ch==3):
    n1=int(input())
    if n1 in inset:
        inset.remove(n1)
        upset=sorted(inset,reverse=True)
        print(f"{{{', '.join(map(str, upset))}}}")
    else:
        print(f"{{{', '.join(map(str, orset))}}}")
else:
    print("Invalid choice")
```

Status : Correct

Marks : 10/10

2. Problem Statement

Maya wants to create a dictionary that maps each integer from 1 to a given number n to its square. She will use this dictionary to quickly reference the square of any number up to n .

Help Maya generate this dictionary based on the input she provides.

Input Format

The input consists of an integer n , representing the highest number for which Maya wants to calculate the square.

Output Format

The output displays the generated dictionary where each key is an integer from 1 to n , and the corresponding value is its square.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 5

Output: {1: 1, 2: 4, 3: 9, 4: 16, 5: 25}

Answer

You are using Python

```
n=int(input())
sqdict={i:i**2 for i in range(1,n+1)}
print(sqdict)
```

Status : Correct

Marks : 10/10

3. Problem Statement

Jordan is creating a program to process a list of integers. The program should take a list of integers as input, remove any duplicate integers while preserving their original order, concatenate the remaining unique integers into a single string, and then print the result.

Help Jordan in implementing the same.

Input Format

The input consists of space-separated integers representing the elements of the set.

Output Format

The output prints a single integer formed by concatenating the unique integers from the input in the order they appeared.

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: 11 11 33 50

Output: 113350

Answer

```
lstin=input().split()
ulst=[]
for num in lstin:
    if num not in ulst:
        ulst.append(num)
result="".join(ulst)
```

```
print(result)
```

Status : Correct

Marks : 10/10

4. Problem Statement

Tom wants to create a dictionary that lists the first n prime numbers, where each key represents the position of the prime number, and the value is the prime number itself.

Help Tom generate this dictionary based on the input she provides.

Input Format

The input consists of an integer n, representing the number of prime numbers Tom wants to generate.

Output Format

The output displays the generated dictionary where each key is an integer from 1 to n, and the corresponding value is the prime number.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 4

Output: {1: 2, 2: 3, 3: 5, 4: 7}

Answer

```
# You are using Python
n=int(input())
d1={}
k=1
i=2
while(True):
    f=1
    for j in range(2,i):
        if (i%j==0):
            f=0
```

```
        break
    if (f==1):
        d1[k]=i
        k+=1
    if k==n+1:
        break
    i+=1
print(dict(d1))
```

Status : Correct

Marks : 10/10

5. Problem Statement

Mia is organizing a list of integers into a series of pairs for his new project. She wants to create pairs of consecutive integers from the list. The last integer should be paired with None to complete the series. The pairing happens as follows: ((Element 1, Element 2), (Element 2, Element 3)..... (Element n, None)).

Your task is to help Henry by writing a Python program that reads a list of integers, forms these pairs, and displays the result in tuple format.

Input Format

The first line of input consists of an integer n, representing the number of elements in the tuple.

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

Output Format

The output displays a tuple containing pairs of consecutive integers from the input. The last integer in the tuple is paired with 'None'.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 3

5 10 15

Output: ((5, 10), (10, 15), (15, None))

Answer

You are using Python

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

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

```
pairs=tuple((el[i],el[i+1])for i in range(n-1))+((el[-1], None),)
```

```
print(pairs)
```

Status : Correct

Marks : 10/10

6. Problem Statement

Sophia is organizing a list of event IDs representing consecutive days of an event. She needs to group these IDs into consecutive sequences. For example, if the IDs 3, 4, and 5 appear consecutively, they should be grouped.

Write a program that helps Sophia by reading the total number of event IDs and the IDs themselves, then display each group of consecutive IDs in tuple format.

Input Format

The first line of input consists of an integer n , representing the number of event IDs.

The next n lines contain integers representing the event IDs, where each integer corresponds to an event ID.

Output Format

The output should display each group of consecutive event IDs in a tuple format. Each group should be printed on a new line, and single event IDs should be displayed as a single-element tuple.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 3

1
2
3

Output: (1, 2, 3)

Answer

```
# You are using Python
n=int(input())
evid=[int(input()) for _ in range(n)]
evid.sort()
gr=[]
crgr=[evid[0]]
for i in range(1,n):
    if evid[i]==evid[i-1]+1:
        crgr.append(evid[i])
    else:
        gr.append(tuple(crgr))
        crgr=[evid[i]]
gr.append(tuple(crgr))
for grp in gr:
    if(len(grp)==1):
        print(f'({grp[0]})')
    else:
        print(f'({", ".join(map(str, grp))})')
```

Status : Correct

Marks : 10/10