Rajalakshmi Engineering College

Name: Mohamed Ismail Fawaz K P
Email: 240701324@rajalakshmi.edu.in
Roll no: 240701324
Phone: 9791737785
Branch: REC
Department: I CSE FC
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 will be the output of the following Python code?

```python
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*

2. Which of the following functions can take a lambda function as a parameter in Python?

Answer

All of the mentioned options

Status : Wrong   *Marks : 0/1*

3. 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*

4. 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*

5. 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*

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

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

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

Answer

True

Status : Correct   *Marks : 1/1*

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

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

Answer

56

Status : Correct   *Marks : 1/1*

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

```python
def display(*args):
    for arg in args:
        print(arg)

display(10, 20, 30)
```

Answer

102030

Status : Correct   *Marks : 1/1*

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

```
num = -5
result = abs(num)
print(result)
```

Answer

5

Status : Correct   *Marks : 1/1*

10.   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*

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. How is a lambda function different from a regular named function in Python?

Answer

A lambda function does not have a name, while a regular function does

Status : Correct   *Marks : 1/1*

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

```
def display(b, n):
    while n > 0:
        print(b,end="")
        n=n-1
display('z',3)
```

Answer

zzz

Status : Correct   *Marks : 1/1*

14. 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 10a is 25 and b is 5 and c is 24a is 100 and b is 5 and c is 50

Status : Correct   *Marks : 1/1*

15. 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 : Correct   *Marks : 1/1*

Rajalakshmi Engineering College

Name: Mohamed Ismail Fawaz K P
Email: 240701324@rajalakshmi.edu.in
Roll no: 240701324
Phone: 9791737785
Branch: REC
Department: I CSE FC
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

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

```python
def analyze_string(input_string):
# You are using Python
    #Type your code here
    uppercase_count=0
    lowercase_count=0
    digit_count=0
    special_count=0
    for i in input_string:
        if i.isupper():
            uppercase_count+=1
        elif i.islower():
            lowercase_count+=1
        elif i.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*

2. 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 base^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 base^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

```python
# You are using Python
b=int(input())
e=int(input())
res=pow(b,e)
power=[pow(b,i) for i in range(1,e+1)]
sum1=sum(power)
print(res)
print(power)
print(sum1)
```

Status : Correct   *Marks : 10/10*

3. 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

```python
# You are using Python
def arms(a):
    l=len(a)
    n=int(a)
    sum=0
    for i in a:
        b=int(i)
        sum=sum+(b**l)
    if (sum==n):
        print(f"{a} is an Armstrong number.")
    else:
        print(f"{a} is not an Armstrong number.")
a=input()
arms(a)
```

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

4. 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
start=lambda str1,x:str1[:len(x)]==x
str1=input().strip()
x=input().strip()
result=start(str1,x)
print(result)
```

Status : Correct   *Marks : 10/10*

5. 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

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

Status : Correct   *Marks : 10/10*

Rajalakshmi Engineering College

Name: Mohamed Ismail Fawaz K P
Email: 240701324@rajalakshmi.edu.in
Roll no: 240701324
Phone: 9791737785
Branch: REC
Department: I CSE FC
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

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

```python
# You are using Python
a=float(input())
b=float(input())
res=abs(a-b)
degree='\u00B0'
print(f"Temperature difference: {res:.2f} {degree}C")
```

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

2. 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

```python
# You are using Python
def compress_string(*args):
    s=args[0]
    def helper(i,prev,count,res):
        if i==len(s):
            res+=prev+str(count) if count >1 else prev
            return res
        if s[i]==prev:
            return helper(i+1,prev,count+1,res)
        else:
            res+=prev+str(count) if count >1 else prev
            return helper(i+1,s[i],1,res)
    return helper(1,s[0],1,"") if s else ""
str1=input()
print(compress_string(str1))
```

Status : Correct   *Marks : 10/10*

3. 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

```python
# You are using Python
n=int(input())
l1=list(map(int,input().split()))
div3=list(filter(lambda x:x%3==0,l1))
div5=list(filter(lambda x:x%5==0,l1))
print(len(div3))
print(len(div5))
print(sum(div3))
```

```
print(sum(div5))
```

4. 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())
# You are using Python
def digital_root(num):
    while num>=10:
        num=sum(int(digit) for digit in str(num))
    return num

print(digital_root(num))
```

Status : Correct   *Marks : 10/10*

5. 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 = Weight/(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

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

# You are using Python
def calculate_bmi(weight,height):
    bmi=weight/height**2
    return round(bmi,2)
calculate_bmi(weight,height)
print(f"Your BMI is: ",calculate_bmi(weight,height))

calculate_bmi(weight, height)
```

Status : Correct   *Marks : 10/10*

6. 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())

```python
# You are using Python
def sum_digits(a):
    return sum(int(digits) for digits in str(a))
```

sum = sum_digits(num)
print(sum)

Status : Correct   *Marks : 10/10*

Rajalakshmi Engineering College

Name: Mohamed Ismail Fawaz K P
Email: 240701324@rajalakshmi.edu.in
Roll no: 240701324
Phone: 9791737785
Branch: REC
Department: I CSE FC
Batch: 2028
Degree: B.E - CSE

NeoColab_REC_CS23221_Python Programming

REC_Python_Week 4_CY

Attempt : 1
Total Mark : 40
Marks Obtained : 40

Section 1 : Coding

1. 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

```
#
# You are using Python
def total_cost(item_cost):
    cost=item_cost+((8/100)*item_cost)
    return cost
SALES_TAX_RATE=0.08
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*

2. Problem Statement

Develop a text analysis tool that needs to count the occurrences of a specific substring within a given text string.

Write a function count_substrings(text, substring) that takes two inputs: the text string and the substring to be counted. The function should count how many times the substring appears in the text string and return the count.

Function Signature: count_substrings(text, substring)

Input Format

The first line of the input consists of a string representing the text.

The second line consists of a string representing the substring.

Output Format

The output should display a single line of output containing the count of occurrences of the substring in the text string.

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: programming is fun and programming is cool
programming
Output: The substring 'programming' appears 2 times in the text.

Answer

```
# You are using Python
a=input()
b=input()
n=a.count(b)
print(f"The substring '{b}' appears {n} times in the text. ")
```

Status : Correct   *Marks : 10/10*

3. Problem Statement

Create a program for a mathematics competition where participants need to find the smallest positive divisor of a given integer n. Your program should efficiently determine this divisor using the min() function and display the result.

Input Format

The input consists of a single positive integer n, representing the number for which the smallest positive divisor needs to be found.

Output Format

The output prints the smallest positive divisor of the input integer in the format: "The smallest positive divisor of [n] is: [smallest divisor]".

Refer to the sample output for the exact format.

Sample Test Case

Input: 24
Output: The smallest positive divisor of 24 is: 2

Answer

```python
# You are using Python
a=int(input())
list1=[n for n in range (2,a+1) if a%n==0]
div=min(list1)
print(f"The smallest positive divisor of {a} is: ",div)
```

Status : Correct   *Marks : 10/10*

4. 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

```python
# You are using Python
l1=list(map(int,input().split()))
lis2=list(map(lambda x:x**2,l1))
lis3=list(map(lambda x:x**3,l1))
print(lis2)
print(lis3)
```

Status : Correct   *Marks : 10/10*