Rajalakshmi Engineering College

Name: nila ela
Email: 241901074@rajalakshmi.edu.in
Roll no: 241901074
Phone: 9025155667
Branch: REC
Department: I CSE (CS) FB
Batch: 2028
Degree: B.E - CSE (CS)

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 is the output of the following?

```
i = 2
while True:
  if i%3 == 0:
    break
  print(i)
  i += 2
```

Answer

2 4

Status : Correct ***Marks : 1/1***

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

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

Answer

Error

Status : Correct *Marks : 1/1*

3. 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*

4. 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

2

Status : Wrong ***Marks : 0/1***

5. What is the purpose of the pass statement in Python?

Answer

To do nothing and act as a placeholder.

Status : Correct ***Marks : 1/1***

6. What is the output of the following code?

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

Answer

0 1 2 3 4 Here

Status : Correct ***Marks : 1/1***

7. 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)
   i += 2
```

Answer

1 3 5 7 9

Status : Correct ***Marks : 1/1***

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

```
i = 5
while True:
    if i%0O11 == 0:
        break
    print(i, end = " ")
    i += 1
```

Answer

5 6 7 8 9 10

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 code?

```
i = 1
while True:
    if i%0O7 == 0:
        break
    print(i)
    i += 1
```

Answer

1 2 3 4 5 6

Status : Correct ***Marks : 1/1***

11. 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***

12. 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***

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

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

Answer

0120

Status : Wrong ***Marks : 0/1***

14. What is the output of the following code?

```
i = 5
while True:
    if i%0O9 == 0:
        break
    print(i)
    i += 1
```

Answer

5 6 7 8

Status : Wrong ***Marks : 0/1***

15. 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***

Rajalakshmi Engineering College

Name: nila ela
Email: 241901074@rajalakshmi.edu.in
Roll no: 241901074
Phone: 9025155667
Branch: REC
Department: I CSE (CS) FB
Batch: 2028
Degree: B.E - CSE (CS)

Scan to verify results

NeoColab_REC_CS23221_Python Programming

REC_Python_Week 2_COD_Updated

Attempt : 1
Total Mark : 50
Marks Obtained : 32.5

Section 1 : Coding

1. 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
a=int(input())
b=int(input())
sum=0
for i in range(a,b):
    n=i//10
    m=i%10
    sum+=n+m
print(sum)
```

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

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 count_non_fibo(start,end):
    fib_num=set()
    a,b=0,1
    while a<end:
        fib_num.add(a)
        a,b=b,a+b


        non_fib_count=0
        for num in range(start,end):
            if num not in fib_num:
                non_fib_count += 1

        return non_fib_count

start=int(input())
end=int(input())

print(count_non_fibo(start,end))
```

Status : Partially correct ***Marks : 2.5/10***

3. 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
def extract_consonants(sentence):
    vowels = "aeiouAEIOU"
    consonants=[]

    for char in sentence:
        if char.isalpha() and char not in vowels:
            consonants.append(char)
    print(" ".join(consonants))

sentence = input()

extract_consonants(sentence)
```

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

4. 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
def find_per_num(start,end):
    def is_per_num(n):
        divisor_sum=0
        for i in range(1,n):
            if n%i==0:
                divisor_sum += i
        return divisor_sum == n


    per_num=[]
    for num in range(start,end):
        if is_per_num(num):
            per_num.append(num)

    print(" ".join(map(str,per_num)))

start=int(input())
end=int(input())

find_per_num(start,end)
```

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

5. 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
def print_even_sq(n):

    for i in range(1, n+1):

        if i%2==0:

            print(i**2)

n=int(input())

print_even_sq(n)
```

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

Rajalakshmi Engineering College

Name: nila ela
Email: 241901074@rajalakshmi.edu.in
Roll no: 241901074
Phone: 9025155667
Branch: REC
Department: I CSE (CS) FB
Batch: 2028
Degree: B.E - CSE (CS)

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

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
start=int(input())
end=int(input())
sum_of_numbers=0
for num in range(start,end+1):
   if num%3==0:
      continue
   sum_of_numbers += num
print(sum_of_numbers )
```

__Status :__ Correct *__Marks : 10/10__*

2. 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 calculate_bill(units):
    if units<=100:
        amount=0
    elif units<=200:
        amount=(units-100)*5
    else:
        amount=(100*5)+((units-200)*10)
    print(f"Rs. {amount}")
units=int(input())
calculate_bill(units)
```

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

3. 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

```
# You are using Python
def is_leap_yr(year):
   return(year%4==0 and (year%100 !=0 or year%400==0))
start=int(input())
end=int(input())
for year in range(start,end+1):
   if is_leap_yr(year):
      print(year)
```

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

4. 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
def fac(num):
    result=1
    for i in range(1,num+1):
        result *=i
    return result

def sum_of_digits(number):
    return sum(int(digit) for digit in str(number))
```

```
def odd_factorial_sum(n):
    for num in range(1,n+1):
        if num %2 != 0:
            fact=fac(num)
            digit_sum=sum_of_digits(fact)
            print(f"{num}! = {fact}, sum of digits={digit_sum}")

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

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

5. 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 c_c(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
   if steps<=100:
      print(f"Steps taken to reach 1: {steps}")
try:
   n=int(input())
   if n>0:
      c_c(n)
   else:
      print("Please enter a positive integer.")
except ValueError:
   print("Invalid input. Please enter a positive integer.")
```

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

6. 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 sum_e_o(n):
    e_s=0
    o_s=0
    for num in range(1,n+1):
        if num%2==0:
            e_s += num
        else:
            o_s += num
    print(f"Sum of even numbers from 1 to {n} is {e_s}")
    print(f"Sum of odd numbers from 1 to {n} is {o_s}")
N=int(input())
sum_e_o(N)
```

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

Rajalakshmi Engineering College

Name: nila ela
Email: 241901074@rajalakshmi.edu.in
Roll no: 241901074
Phone: 9025155667
Branch: REC
Department: I CSE (CS) FB
Batch: 2028
Degree: B.E - CSE (CS)

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 is_prime(digit):
   if digit in[2,3,5,7]:
      return True
   return False
def sum_of_non_prime_digits(numbers):
   total=0
   for digit in str(number):
      digit = int(digit)
      if not is_prime(digit):
         total += digit
   return total
number = int(input())
result= sum_of_non_prime_digits(number)
print(result)
```

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

2. Problem Statement

Rohith is a data analyst who needs to categorize countries based on their population growth rates. Each country is assigned a unique code. Rohith will receive a code and corresponding data based on the code. If the data falls within specific thresholds, he needs to classify the country's priority level.

Your task is to write a program that reads a country code and its associated data, and then determines if the priority is "High" or "Low."

Thresholds:France: Priority is "High" if the percentage < 50, else "Low".Japan: Priority is "High" if life expectancy > 80, else "Low".Brazil: Priority is "High" if the urban population > 80, else "Low".

Input Format

The first line of input consists of an integer, representing the country code (1 for France, 2 for Japan, 3 for Brazil).

If the country code is 1,

- The second line consists of a floating-point value N, representing the percentage of the English-speaking population.

If the country code is 2,

- The second line consists of a floating-point value A, representing the average life expectancy in years.

If the country code is 3,

- The second line consists of a floating-point value P, representing the percentage of the urban population.

Output Format

The first line of output displays "Priority: High" or "Priority: Low" based on the input data.

If the country code is invalid, print "Invalid".

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 1
30.0

Output: Priority: High

Answer

```
# You are using Python
def det_priority(code,value):
    if code==1:
        if value<50:
            return "High"
        else:
            return "Low"
    elif code==2:
        if value>80:
            return "High"
        else:
            return "Low"
    elif code==3:
        if value>80:
            return "High"
        else:
            return "Low"
    else:
        return "Invalid country code"
code=int(input())
if code in[1,2,3]:
    value=float(input())
    priority=det_priority(code,value)
    print("Priority:",priority)
else:
    print("Invalid")
```

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 check_sec_code(n):
    str_n=str(n)
    first_digit=int(str_n[0])
    last_three_digit=int(str_n[-3:])
    if last_three_digit % first_digit==0:
        print("Logged In")
    else:
        print("Incorrect code")
n=int(input())
check_sec_code(n)
```

Status : Correct *Marks : 10/10*

4. Problem Statement

Gabriel is working on a wildlife research project where he needs to compute various metrics for different animals based on their characteristics. Each animal type requires a different calculation: a deer's distance traveled, a bear's weight based on footprint size, or a bird's altitude based on its flying pattern.

Conditions:

For Deer (Mode 'D' or 'd'): Distance = speed of sound * time taken, where the speed of sound in air is 343 meters per second.For Bear (Mode 'B' or 'b'): Weight = footprint size * average weight, where the average weight per square inch for a bear is 5.0 pounds.For Bird (Mode 'F' or 'f'): Altitude = flying pattern * distance covered (in meters).

Write a program to help Gabriel analyze the characteristics of animals based on the given inputs.

Input Format

The first line of input consists of a character, representing the type of animal 'D/d' for deer, 'B/b' for bear, and 'F/f' for bird.

If the choice is 'D' or 'd':

The second line of input consists of a floating-point value T, representing the time taken from the deer's location to the observer.

If the choice is 'B' or 'b':

The second line of input consists of a floating-point value S, representing the size of the bear's footprint in square inches.

If the choice is 'F' or 'f':

1. The second line of input consists of a floating-point value P, representing the bird's flying pattern.
2. The third line consists of a floating-point value D, representing the distance

covered by the bird in meters.

Output Format

The output prints one of the following:

If the choice is 'D' or 'd':

The output prints "Distance: X m" where X is a floating point value rounded off to two decimal places, representing the calculated distance traveled by the sound wave in meters.

If the choice is 'B' or 'b':

The output prints "Weight: Y lb" where Y is a floating point value rounded off to two decimal places, representing the estimated weight of the bear in pounds.

If the choice is 'F' or 'f':

The output prints "Altitude: Z m" where Z is a floating point value rounded off to two decimal places, representing the calculated altitude of the bird's flight in meters.

If the given choice is invalid, print "Invalid".

Refer to the sample output for formatting specifications.

Sample Test Case

Input: d
2.5

Output: Distance: 857.50 m

Answer

```
# You are using Python
def cal_met(mode,value):
    s=343
    a=5.0
    if mode in['D','d']:
        dis=s*value
        print(f"Distance: {dis:.2f}m")
    elif mode in ['B','b']:
        w=value*a
        print(f"Weight: {w:.2f}lb")
    elif mode in ['F','f']:
        al=value*float(input())
        print(f"Altitude: {al:.2f}m")
    else:
        print("Invalid animal type")
mode=input().strip()
if mode in ['D','d']:
    T=float(input())
    cal_met(mode,T)
elif mode in ['B','b']:
    ft=float(input())
    cal_met(mode,ft)
elif mode in ['F','f']:
    fp=float(input())
    cal_met(mode,fp)
else:
    print("Invalid")
```

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