Rajalakshmi Engineering College

Name: Nitin Aakash
Email: 240701370@rajalakshmi.edu.in
Roll no: 240701370
Phone: 9498349045
Branch: REC
Department: I CSE FD
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

John is tasked with configuring the lighting for a high-profile event, where different lighting modes affect the ambiance of the venue. He can choose from three distinct lighting modes, each requiring a specific adjustment to the initial light intensity:

Ambient Lighting (Mode 1): The intensity level is multiplied by 1.5.Stage Lighting (Mode 2): The intensity level is multiplied by 2.0.Spotlight (Mode 3): The intensity level is multiplied by 1.8.

In the event that an invalid mode is provided, the program should output an error message indicating the invalid selection.

Your task is to write a program that reads the selected lighting mode and the initial intensity level, applies the appropriate adjustment, and prints the final intensity.

Input Format

The first line of input is an integer n, representing the lighting mode.

The second line is a floating value m, representing the initial intensity level of the light.

Output Format

The output displays "Intensity: " followed by a float representing the adjusted intensity level, formatted to two decimal places, if the mode is valid.

If the mode is invalid, the output should display "Invalid".

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 1
10.0

Output: Intensity: 15.00

Answer

```
def adjust_light_intensity(mode,intensity):
   if mode==1:
      result=intensity*1.5
   elif mode==2:
      result=intensity*2.0
   elif mode==3:
      result=intensity*1.8
   else:
      print("Invalid")
      return
   print(f"Intensity: {result:.2f}")

mode=int(input())
intensity=float(input())
adjust_light_intensity(mode,intensity)
```

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

2. 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

```
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 generator(n):
    tot = 0
    while n != 0:
        if not isprime(n%10):
            tot = tot + (n%10)
        n = n// 10
    print(tot)

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

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

3. 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

```
def isprime(num):
    if num < 2:
        return False

    for i in range(3 , int(num**0.5)+1):
        if num % i == 0:
            return False
    return True

def twinprime(n):
    prev = 2
    count = 0
    i = 3
    while count < n:
        if isprime(i):
            if i - prev == 2:
                print(prev , i)
                count += 1
            prev = i
        i += 2

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

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

4. 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

```
def is_prime(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 fibonacci_primes(n):
    fib1, fib2 = 0, 1
    primes = []

    while len(primes) < n:
        fib_next = fib1 + fib2
        if is_prime(fib_next):
          primes.append(fib_next)
        fib1, fib2 = fib2, fib_next

    return primes
n=int(input())
print(" ".join(map(str,fibonacci_primes(n))))
```

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