Rajalakshmi Engineering College

Name: Tharun Sathishkumar
Email: 240701563@rajalakshmi.edu.in
Roll no: 240701563
Phone: 9363661870
Branch: REC
Department: I CSE FF
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

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 isprime(num):
    if num <=1:
        return False
    if num== 2:
        return True
    if num%2==0:
        return False
    for i in range(3,int(num**0.5)+1,2):
        if num%i==0:
            return False
    return True
def prime(n):
    fibprime=[]
    a,b=1,1
    count=0
    while True:
        if isprime(a) and a not in fibprime:
            fibprime.append(a)
            count+=1
            if count==n:
                break
        a,b=b,a+b
    return fibprime
n=int(input())
result=prime(n)
print(" ".join(map(str,result)))
```

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

2. Problem Statement

Taylor is tasked with a mathematical challenge that requires finding the smallest positive number divisible by all integers from 1 to n.

Help Taylor to determine the smallest positive number that is divisible by all integers from 1 to n. Make sure to employ the break statement to ensure efficiency in the program.

Input Format

The input consists of a single integer, n.

Output Format

The output displays the smallest positive number that is divisible by all integers from 1 to n.

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: 10

Output: 2520

Answer

```
import math
n=int(input())
lcm=1
for i in range(1,n+1):
    lcm=lcm*i//math.gcd(lcm,i)
print(lcm)
```

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

3. 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(d):
    return d in {2,3,5,7}
x=input().strip()
sum=0
for digit in x:
```

```
    d=int(digit)
    if not isprime(d):
        sum+=d
print(sum)
```

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

4. 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

```
ccode=int(input())
if ccode==1:
    n=float(input())
    if n<50.0:
        print("Priority: High")
    else:
        print("Priority: Low")
elif ccode==2:
    n=float(input())
    if n>80.0:
        print("Priority: High")
    else:
        print("Priority: Low")
elif ccode==3:
    n=float(input())
    if n>80.0:
        print("Priority: High")
    else:
        print("Priority: Low")
else:
    print("Invalid")
```

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