

# 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 isfibo(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 isfibo(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