CO2

1. Program to find the factorial of a number
2. Generate Fibonacci series of N terms
3. Find the sum of all items in a list
4. Generate a list of four digit numbers in a given range with all their digits even and the number is a perfect square.
5. Display the given pyramid with step number accepted from user.
   Eg: N=4

```
1
2 4
3 6 9
4 8 12 16
```

6. Count the number of characters (character frequency) in a string.
7. Add ‘ing’ at the end of a given string. If it already ends with ‘ing’, then add ‘ly’
8. Accept a list of words and return length of longest word.
9. Construct following pattern using nested loop

```
*
* *
* * *
* * * *
* * * * *
* * * *
* * *
* *
*
```

10. Generate all factors of a number.
11. Write lambda functions to find the area of a square, rectangle, and triangle.

1. Program to find the factorial of a number

```
def factorial(n):
    if n == 0:
        return 1
    else:
        return n * factorial(n - 1)

num = int(input("Enter a number:"))
if num < 0:
    print("Factorial is not defined for negative numbers.")
elif num == 0:
    print(f"{num}! = 1")
else:
    fact = factorial(num)
    print(f"{num} ! = {fact}")
```

Output

```
Enter a number:5
5 ! = 120

Enter a number:-1
Factorial is not defined for negative numbers.

Enter a number:0
0! = 1

Enter a number:1
  1 ! = 1
```

2. Generate Fibonacci series of N terms

```
def fib(n):
    f1=0
    f2=1
    print(f1)
    print(f2)
    for i in range(3,n+1):
        f3=f1+f2
        print(f3)
        f1=f2
        f2=f3
num=int(input("Enter the limit: "))
fib(num)
```

Output

```
Enter the limit: 5
0
1
1
2
3
```

3. Find the sum of all items in a list using function

```
def sumList(list1):
    total = 0
    for i in list1:
        total += i
    return total

list1 = [1, 2, 3, 4, 5]
result = sumList(list1)
print("Sum of the list:", result)
```

Output

```
Sum of the list: 15
```

4. Generate a list of four digit numbers in a given range with all their digits even and the number is a perfect square

```
def Even_PerfectSq(start, end):
    evenSq = []
    for num in range(start, end + 1):
        # Check if all digits are even
        if all(int(digit) % 2 == 0
            for digit in str(num)):
            # Check if the number is a perfect square
               sqrt = int(num ** 0.5)
               if sqrt * sqrt == num:
                   evenSq.append(num)
    return evenSq

start = 1000
end = 9999
result = Even_PerfectSq(start, end)
print(result)
```

Output

```
[4624, 6084, 6400, 8464]
```

5. Display the given pyramid with step number accepted from user.
   Eg: N=4

```
1
2 4
3 6 9
4 8 12 16
```

```
def disp_pyramid(n):
    for i in range(1, n + 1):
        for j in range(1, i + 1):
            print(i * j, end=" ")
        print()

n = int(input("Enter the number of steps: "))
if n < 1:
     print("Please enter a positive integer.")
else:
        disp_pyramid(n)
```

Output

```
Enter the number of steps: 4
1
2 4
3 6 9
4 8 12 16
```

6. Count the number of characters (character frequency) in a string using a dictionary

```
def countChar(inputString):
    Count = {}
    for char in inputString:
        if char in Count:
            Count[char] += 1
        else:
            Count[char] = 1
    return Count
str2 = input("Enter a string: ")
result = countChar(str2)
print(result)
```

Output

```
Enter a string: ilahia college
{'i': 2, 'l': 3, 'a': 2, 'h': 1, ' ': 1, 'c': 1, 'o': 1, 'e': 2,
'g': 1}
```

7. Add ‘ing’ at the end of a given string. If it already ends with ‘ing’, then add ‘ly’

```
def modifyStr(str1):
    if str1.endswith("ing"):
        return str1 + "ly"
    else:
        return str1 + "ing"
str1 = input("Enter a string: ")
result = modifyStr(str1)
print(result)
```

Output

```
Enter a string: running
runningly
```

8. Accept a list of words and return length of the longest word.

```
def longestWord(word):
    max_length = len(word[0])  # Initialize with the length of the first
word
    for item in word:
        length = len(item)
        if length > max_length:
            max_length = length
    return max_length
words= input("Enter a list of words separated by spaces: ")
word = words.split()
result = longestWord(word)
print(f"The length of the longest word is: {result}")
```

Output

```
Enter a list of words separated by spaces: sheena leena reshy
The length of the longest word is: 6
```

9. Construct the following pattern using nested loop

```
*
* *
* * *
* * * *
* * * * *
* * * *
* * *
* *
*
```

```
def pattern(n):
  for i in range(n):
     for j in range(i):
        print('*',end=" ")
     print(' ')
  for i in range(n,0,-1):
     for j in range(i):
       print('*',end=" ")
     print(' ')
n = int(input("Enter the number of rows: "))
pattern(n)
```

Output

```
*
* *
* * *
* * * *
* * * * *
* * * *
* * *
* *
*
```

10. Generate all factors of a number.

```
def find_factors(number):
    factors = []
    for i in range(1, number + 1):
        if number % i == 0:
            factors.append(i)
    return factors
num = int(input("Enter the number of rows: "))
result = find_factors(num)
print(f"The factors of {num} are: {result}")
```

Output

```
Enter the number of rows: 24
The factors of 24 are: [1, 2, 3, 4, 6, 8, 12, 24]
```

11. Write lambda functions to find the area of a square, rectangle, and triangle.

```
square_area = lambda a: a**2
rectangle_area = lambda l, b: l * b
triangle_area = lambda ba, h: 0.5 * ba * h


a = int(input("Enter the length: "))
print("Area of the square:", square_area(a))

l = int(input("Enter the length: "))
b = int(input("Enter the breadth: "))
print("Area of the triangle:", rectangle_area(l, b))

ba = int(input("Enter the base: "))
h = int(input("Enter the height: "))
print("Area of the rectangle:", triangle_area(ba, h))
```

Output

```
Enter the length: 4
Area of the square: 16
Enter the length: 2
Enter the breadth: 3
Area of the triangle: 6
Enter the base: 2
Enter the height: 5
Area of the rectangle: 5.0
```