CYCLE 4
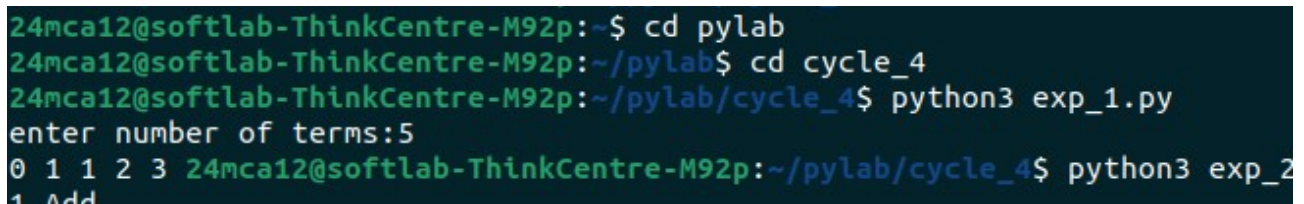
PROGRAM 1

Aim : Write a program to print the Fibonacci series using recursion.

Source code :

```
def fibonacci(n):
    if n<=1:
        return n
    else:
        return fibonacci(n-1)+fibonacci(n-2)
n=int(input("enter number of terms:"))
for i in range(n):
    print(fibonacci(i),end=" ")
```

Output :

PROGRAM 2

Aim : Write the to implement a menu-driven calculator. Use separate functions for the different operations.

Source code :

```
def add(a,b):
    return a+b
def subtract(a,b):
    return a-b
def multiply(a,b):
    return a*b
def divide(a,b):
    return a/b
def calculator():
    while True:
        print("1.Add\n2.Subtract\n3.Multiply\n4.Divide\n5.Exit")
```

```python
        choice=int(input("Enter your choice : "))
        a=float(input("Enter first number : "))
        b=float(input("Enter second number : "))
        if choice ==1:
            print("Result : ",add(a,b))
        elif choice==2:
            print("Result : ",subtract(a,b))
        elif choice==3:
            print("Result : ",multiply(a,b))
        elif choice==4:
            print("Result : ",divide(a,b))
        elif choice==5:
            break
        else :
            print("Invalid choice!")
calculator()
```

Output :

```
1.Add
2.Subtract
3.Multiply
4.Divide
5.Exit
Enter your choice : 1
Enter first number : 4
Enter second number : 5
Result :  9.0
1.Add
2.Subtract
3.Multiply
4.Divide
5.Exit
Enter your choice : 2
Enter first number : 4
Enter second number : 7
Result :  -3.0
1.Add
2.Subtract
3.Multiply
4.Divide
5.Exit
Enter your choice : 3
Enter first number : 2
Enter second number : 4
Result :  8.0
```
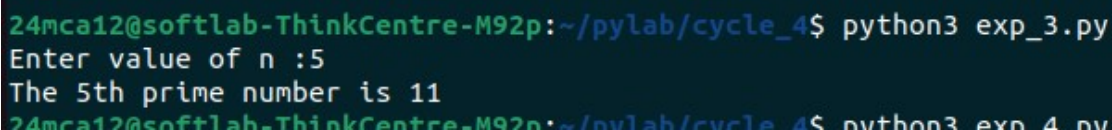
PROGRAM 3

Aim : Write a program to print the nth prime number.

Source code :

```python
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 nth_prime(n):
    count=0
    num=1
    while  count<n :
        num+=1
        if  is_prime(num):
            count+=1
    return num
n=int(input("Enter value of n :"))
print(f"The {n}th prime number is",nth_prime(n))
```

Output :

```
24mca12@softlab-ThinkCentre-M92p:~/pylab/cycle_4$ python3 exp_3.py
Enter value of n :5
The 5th prime number is 11
24mca12@softlab-ThinkCentre-M92p:~/pylab/cycle_4$ python3 exp_4.py
```
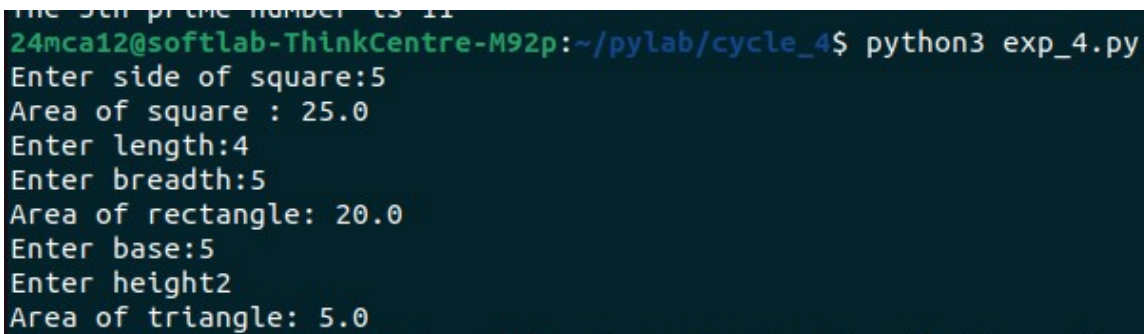
PROGRAM 4

Aim : Write lambda functions to find the area of square, rectangle and triangle.

Source code :

```python
square=lambda side:side*side
rectangle=lambda length,breadth:length*breadth
triangle=lambda base,height:0.5*base*height
side=float(input("Enter side of square:"))
print("Area of square :",square(side))
length=float(input("Enter length:"))
breadth=float(input("Enter breadth:"))
print("Area of rectangle:",rectangle(length,breadth))
```

```
base=float(input("Enter base:"))
height=float(input("Enter height"))
print("Area of triangle:",triangle(base,height))
```

Output :

```
Enc 5th prime number is 11
24mca12@softlab-ThinkCentre-M92p:~/pylab/cycle_4$ python3 exp_4.py
Enter side of square:5
Area of square : 25.0
Enter length:4
Enter breadth:5
Area of rectangle: 20.0
Enter base:5
Enter height2
Area of triangle: 5.0
```
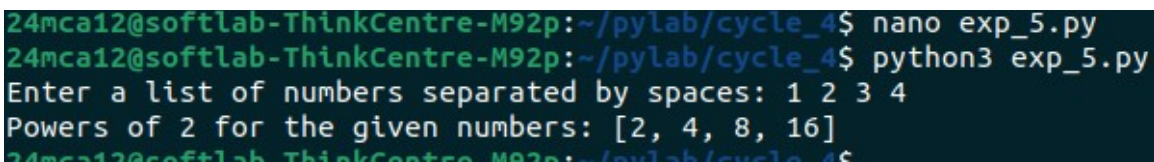
PROGRAM 5

Aim : Write a program to display powers of 2 using anonymous function.

Source code :

```
user_input = input("Enter a list of numbers separated by spaces: ")
numbers = list(map(int, user_input.split()))
powers_of_2 = list(map(lambda x: 2 ** x, numbers))
print(f"Powers of 2 for the given numbers: {powers_of_2}")
```

Output :

```
24mca12@softlab-ThinkCentre-M92p:~/pylab/cycle_4$ nano exp_5.py
24mca12@softlab-ThinkCentre-M92p:~/pylab/cycle_4$ python3 exp_5.py
Enter a list of numbers separated by spaces: 1 2 3 4
Powers of 2 for the given numbers: [2, 4, 8, 16]
24mca12@softlab ThinkCentre M92p:~/pylab/cycle_4$
```

PROGRAM 6

Aim : Write a program to display multiples of 3 using anonymous function.

Source code :

```
user_input = input("Enter a list of numbers separated by spaces: ")
numbers = list(map(int, user_input.split()))
```

```
multiples_of_3 = list(map(lambda x: 3 * x, numbers))
print(f"Multiples of 3 for the given numbers: {multiples_of_3}")
```

Output :

```
24mca12@softlab-ThinkCentre-M92p:~/pylab/cycle_4$ nano exp_6.py
24mca12@softlab-ThinkCentre-M92p:~/pylab/cycle_4$ python3 exp_6.py
Enter a list of numbers separated by spaces: 1 2 3 4 5
Multiples of 3 for the given numbers: [3, 6, 9, 12, 15]
```

PROGRAM 7

Aim : Write a program to sum the series $1/1! + 4/2! + 27/3! + \ldots + nth$ term.

Source code :

```
def factorial(n):
    fact = 1
    for i in range(1, n + 1):
        fact *= i  # Corrected: multiply by i to calculate the factorial
    return fact

def sum_series(n):
    total = 0
    for i in range(1, n + 1):
        total += (i**3) / factorial(i)  # Sum of i^3 / i!
    return total

n = int(input("Enter number of terms: "))
print(f"Sum of series: {sum_series(n)}")
```

Output :

```
24mca12@softlab-ThinkCentre-M92p:~/pylab/cycle_4$ python3 exp_7.py
Enter number of terms:5
Sum of series:13.208333333333332
```
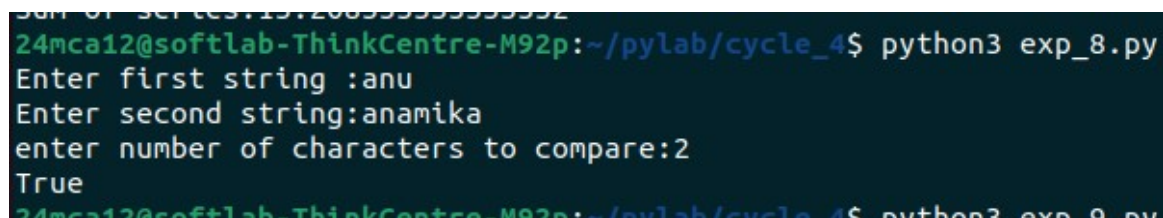
PROGRAM 8

Aim : Write a function called compare which takes two strings S1 and S2 and an integer n as arguments. The function should return True if the first n characters of both the strings are the same else the function should return False.

Source code :

```
def compare(s1,s2,n):
    return s1[:n]==s2[:n]
s1=input("Enter first string :")
s2=input("Enter second string:")
n=int(input("enter number of characters to compare:"))
print(compare(s1,s2,n))
```

Output :

```
Sum of series:15.20899999999992
24mca12@softlab-ThinkCentre-M92p:~/pylab/cycle_4$ python3 exp_8.py
Enter first string :anu
Enter second string:anamika
enter number of characters to compare:2
True
24mca12@softlab-ThinkCentre-M92p:~/pylab/cycle_4$ python3 exp_9.py
```
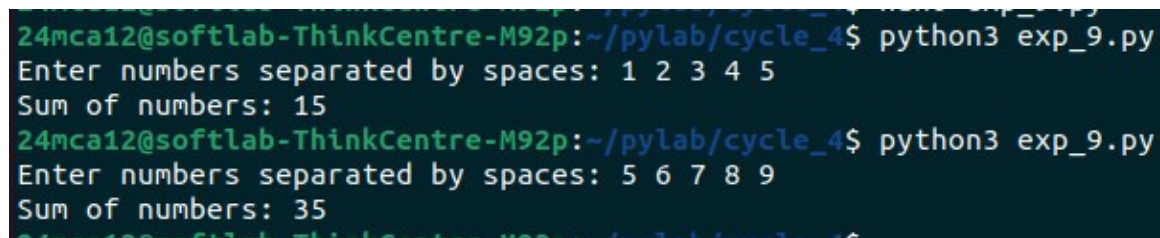
PROGRAM 9

Aim : Write a program to add variable length integer arguments passed to the function.

Source code :

```
def add(*args):
    """Function to add variable length arguments"""
    return sum(args)
user_input = input("Enter numbers separated by spaces: ")
numbers = list(map(int, user_input.split()))
print("Sum of numbers:", add(*numbers))
```

Output :

```
24mca12@softlab-ThinkCentre-M92p:~/pylab/cycle_4$ python3 exp_9.py
Enter numbers separated by spaces: 1 2 3 4 5
Sum of numbers: 15
24mca12@softlab-ThinkCentre-M92p:~/pylab/cycle_4$ python3 exp_9.py
Enter numbers separated by spaces: 5 6 7 8 9
Sum of numbers: 35
24mca12@softlab-ThinkCentre-M92p:~/pylab/cycle_4$
```

PROGRAM 10

Aim : Write a program using functions to implement these formulae for permutations and combinations.

Source code :

```
def factorial(n):
    fact = 1
    for i in range(1, n + 1):
        fact *= i
    return fact

def permutations(n, r):
    return factorial(n) // factorial(n - r)

def combinations(n, r):
    return factorial(n) // (factorial(r) * factorial(n - r))

n = int(input("Enter n: "))
r = int(input("Enter r: "))

print(f"Permutations p({n},{r}) = {permutations(n, r)}")
print(f"Combinations c({n},{r}) = {combinations(n, r)}")
```

Output :

```
24mca12@softlab-ThinkCentre-M92p:~/pylab/cycle_4$ python3 exp_10.py
Enter n :5
Enter r:2
Permutations(p(5,2)):20
Combinations(c(5,2)):10
```