**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:

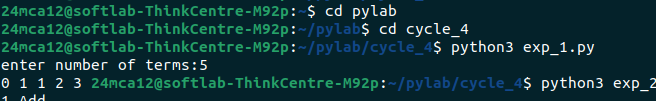
return fibonacci(n-1)+fibonacci(n-2)

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")

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:

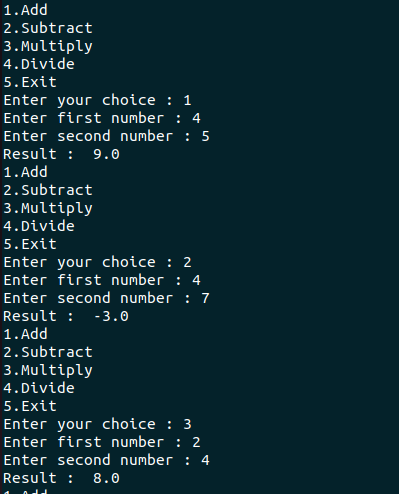
break

else :

print("Invalid choice!")

calculator()

**Output :**

****

**PROGRAM 3**

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

**Source code :**

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):

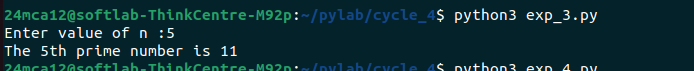
count+=1

return num

n=int(input("Enter value of n :"))

print(f"The {n}th prime number is",nth\_prime(n))

**Output :**

****

**PROGRAM 4**

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

**Source code :**

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:"))

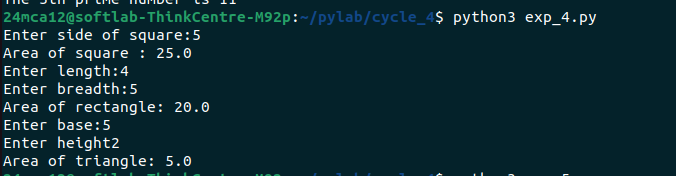
print("Area of rectangle:",rectangle(length,breadth))

base=float(input("Enter base:"))

height=float(input("Enter height"))

print("Area of triangle:",triangle(base,height))

**Output :**

****

**PROGRAM 5**

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

**Source code :**

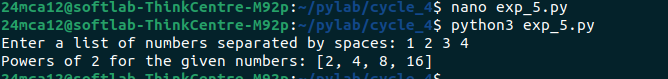
user\_input = input("Enter a list of numbers separated by spaces: ")

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 :**

****

**PROGRAM 6**

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

**Source code :**

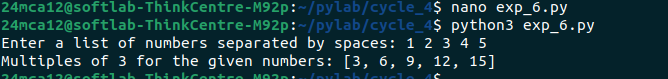
user\_input = input("Enter a list of numbers separated by spaces: ")

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 :**

****

**PROGRAM 7**

**Aim :** Write a program to sum the series 1/1! + 4/2! + 27/3! + ….. + 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):

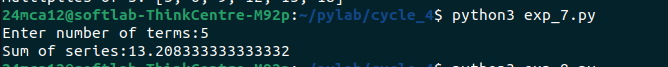
total += (i\*\*3) / factorial(i) # Sum of i^3 / i!

return total

n = int(input("Enter number of terms: "))

print(f"Sum of series: {sum\_series(n)}")

**Output :**

****

**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]

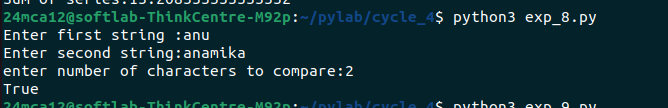
s1=input("Enter first string :")

s2=input("Enter second string:")

n=int(input("enter number of characters to compare:"))

print(compare(s1,s2,n))

**Output :**

****

**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"""

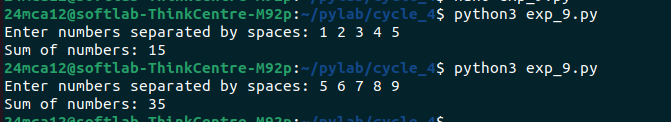
return sum(args)

user\_input = input("Enter numbers separated by spaces: ")

numbers = list(map(int, user\_input.split()))

print("Sum of numbers:", add(\*numbers))

**Output :**

****

**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 :**

