**CYCLE 3**

**PROGRAM 1**

**Aim** : Program to find the factorial of a number.

**Source code :**

def factorial(n):

fact=1

for i in range(1,n+1):

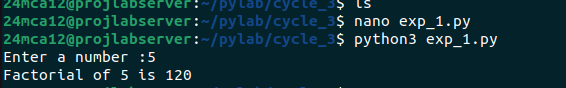
fact \*= i

return fact

n=int(input("Enter a number :"))

print("Factorial of",n,"is",factorial(n))

**Output :**

****

**PROGRAM 2**

**Aim :** Program to Generate Fibonacci series of N terms.

**Source code :**

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

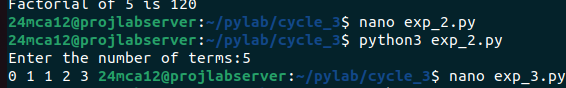
a,b=0,1

for i in range(n):

print(a,end=" ")

a,b=b,a+b

**Output :**

****

**PROGRAM 3**

**Aim :** Write a program to find the sum of all items in a list. [Using for loop]

**Source code :**

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

sum=0

num=[]

for i in range(n):

n1=int(input("enter numbers:"))

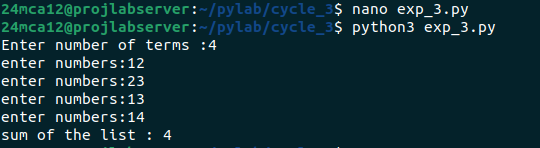
num.append(n1)

for i in num:

sum+=i

print("sum of the list :",sum)

**Output :**

****

**PROGRAM 4**

**Aim :** Generate a list of four digit numbers in a given range with all their digits even and

the number is a perfect square.

**Source code :**

import math

start = int(input("Enter starting(1000-9999):"))

end = int(input("Enter ending(1000-9999):"))

for num in range(start,end+1):

temp = num

all\_even = True

while temp > 0 :

if(temp % 10) % 2 != 0:

all\_even = False

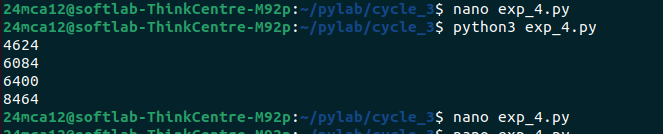
break

temp //= 10

if all\_even and math.isqrt(num) \*\* 2 == num:

print(num)

**Output :**

****

**PROGRAM 5**

**Aim :** Write a program using a for loop to print the multiplication table of n, where n is entered by the user.

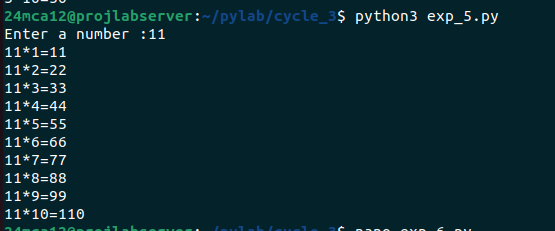
**Source code :**

n=int(input("Enter a number :"))

for i in range(1,11):

print(f"{n}\*{i}={n\*i}")

**Output :**

****

**PROGRAM 6**

**Aim :** Write a program to display alternate prime numbers till N (obtain N from the user).

**Source code :**

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

primes = []

for num in range(2,n+1):

is\_prime = True

for i in range(2,int(num \*\* 0.5)+1):

if num % i == 0:

is\_prime = False

break

if is\_prime :

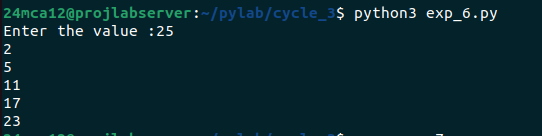
primes.append(num)

for i in range(0,len(primes),2):

print(primes[i],end=" ")

print( )

**Output :**

****

**PROGRAM 7**

**Aim :** Write a program to compute and display the sum of all integers that are divisible

By 6 but not by 4, and that lie below a user-given upper limit.

**Source code :**

limit=int(input("Enter the upper limit :"))

total = 0

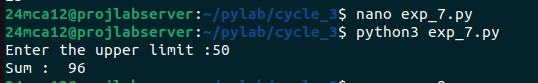
for num in range (1,limit):

if num % 6 == 0 and num % 4 != 0:

total += num

print("Sum : ",total)

**Output :**

****

**PROGRAM 8**

**Aim :** Calculate the sum of the digits of each number within a specified range (from 1 to

A user-defined upper limit). Print the sum only if it is prime.

**Source code :**

limit=int(input("Enter the upper limit : "))

for num in range(1,limit+1):

sum=0

for digit in str(num):

sum += int(digit)

if sum > 1:

is\_prime = True

for i in range(2,int(sum\*\*0.5)+1):

if sum % i == 0:

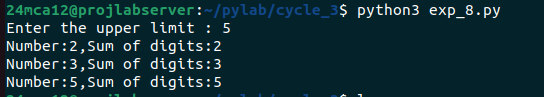
is\_prime = False

break

if is\_prime:

print(f"Number:{num},Sum of digits:{sum}")

**Output :**

****

**PROGRAM 9**

**Aim :** A number is input through the keyboard. Write a program to determine if it’s

Palindromic.

**Source code :**

n=int(input("Enter a number :"))

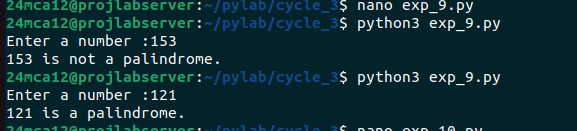
if str(n) == str(n)[::-1]:

print(n,"is a palindrome.")

else:

print(n,"is not a palindrome.")

**Output :**

****

**PROGRAM 10**

**Aim :** Write a program to generate all factors of a number. [use while loop]

**Source code :**

n=int(input("Enter a number:"))

print("Factors :")

i=1

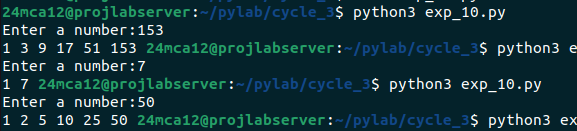
while i <= n:

if n % i == 0:

print(i,end=" ")

i += 1

**Output :**



**PROGRAM 11**

**Aim :** Write a program to find whether the given number is an Armstrong number or

not. [use while loop]

**Source code :**

n=int(input("Enter a number:"))

num = n

result = 0

while n > 0:

digit = n % 10

result += digit \*\* 3

n //= 10

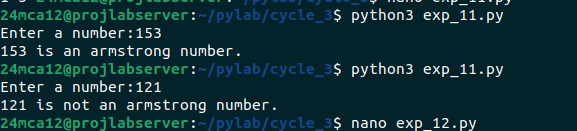
if result == num:

print(num,"is an armstrong number.")

else:

print(num,"is not an armstrong number.")

**Output :**

****

**PROGRAM 12**

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

1

2 4

3 6 9

4 8 12 16

**Source code :**

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

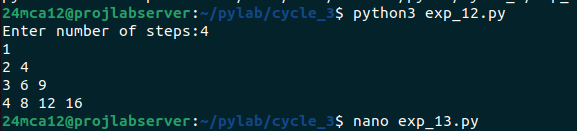
for i in range(1,n+1):

for j in range(1,i+1):

print(i\*j,end=" ")

print( )

**Output :**

****

**PROGRAM 13**

**Aim :** Construct pattern using nested loop.

**Source code :**

n=5

for i in range(1,n+1):

pattern = '\*' \* i

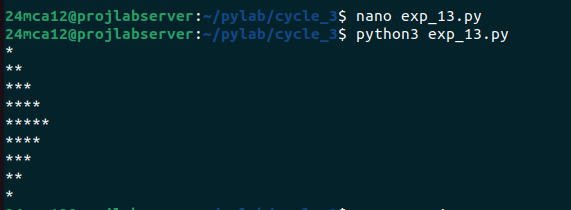
print(pattern)

for i in range(n-1,0,-1):

pattern = '\*' \* i

print(pattern)

**Output :**

****