CYCLE 3

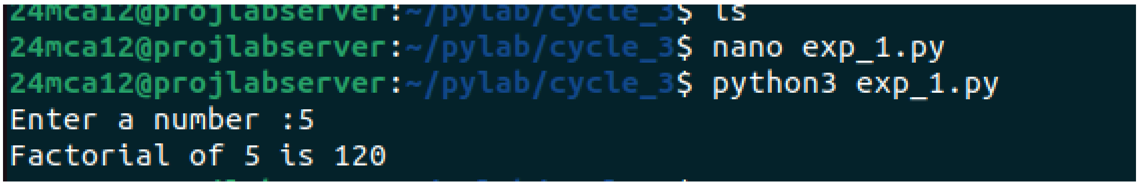
PROGRAM 1

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 :



```
24mca12@projlabserver:~/pylab/cycle_3$ ls
24mca12@projlabserver:~/pylab/cycle_3$ nano exp_1.py
24mca12@projlabserver:~/pylab/cycle_3$ python3 exp_1.py
Enter a number :5
Factorial of 5 is 120
```

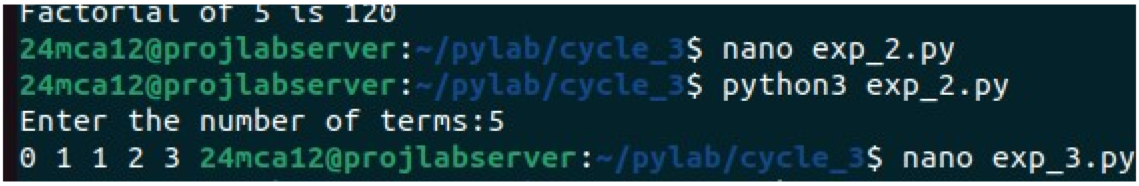
PROGRAM 2

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 :



```
Factorial of 5 is 120
24mca12@projlabserver:~/pylab/cycle_3$ nano exp_2.py
24mca12@projlabserver:~/pylab/cycle_3$ python3 exp_2.py
Enter the number of terms:5
0 1 1 2 3 24mca12@projlabserver:~/pylab/cycle_3$ nano exp_3.py
```

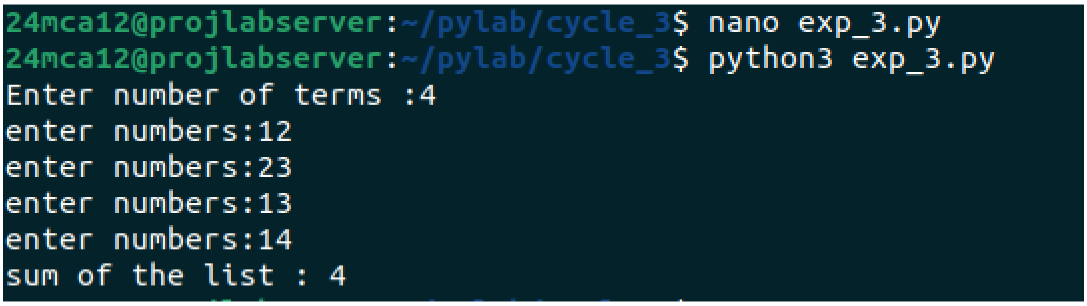
PROGRAM 3

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 :



```
24mca12@projlabserver:~/pylab/cycle_3$ nano exp_3.py
24mca12@projlabserver:~/pylab/cycle_3$ python3 exp_3.py
Enter number of terms :4
enter numbers:12
enter numbers:23
enter numbers:13
enter numbers:14
sum of the list : 4
```

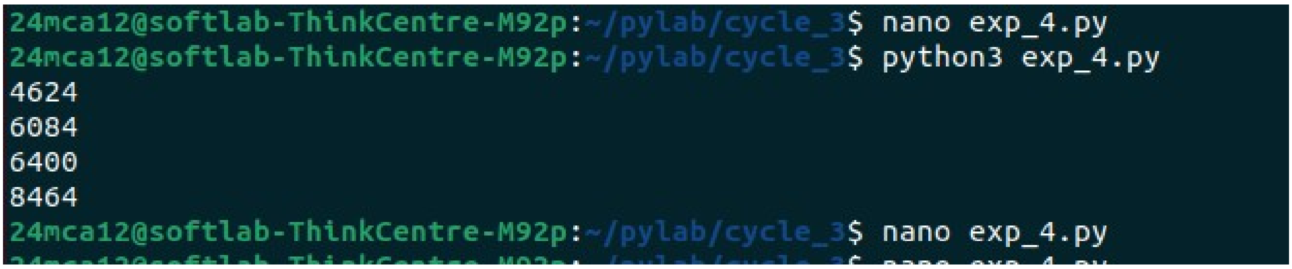
PROGRAM 4

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 :



```
24mca12@softlab-ThinkCentre-M92p:~/pylab/cycle_3$ nano exp_4.py
24mca12@softlab-ThinkCentre-M92p:~/pylab/cycle_3$ python3 exp_4.py
4624
6084
6400
8464
24mca12@softlab-ThinkCentre-M92p:~/pylab/cycle_3$ nano exp_4.py
```

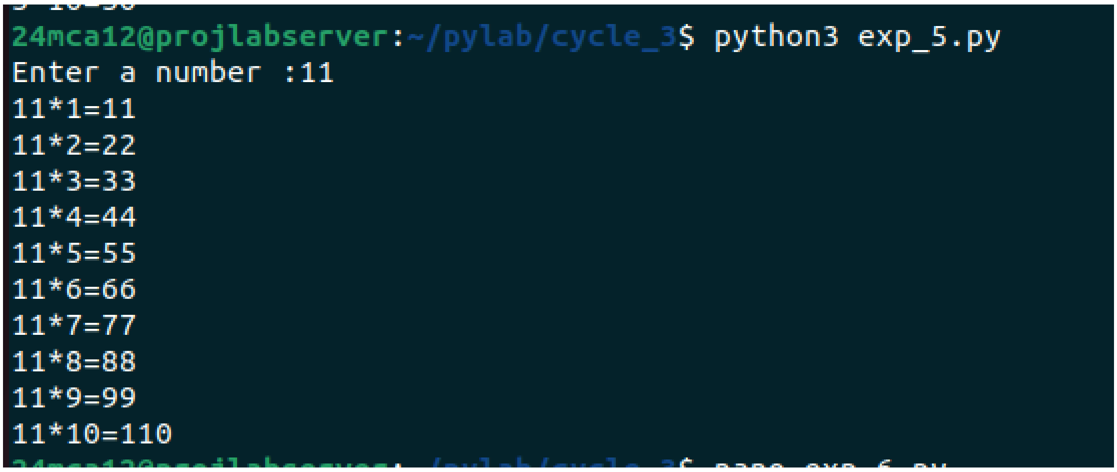
PROGRAM 5

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 :



```
24mca12@projlabserver:~/pylab/cycle_3$ python3 exp_5.py
Enter a number :11
11*1=11
11*2=22
11*3=33
11*4=44
11*5=55
11*6=66
11*7=77
11*8=88
11*9=99
11*10=110
```

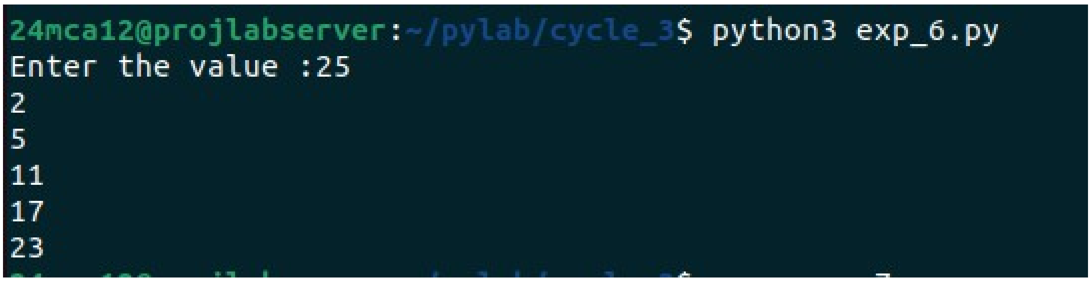
PROGRAM 6

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 :



```
24mca12@projlabserver:~/pylab/cycle_3$ python3 exp_6.py
Enter the value :25
2
5
11
17
23
```

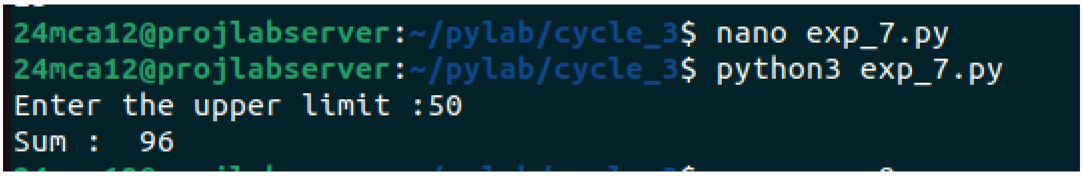
PROGRAM 7

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 :



```
24mca12@projlabserver:~/pylab/cycle_3$ nano exp_7.py
24mca12@projlabserver:~/pylab/cycle_3$ python3 exp_7.py
Enter the upper limit :50
Sum :  96
```

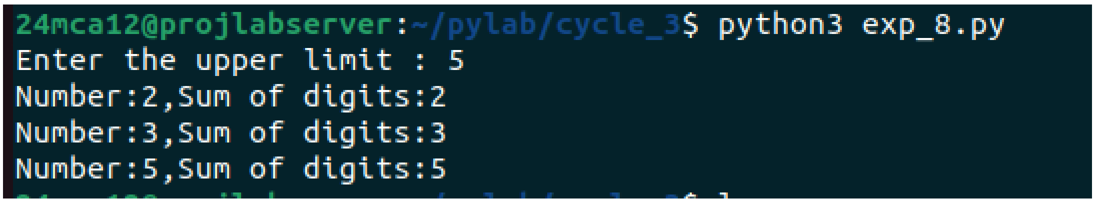
PROGRAM 8

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 :



```
24mca12@projlabserver:~/pylab/cycle_3$ python3 exp_8.py
Enter the upper limit : 5
Number:2,Sum of digits:2
Number:3,Sum of digits:3
Number:5,Sum of digits:5
```

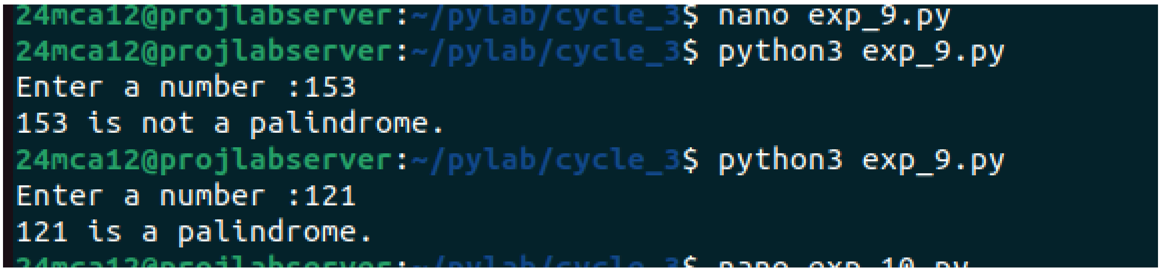
PROGRAM 9

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 :



```
24mca12@projlabserver:~/pylab/cycle_3$ nano exp_9.py
24mca12@projlabserver:~/pylab/cycle_3$ python3 exp_9.py
Enter a number :153
153 is not a palindrome.
24mca12@projlabserver:~/pylab/cycle_3$ python3 exp_9.py
Enter a number :121
121 is a palindrome.
24mca12@projlabserver:~/pylab/cycle_3$ nano exp_10.py
```

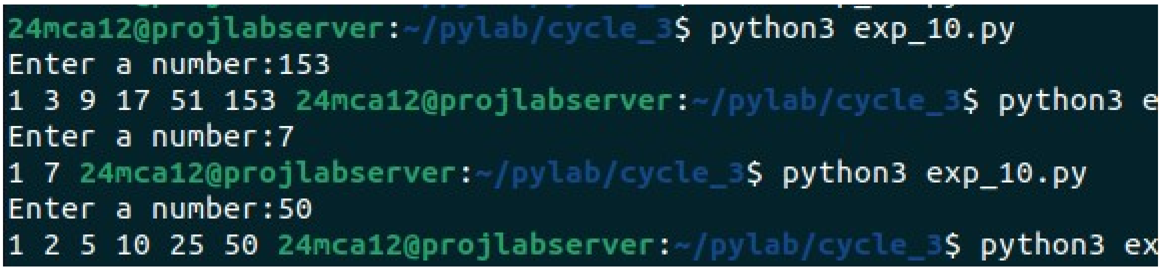
<u>**PROGRAM 10**</u>

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 :



```
24mca12@projlabserver:~/pylab/cycle_3$ python3 exp_10.py
Enter a number:153
1 3 9 17 51 153 24mca12@projlabserver:~/pylab/cycle_3$ python3 e
Enter a number:7
1 7 24mca12@projlabserver:~/pylab/cycle_3$ python3 exp_10.py
Enter a number:50
1 2 5 10 25 50 24mca12@projlabserver:~/pylab/cycle_3$ python3 ex
```

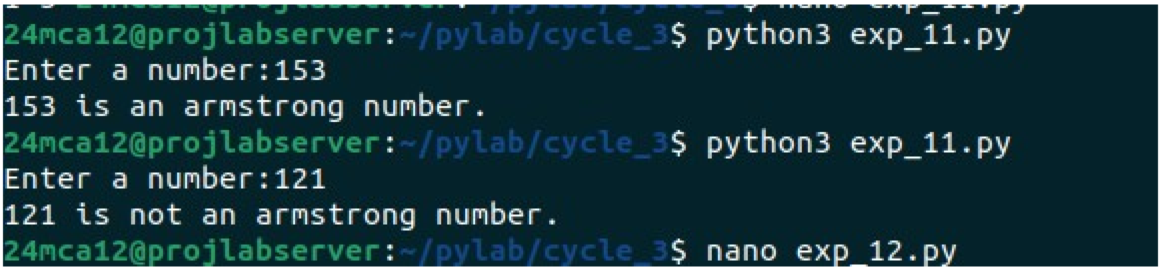
PROGRAM 11

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 :



```
24mca12@projlabserver:~/pylab/cycle_3$ python3 exp_11.py
Enter a number:153
153 is an armstrong number.
24mca12@projlabserver:~/pylab/cycle_3$ python3 exp_11.py
Enter a number:121
121 is not an armstrong number.
24mca12@projlabserver:~/pylab/cycle_3$ nano exp_12.py
```

PROGRAM 12

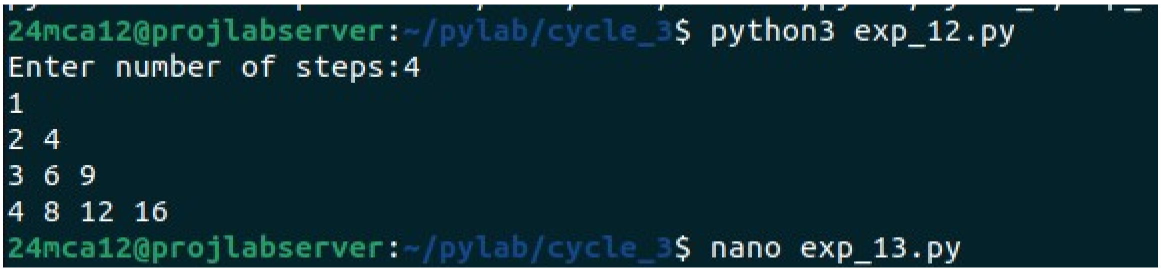
Aim : Display the given pyramid with the step number accepted from the user. Eg: N=4

```
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 :



```
24mca12@projlabserver:~/pylab/cycle_3$ python3 exp_12.py
Enter number of steps:4
1
2 4
3 6 9
4 8 12 16
24mca12@projlabserver:~/pylab/cycle_3$ nano exp_13.py
```

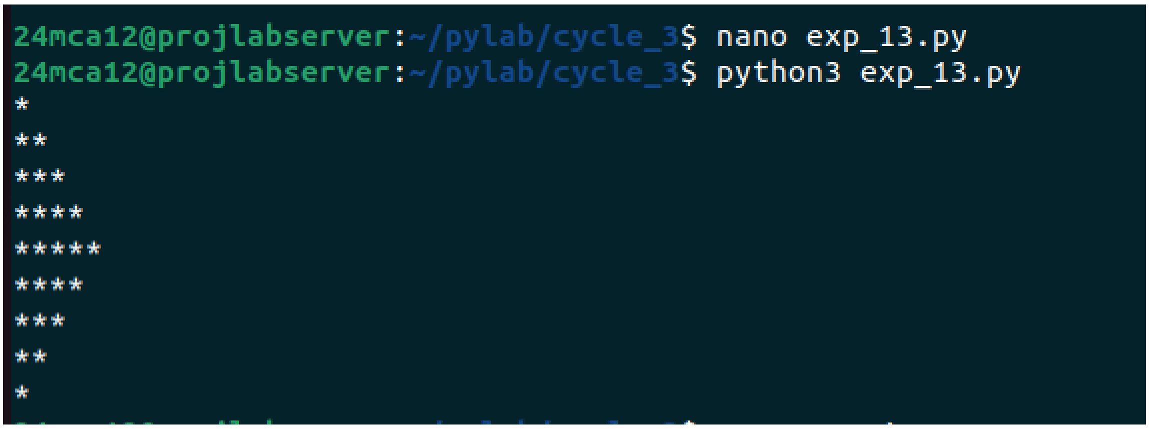
PROGRAM 13

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 :



```
24mca12@projlabserver:~/pylab/cycle_3$ nano exp_13.py
24mca12@projlabserver:~/pylab/cycle_3$ python3 exp_13.py
*
**
***
****
*****
****
***
**
*
```