Ex. No: 11

Aim: *Write a Python script that prints prime numbers less than 20.*

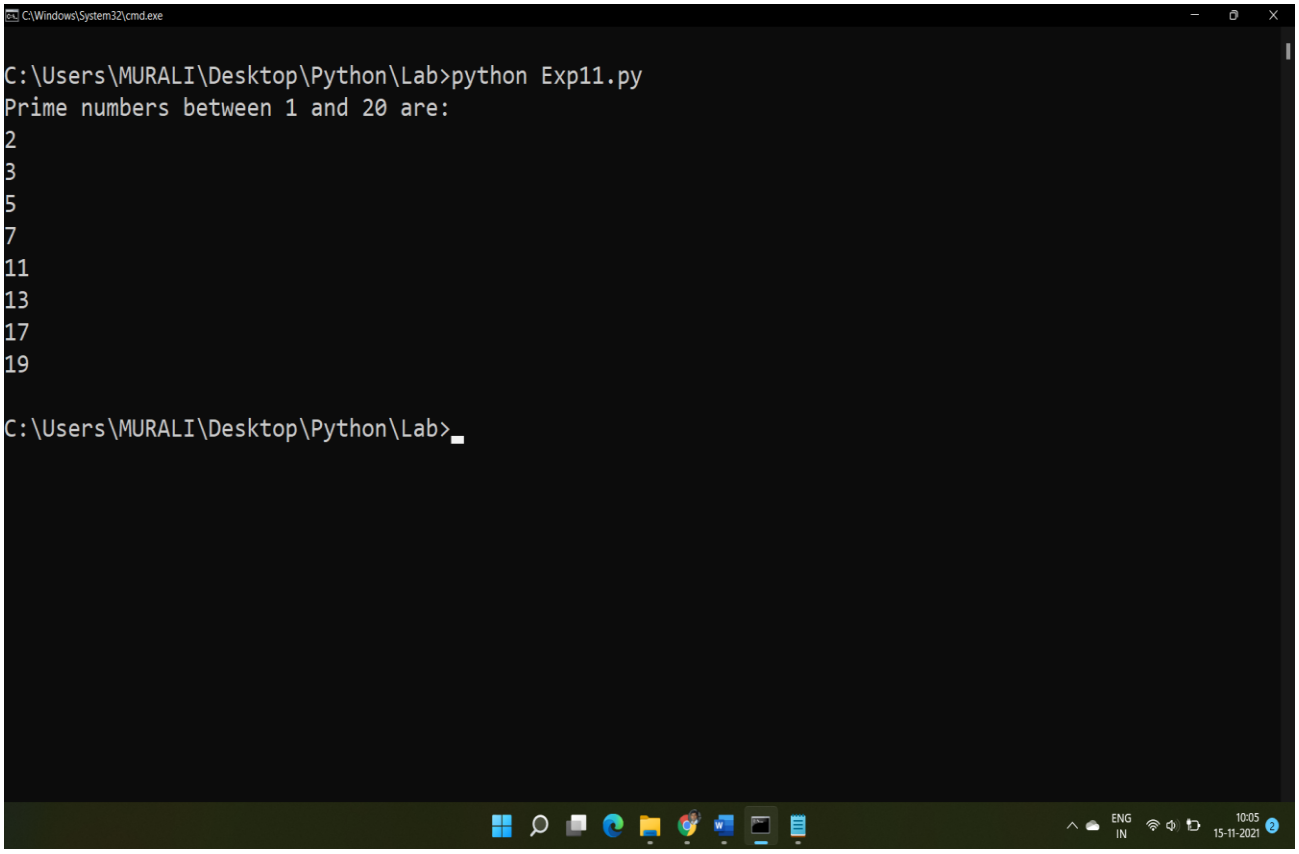
Theory:

Prime Number: A number greater than 1 with exactly two factors, i.e., 1 and the number itself is defined as a prime number.

Source Code:

```
print("Prime numbers between 1 and 20 are:")
for num in range(20):
  if num>1:
     for i in range(2,num):
        if(num%i)==0:
           break
     else:
        print(num)
```

Output:



Ex. No: 12

Aim: *Write a python program to find factorial of a number using Recursion.*

Theory:

Factorial: The factorial of a number is the function that multiplies the number by every natural number below it.

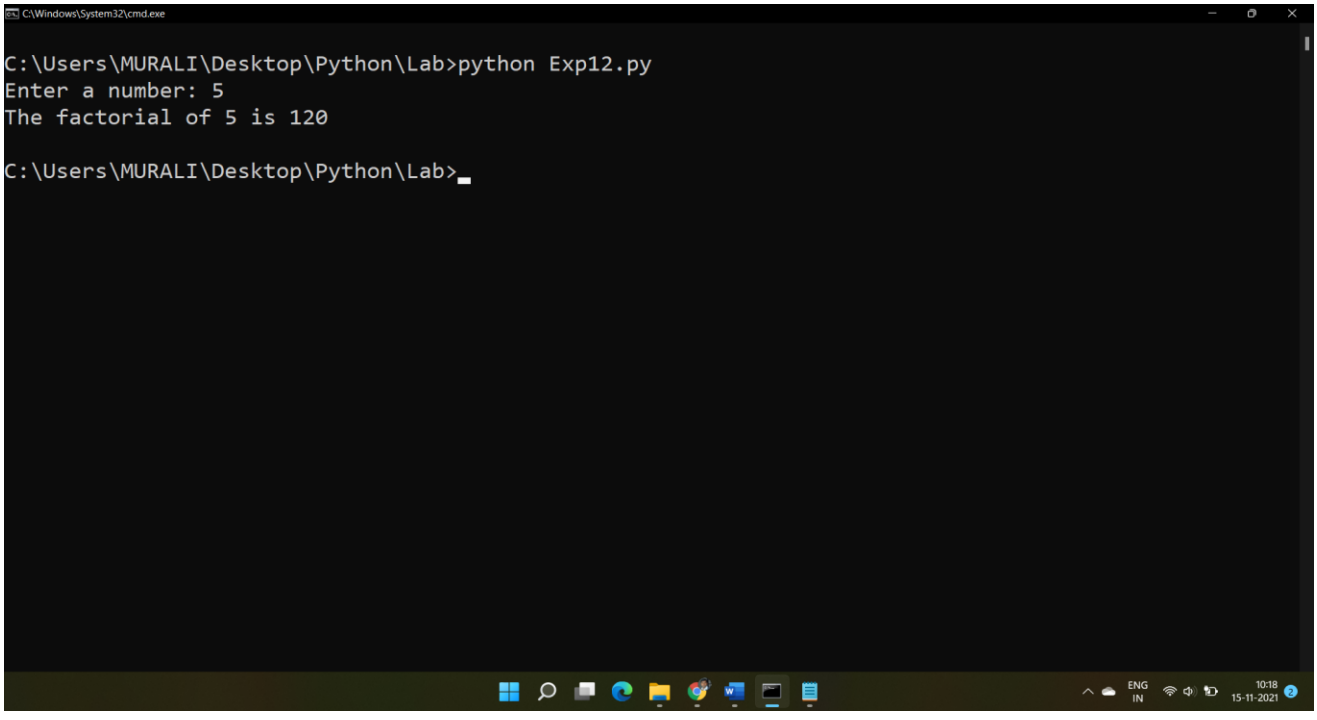
Recursion: The process in which a function calls itself directly or indirectly is called recursion and the corresponding function is called as recursive function.

Source Code:

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

num = int(input("Enter a number: "))
if num < 0:
    print("Sorry, factorial does not exist for negative numbers")
elif num == 0:
    print("The factorial of 0 is 1")
else:
    print("The factorial of",num,"is",factorial(num))
```

Output:



Ex. No: 13

Aim: *Write a program that accepts the lengths of three sides of a triangle as inputs. The program output should indicate whether or not the triangle is a right triangle (Recall from the Pythagorean Theorem that in a right triangle, the square of one side equals the sum of the squares of the other two sides).*

Theory:

Pythagorean Theorem: Pythagorean theorem states that the square of the length of the hypotenuse will be equal to the sum of the squares of the lengths of the other two sides of the right-angled triangle.

The Formula of Pythagorean Theorem:

So, mathematically, we represent the Pythagoras theorem as:

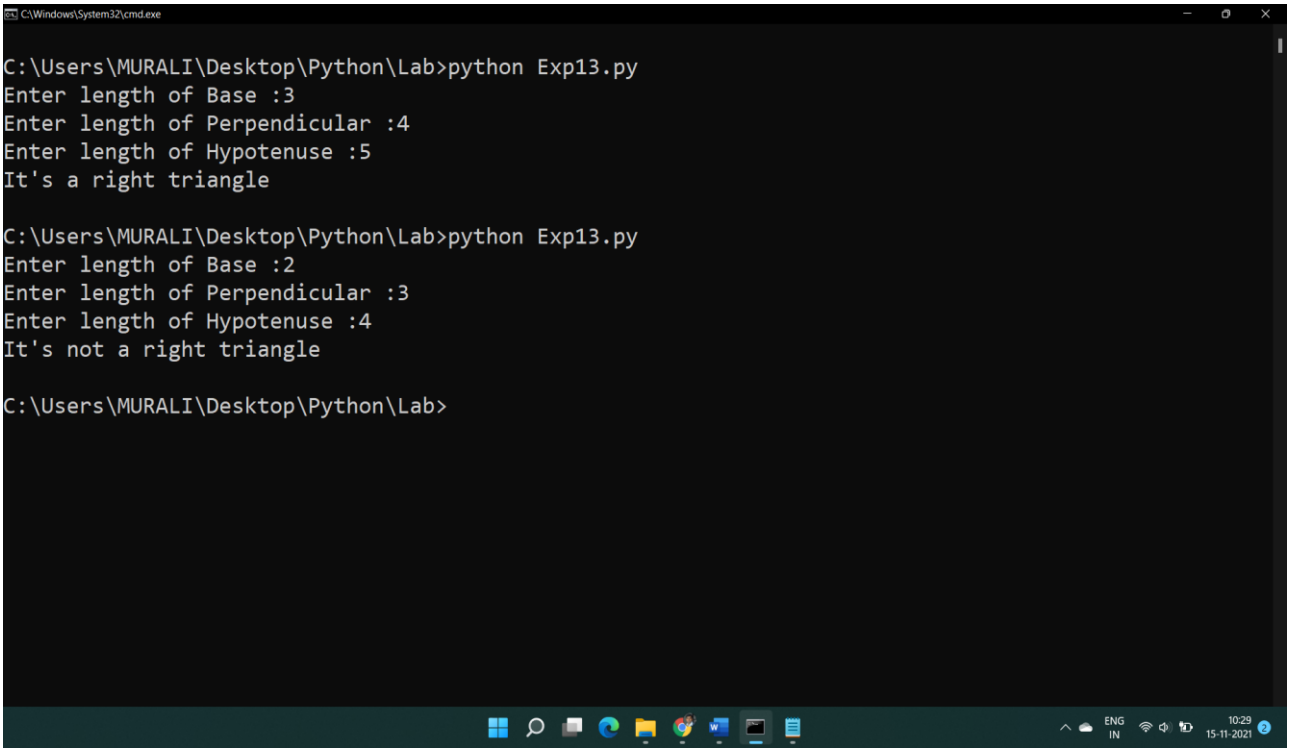
$$Hypotenuse^2=Perpendicular^2+Base^2$$

Source Code:

```
base=float(input("Enter length of Base :"))
perp=float(input("Enter length of Perpendicular :"))
hypo=float(input("Enter length of Hypotenuse :"))

if hypo**2==((base**2)+(perp**2)):
    print("It's a right triangle")
else:
    print("It's not a right triangle")
```

Output:



Ex. No: 14

Aim: *Write a python program to define a module to find Fibonacci Numbers and import the module to another program.*

Theory:

Fibonacci numbers: The Fibonacci Sequence is the series of numbers: 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, ... The next number is found by adding up the two numbers before it.

Python module: A Python module is a file containing Python definitions and statements. A module can define functions, classes, and variables.

Source Code:

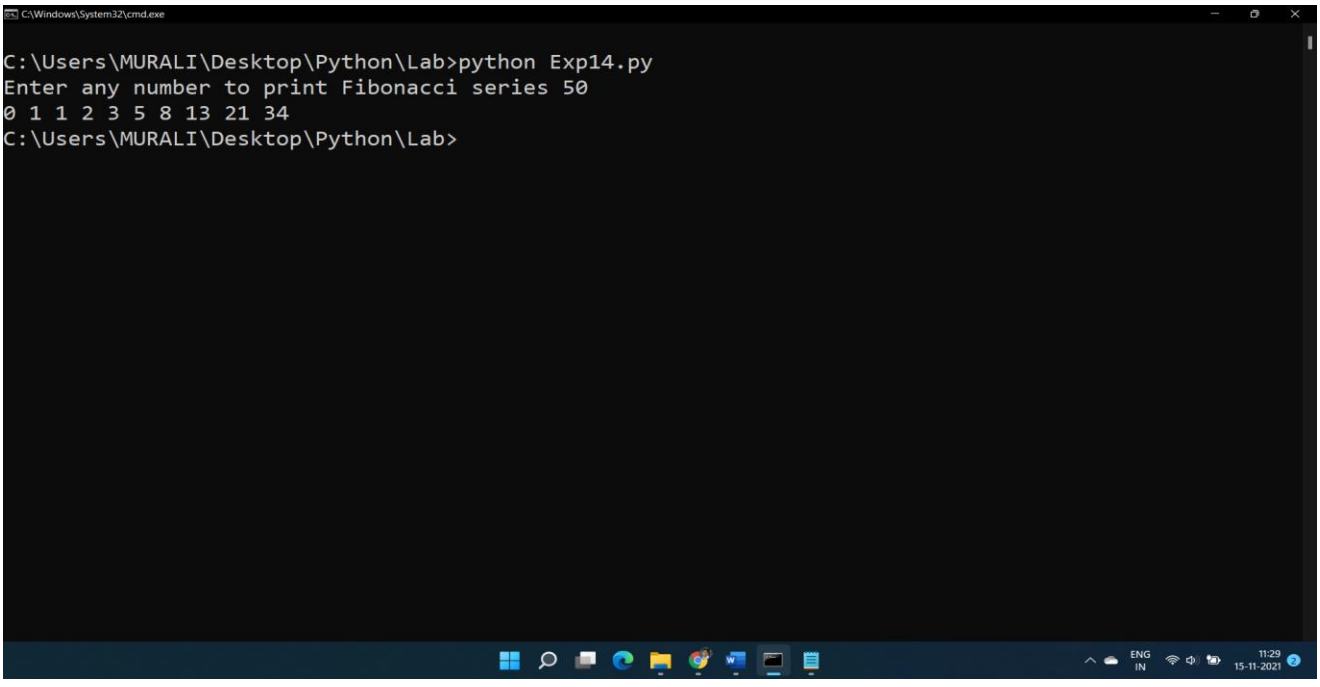
Fibonacci.py

```
def fib(n):
    a, b = 0, 1
    print(a,end =" ")
    while b < n:
        print(b, end =" ")
        a, b = b, a+b
```

Exp14.py

```
import fibonacci
num=int(input("Enter any number to print Fibonacci series "))
fibonacci.fib(num)
```

Output:



Ex. No: 15

Aim: *Write a python program to define a module and import a specific function in that module to another program.*

Source Code:

arth.py

```
def Add(a,b):
    c=a+b
    return c
def Sub(a,b):
    c=a-b
    return c
def Mul(a,b):
    c=a*b
    return c
def Div(a,b):
    c=a/b
    return c
```

Exp15.py

```
from arth import Add
num1=float(input("Enter first Number : "))
num2=float(input("Enter second Number : "))
print("Addition is : ",Add(num1,num2))
print("Subtraction is : ",Sub(num1,num2))
print("Multiplication is : ",Mul(num1,num2))
print("Division is : ",Div(num1,num2))
```

Output: