



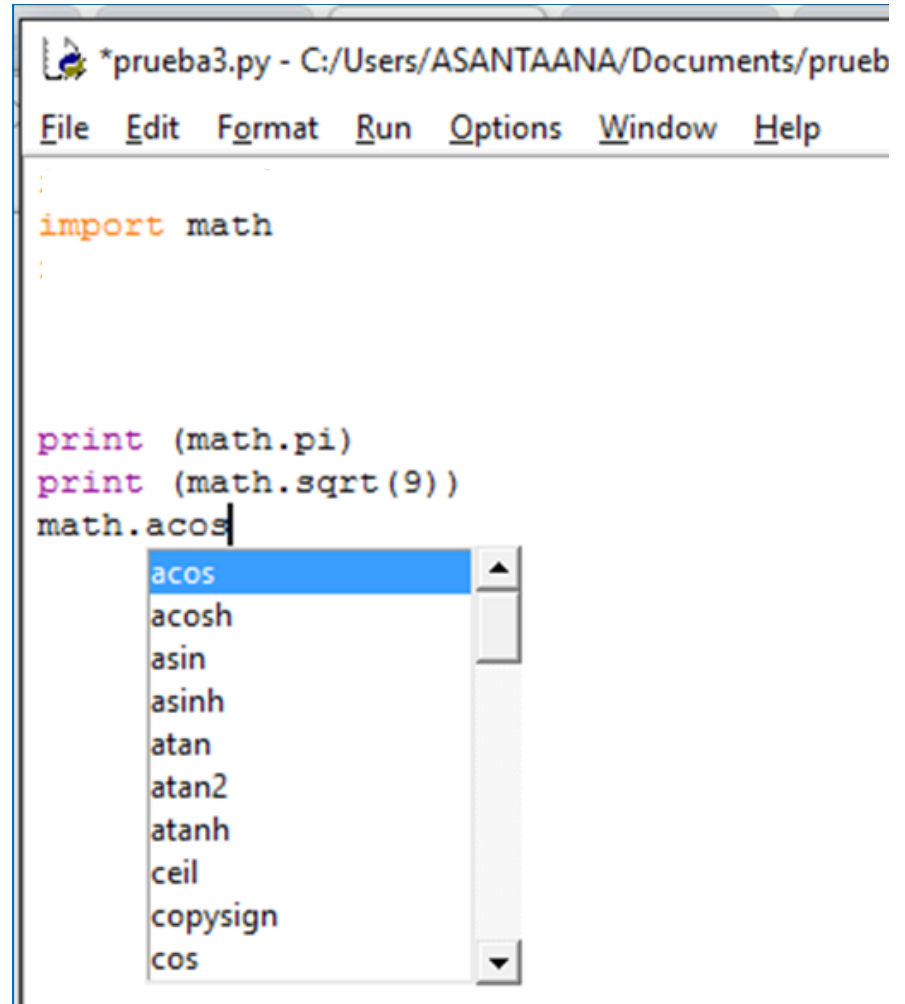
# **Unidad 4**

**Manejo de arreglos y funciones en Python**

# Módulo

Son pedazos de códigos que otras personas han escrito para cumplir tareas comunes tales como generar número aleatorios, realizar operaciones matemáticas.

La manera básica de utilizar un módulo es agregar **import nombre\_modulo** en la parte superior del código y luego utilizarlo



```
*prueba3.py - C:/Users/ASANTAANA/Documents/prueb
File Edit Format Run Options Window Help

import math

print (math.pi)
print (math.sqrt(9))
math.acos
```

The screenshot shows a Python IDE window titled '\*prueba3.py - C:/Users/ASANTAANA/Documents/prueb'. The menu bar includes File, Edit, Format, Run, Options, Window, and Help. The code editor contains the following Python code:

```
import math

print (math.pi)
print (math.sqrt(9))
math.acos
```

A dropdown menu is open below the text 'math.acos', displaying a list of mathematical functions from the 'math' module. The functions listed are: acos, acosh, asin, asinh, atan, atan2, atanh, ceil, copysign, and cos. The 'acos' function is currently selected and highlighted in blue.

# Módulo

Otra clase de **import** si solo se necesita utilizar ciertas funciones de un módulo.

**from** módulo **import** objeto

No se recomienda porque, se pueden confundir las variables del código con las del módulo

**from** math **import** pi

```
print("Este programa hallará el área de una circunferencia.\n")
```

```
r=input("Ingrese el radio de la circunferencia: ")
```

```
area=pi*r**2
```

```
print area
```

```
print pi
```

# Módulo

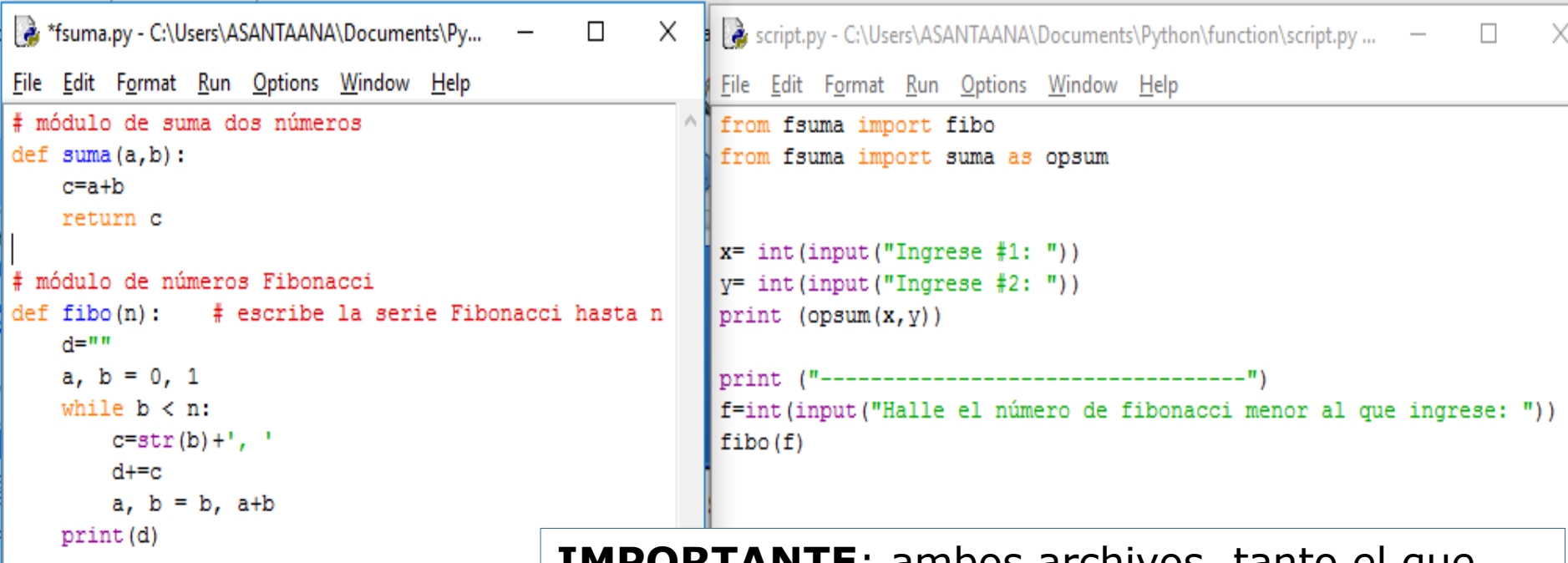
Se puede importar un módulo u objeto bajo un nombre diferente utilizando la palabra clave **as**.

**from** módulo **import** objeto **as** variable

```
from math import sqrt as raiz2  
print (raiz2(64))
```

# Módulo

Se pueden usar las funciones creadas por el programador y utilizarlas como módulos, usando las mismas sentencias de **from ... import ... as ...**



The image shows two side-by-side windows of a Python IDE. The left window, titled '\*fsuma.py', contains two function definitions: 'suma(a,b)' which adds two numbers, and 'fibo(n)' which prints the Fibonacci sequence up to 'n'. The right window, titled 'script.py', imports 'fibo' and 'suma' from 'fsuma' (aliased as 'opsum') and uses them to take user input and calculate the sum of two numbers and the Fibonacci sequence.

```
*fsuma.py - C:\Users\ASANTAANA\Documents\Py...
File Edit Format Run Options Window Help

# módulo de suma dos números
def suma(a,b):
    c=a+b
    return c

# módulo de números Fibonacci
def fibo(n):    # escribe la serie Fibonacci hasta n
    d=""
    a, b = 0, 1
    while b < n:
        c=str(b)+' ', '
        d+=c
        a, b = b, a+b
    print(d)

script.py - C:\Users\ASANTAANA\Documents\Python\function\script.py ...
File Edit Format Run Options Window Help

from fsuma import fibo
from fsuma import suma as opsum

x= int(input("Ingrese #1: "))
y= int(input("Ingrese #2: "))
print (opsum(x,y))

print ("-----")
f=int(input("Halle el número de fibonacci menor al que ingrese: "))
fibo(f)
```

**IMPORTANTE:** ambos archivos, tanto el que contiene las funciones como el que contiene el *scrip* ejecutable, deben estar guardados en la misma carpeta.

# Ejercicio

Utilizando el teorema de seno y el coseno, diseñar un algoritmo que pregunte por la información a ingresar (Lado-Ángulo-Lado o Ángulo-Lado-Ángulo), calcule e imprima los demás lados y ángulos de un triángulo cualquiera.

$$c^2 = a^2 + b^2 - 2*a*b*\cos(C)$$



$$c = \sqrt{a^2 + b^2 - 2*a*b*\cos(C)}$$

$$\sin(A) = \frac{a*\sin(C)}{c} \longrightarrow A = \arcsin\left(\frac{a*\sin(C)}{c}\right)$$