

# Universidad de Buenos Aires

## Facultad de Ingeniería

### 75.40 – Algoritmos y Programación

#### I

Ejercicio obligatorio n°1: Área de polígonos

Curso: 4.

Profesor: Essaya, Diego.

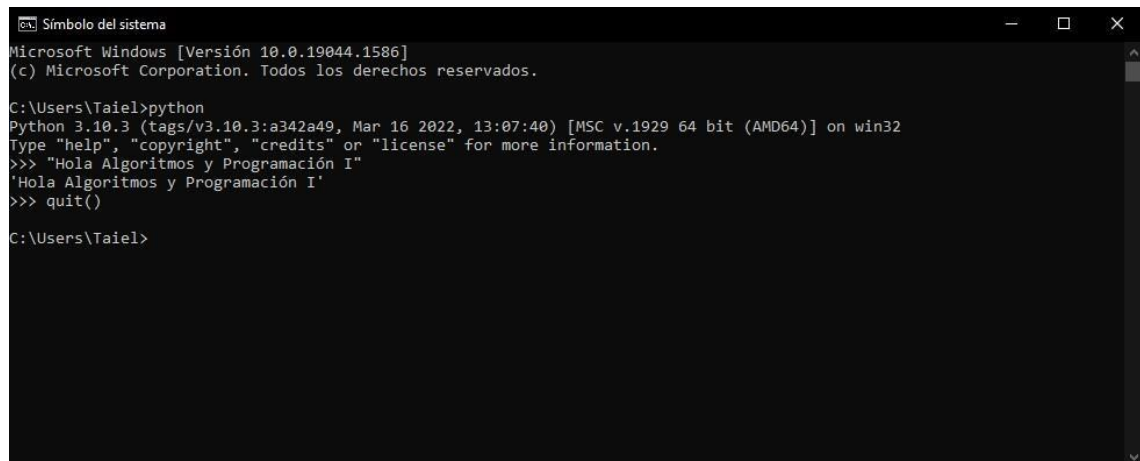
Práctica: Grace.

Alumno: Molina, Taiel Alexis.

Padrón N° 109458.

Ayudate a cargo: Levinas, Alejandro.

## Parte 1.1

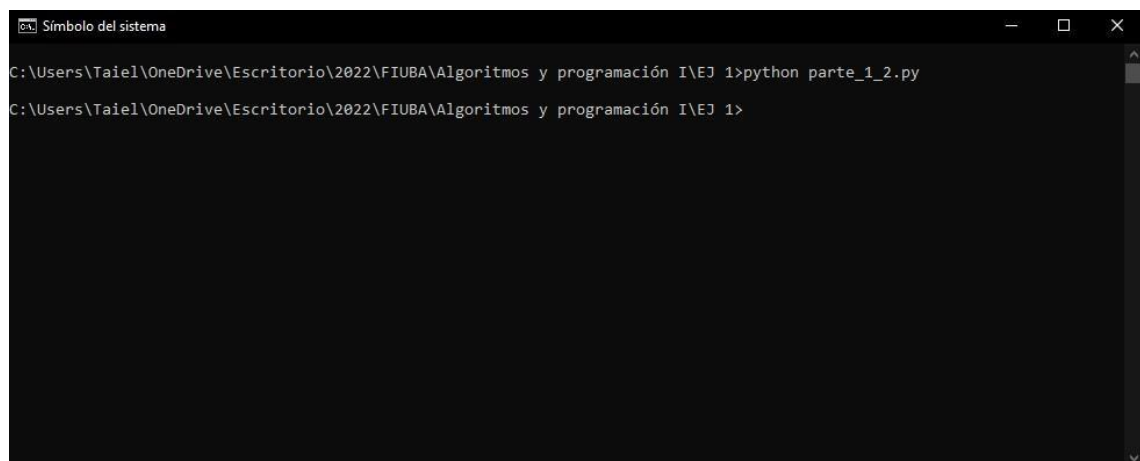


```
Símbolo del sistema
Microsoft Windows [Versión 10.0.19044.1586]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\Taiel>python
Python 3.10.3 (tags/v3.10.3:a342a49, Mar 16 2022, 13:07:40) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> "Hola Algoritmos y Programación I"
'Hola Algoritmos y Programación I'
>>> quit()

C:\Users\Taiel>
```

## Parte 1.2



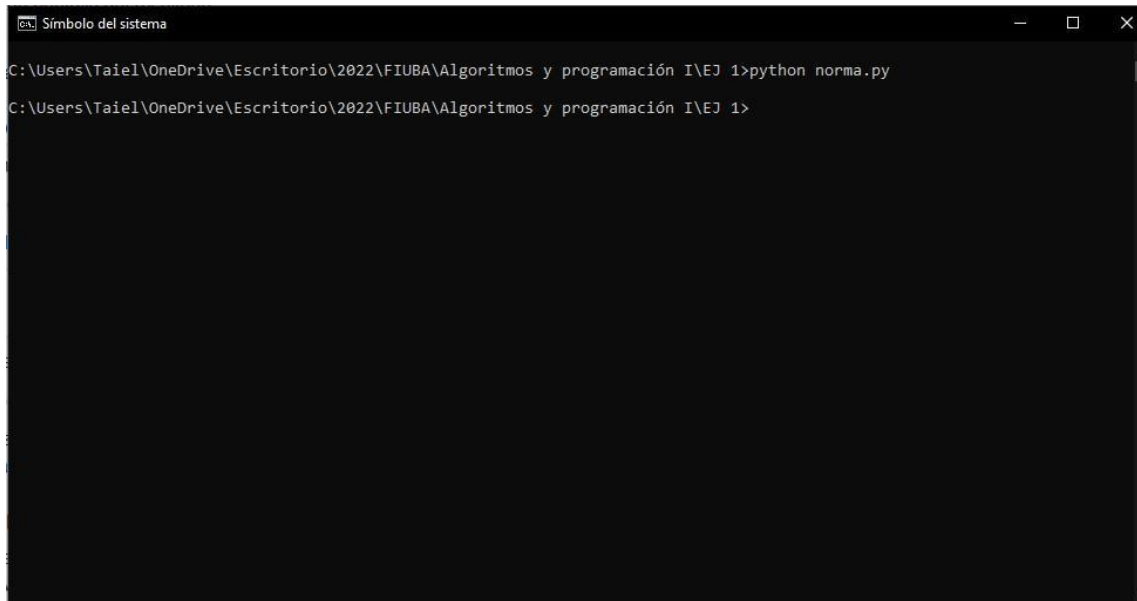
```
Símbolo del sistema

C:\Users\Taiel\OneDrive\Escritorio\2022\FIUBA\Algoritmos y programación I\EJ 1>python parte_1_2.py
C:\Users\Taiel\OneDrive\Escritorio\2022\FIUBA\Algoritmos y programación I\EJ 1>
```

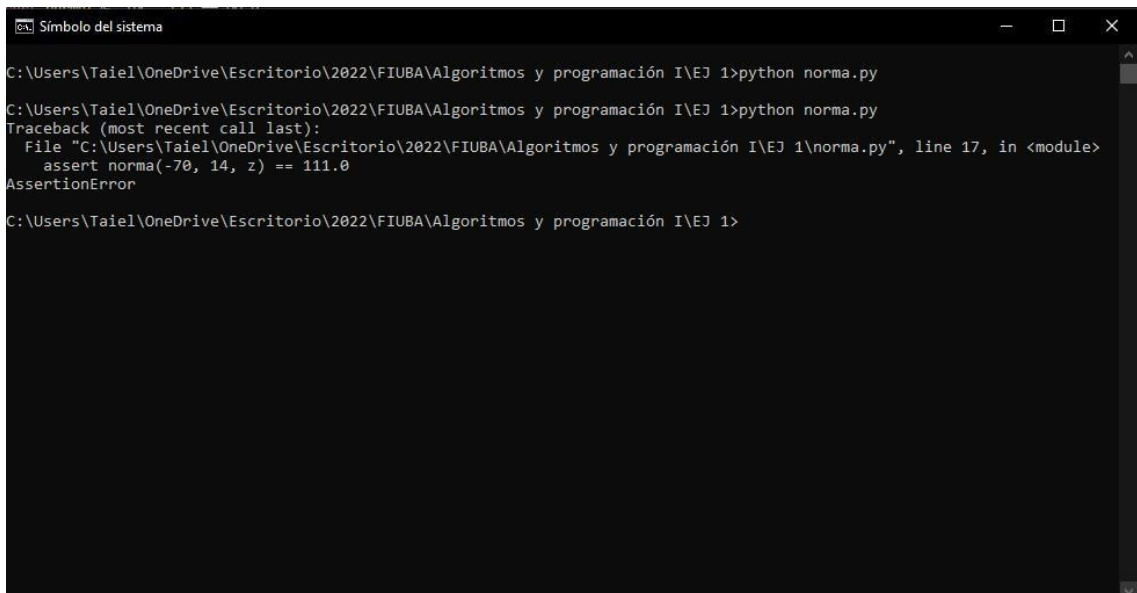
- ¿Qué función debo usar para conseguir el mismo resultado que en la parte 1.1? ¿Por qué en la parte 1.1 vemos el resultado aun sin haber usado esta función?

La función que se debe utilizar para que la consola muestre en pantalla el mensaje de la parte 1.2 es `print()`. En el caso de la parte 1.1 vemos el resultado en la consola gracias al uso del modo interactivo del intérprete de Python.

## Parte 2



```
Símbolo del sistema
C:\Users\Tael\OneDrive\Escritorio\2022\FIUBA\Algoritmos y programación I\EJ 1>python norma.py
C:\Users\Tael\OneDrive\Escritorio\2022\FIUBA\Algoritmos y programación I\EJ 1>
```



```
Símbolo del sistema
C:\Users\Tael\OneDrive\Escritorio\2022\FIUBA\Algoritmos y programación I\EJ 1>python norma.py
C:\Users\Tael\OneDrive\Escritorio\2022\FIUBA\Algoritmos y programación I\EJ 1>python norma.py
Traceback (most recent call last):
  File "C:\Users\Tael\OneDrive\Escritorio\2022\FIUBA\Algoritmos y programación I\EJ 1\norma.py", line 17, in <module>
    assert norma(-70, 14, z) == 111.0
AssertionError
C:\Users\Tael\OneDrive\Escritorio\2022\FIUBA\Algoritmos y programación I\EJ 1>
```

1. ¿Cuál es la salida del programa?
2. ¿Podemos saber en qué línea se generó el error? ¿Cómo?
3. ¿Qué hace la instrucción assert?
4. Solucionar el problema. Hallar el valor de z para que ya no de error.

1. La salida del programa es la función return que indica el final y devuelve el valor de esta.
2. Sí, se puede saber en qué línea se generó el error. En la segunda captura de pantalla, en la consola de Windows se puede leer que dice: "File "C:\Users\...\norma.py" line 17, in <module> assert norma(-70, 14, z) == 111". Si hubiésemos ejecutado el programa desde VSCode o similar, también el mismo

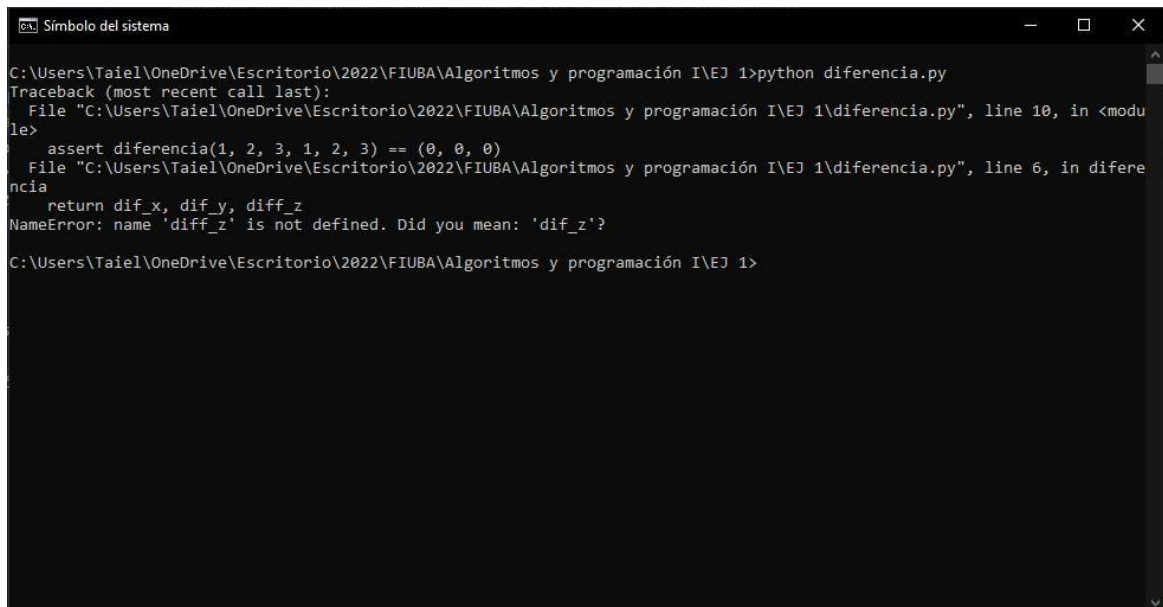
editor de texto te devuelve un mensaje similar (o exactamente el mismo), explicando en qué línea se encuentra el error.

3. La instrucción `assert` nos permite realizar comprobaciones. En el caso de que la expresión de un `assert` sea falsa, nos devolverá un mensaje explicando cual `assert` falló, y ahí podremos revisar nuestro error.
4. Norma de un vector en R3:

$$\vec{u} = (u_1, u_2, u_3)$$
$$|\vec{u}| = \sqrt{u_1^2 + u_2^2 + u_3^2}$$

$$\begin{aligned}\sqrt{(-70)^2 + (14)^2 + z^2} &= 111 \\ 4900 + 196 + z^2 &= 111^2 \\ z^2 &= 12321 - 196 - 4900 \\ |z| &= \sqrt{7225} \\ z &= \pm 85\end{aligned}$$

### Parte 3



```
Símbolo del sistema
C:\Users\Taiel\OneDrive\Escritorio\2022\FIUBA\Algoritmos y programación I\EJ 1>python diferencia.py
Traceback (most recent call last):
  File "C:\Users\Taiel\OneDrive\Escritorio\2022\FIUBA\Algoritmos y programación I\EJ 1\diferencia.py", line 10, in <module>
    le> assert diferencia(1, 2, 3, 1, 2, 3) == (0, 0, 0)
  File "C:\Users\Taiel\OneDrive\Escritorio\2022\FIUBA\Algoritmos y programación I\EJ 1\diferencia.py", line 6, in difere
ncia
    return dif_x, dif_y, diff_z
NameError: name 'diff_z' is not defined. Did you mean: 'dif_z'?

C:\Users\Taiel\OneDrive\Escritorio\2022\FIUBA\Algoritmos y programación I\EJ 1>
```

- ¿Se detectó algún error? ¿Cuál era? ¿Qué significa? ¿Qué línea estaba fallando?

El error que se detecta es que en la línea 6 cuando la instrucción `return` pide que el programa devuelva los valores de `dif_x`, `dif_y` y `dif_z`, hay un error de tipeo y dice `diff_z`, invocando a una variable que no existe.

Es importante que los nombres de la función y las variables de un programa sean representativos porque la idea es que cuando otra persona lea el programa pueda entenderlo. Quizás al ser un programa corto, no hay grandes problemas para captar la información, pero el día de mañana, ese programa pueden ser miles de líneas de código y el mal nombramiento de funciones/variables llevaría a una gran dificultad para entender lo que quiso realizar el programador.

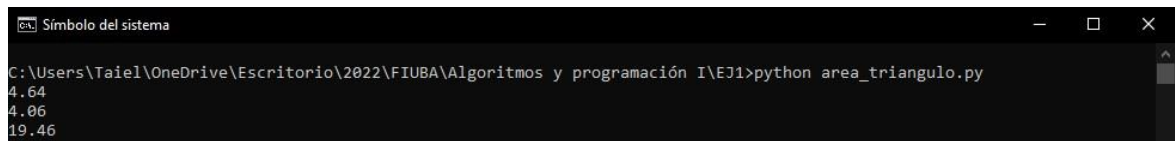
- ¿Se puede escribir el cuerpo de la función en una línea? ¿Cómo?

Sí, se puede escribir el cuerpo de la función en una línea. En vez de declarar cada coordenada del producto vectorial como una variable y luego insertarlas en el return, lo que se debe hacer es poner las operaciones directamente en dicha instrucción.

## Parte 5

Los puntos en el espacio con los que se realizaron las pruebas son:

- I.  $A = (5, 8, -1)$ ;  $B = (-2, 3, 4)$ ;  $C = (-3, 3, 0)$  [Ejemplo de la guía]
- II.  $A = (0, 0, -3)$ ;  $B = (4, 2, 0)$ ;  $C = (3, 3, 1)$
- III.  $A = (-2, 0, 2)$ ;  $B = (-5, 2, 0)$ ;  $C = (6, -3, 7)$



```
Símbolo del sistema
C:\Users\Taie1\OneDrive\Escritorio\2022\FIUBA\Algoritmos y programación I\EJ1>python area_triangulo.py
4.64
4.06
19.46
```

- ¿Cuál es la importancia de reutilizar funciones?

La importancia de reutilizar funciones es simplificar el código lo máximo posible, evitar la repetición de código, y lograr que el mismo sea mas mantenible a través del tiempo.