**Ejercicio:** El módulo deberá contener la función main(), tal como está definida en el archivo original (no debe ser modificada) y la definición de la siguiente función debidamente “documentada” y probada:

- Una función denominada suma\_multi que reciba por parámetros tres números naturales (que son distintos entre sí), determine “el menor” y “el mayor”, y devuelva como resultado la suma de los múltiplos del número restante, que se encuentren en la secuencia de enteros comprendida entre “el menor” y “el mayor” (incluidos).

Para realizar las pruebas, deberás ejecutar la función main(), porque incluye los resultados esperados para siete casos de prueba.

Considerá los primeros 14 ítems de la Matriz de Autoevaluación para revisar la calidad de tu producción.

def suma\_multi(a, b, c):

"""Recibe por parámetro tres números naturales distintos entre sí.

Determina "el menor", "el mayor" y el restante.

Devuelve como resultado la suma de los múltiplos del número restante,

que se encuentren en la secuencia de enteros comprendida entre

"el menor" y "el mayor" (incluidos)

"""

if a > b and a > c:

mayor = a

if b < c:

menor, restante = b, c

else:

menor, restante = c, b

elif b > a and b > c:

mayor = b

if a < c:

menor, restante = a, c

else:

menor, restante = c, a

else:

mayor = c

if a < b:

menor, restante = a, b

else:

menor, restante = b, a

mayor = mayor + 1

suma = 0

for iterador in range (menor, mayor):

if iterador % restante == 0:

suma = suma + iterador

return suma

def main():

"""Pruebas de función de Unidad 4 """

uno = 3

dos = 7

tres = 21

print ('Resultados esperados de función suma\_multi()')

print ('Suma de múltiplos con 3 7 21 es: 42')

print ('Suma de mútliplos con 7 3 21 es: 42')

print ('Suma de mútliplos con 3 21 7 es: 42')

print ('Suma de mútliplos con 21 3 7 es: 42')

print ('Suma de mútliplos con 7 21 3 es: 42')

print ('Suma de mútliplos con 21 7 3 es: 42')

print ('\nResultados reales de función suma\_multi()')

print ('Suma de mútliplos con',uno,dos,tres,'da =>',suma\_multi(uno,dos,tres))

print ('Suma de mútliplos con',dos,uno,tres,'da =>',suma\_multi(dos,uno,tres))

print ('Suma de mútliplos con',uno,tres,dos,'da =>',suma\_multi(uno,tres,dos))

print ('Suma de mútliplos con',tres,uno,dos,'da =>',suma\_multi(tres,uno,dos))

print ('Suma de mútliplos con',dos,tres,uno,'da =>',suma\_multi(dos,tres,uno))

print ('Suma de mútliplos con',tres,dos,uno,'da =>',suma\_multi(tres,dos,uno))

print ('\nResultado esperado de función suma\_multi() con datos erróneos')

print ('Suma de mútliplos con 3 3 3 es: 3')

print ('\nResultado real de función suma\_multi() con datos erróneos')

print ('Suma de mútliplos con',uno,uno,uno,'da =>',suma\_multi(uno,uno,uno))

# help(suma\_multi)

main()

**Prueba:**

======= RESTART: C:\Users\Daniel…

Resultados esperados de función suma\_multi()

Suma de múltiplos con 3 7 21 es: 42

Suma de mútliplos con 7 3 21 es: 42

Suma de mútliplos con 3 21 7 es: 42

Suma de mútliplos con 21 3 7 es: 42

Suma de mútliplos con 7 21 3 es: 42

Suma de mútliplos con 21 7 3 es: 42

Resultados reales de función suma\_multi()

Suma de mútliplos con 3 7 21 da => 42

Suma de mútliplos con 7 3 21 da => 42

Suma de mútliplos con 3 21 7 da => 42

Suma de mútliplos con 21 3 7 da => 42

Suma de mútliplos con 7 21 3 da => 42

Suma de mútliplos con 21 7 3 da => 42

Resultado esperado de función suma\_multi() con datos erróneos

Suma de mútliplos con 3 3 3 es: 3

Resultado real de función suma\_multi() con datos erróneos

Suma de mútliplos con 3 3 3 da => 3

>>>

>>> help(suma\_multi)

Help on function suma\_multi in module \_\_main\_\_:

suma\_multi(a, b, c)

Recibe por parámetro tres números naturales distintos entre sí.

Determina "el menor", "el mayor" y el restante.

Devuelve como resultado la suma de los múltiplos del número restante,

que se encuentren en la secuencia de enteros comprendida entre

"el menor" y "el mayor" (incluidos)

>>>

*La función* ***está documentada*** *y* ***produce los resultados esperados****.*

*Utiliza correctamente “expresiones booleanas” y alternativas de decisión completas y bien ordenadas.*

*Utiliza correctamente ciclo “for” por índices y limita correctamente la condición de iteración.*

*En síntesis, es un muy buen trabajo.*

***No se requiere nuevo envío.*** *Por favor, avisanos cuando hayas descargado este documento y conservalo en tu biblioteca de aprendizajes (las devoluciones pueden ser requeridas más adelante).*

*Saludos,*

*Daniel S. R.*