

# Guia 1: funciones

## Ejercicio 1

Escriba 4 funciones que simule una calculadora básica, esta puede realizar operaciones de suma, resta, multiplicación y división. Cada operación debe estar encapsulada en una función que recibe 2 argumentos. Las funciones deben llamarse: `resta(a,b)`, `suma(a, b)`, `multiplicacion(a,b)` y `division(a,b)`.

Al llamar las funciones con 2 argumentos, debe entregar el resultado de la operación.

Ejemplo:

```
>> print(suma(2,3))
>> 5
```

## Ejercicio 2

El riesgo de que una persona sufra enfermedades coronarias depende de su edad y su índice de masa corporal:

	edad < 45	edad ≥ 45
IMC < 22.0	bajo	medio
IMC ≥ 22.0	medio	alto

El índice de masa corporal es la división entre el peso del individuo en kilos y el cuadrado de su estatura en metros.

Escriba una función llamada `IMC(peso, estatura, edad)` que reciba 3 argumentos, asociados a el peso en kilos, estatura en metros y edad y devuelva la condición de riesgo de la persona.

Ejemplo:

```
>> print(IMC(70, 1.8, 20))
>> Bajo
```

## Ejercicio 3

Escriba un función `ordenar(lista_numeros)`, que reciba un argumento, asociados a una lista de números enteros y devuelva la lista de números ordenada de mayor a menor.

Ejemplo:

```
>> print(ordenar([23, 1, 47]))
>> [47, 23, 1]
```

## Ejercicio 4

En la empresa SEGURITO venden tres tipos de póliza: A,B, C. Desarrolle un programa que permita recibir el sueldo y número de hijos (código main) y llame a una función calculo(sueldo, num\_hijos) que recibe 2 argumentos asociados al sueldo y número de hijos y devuelva el tipo de póliza que le corresponde según los siguientes criterios:

- Póliza A: Si el sueldo es menor o igual a \$250.000 y tiene a lo más un hijo.
- Póliza B: Si el sueldo es mayor a \$250.000 y NO tiene hijos.
- Póliza C: Si el sueldo es mayor a \$250.000 y tiene entre 1 y 5 hijos
- Póliza D: si no le corresponde ninguna de las pólizas anteriores.

Nota: Debe tener un código main, una función definida y desde el código main debe llamar a la función solicitada

## Ejercicio 5

Los tres lados  $a, b$  y  $c$  de un triángulo deben satisfacer la desigualdad triangular: cada uno de los lados no puede ser más largo que la suma de los otros dos.

Escriba un programa que reciba desde teclado los 3 lados de un triángulo (código main) y llame a una función verificar\_triángulo(lista) que reciba una lista con los 3 lados ingresados y devuelva un mensaje:

- Si acaso el triángulo es invalido; y
- Si no lo es, qué tipo de triángulo es (Escaleno, Isósceles o Equilátero).

Considere que un triángulo Escaleno tiene todos sus lados distintos, un triángulo Isósceles tiene 2 lados iguales y un triángulo Equilátero tiene sus 3 lados iguales.

Nota: Debe tener un código main, una función definida y desde el código main debe llamar a la función solicitada.

## Ejercicio 6

Escriba una función llamada cuenta\_letra(palabra) que recibe una palabra en formato string y retorne una lista con el número de vocales y consonantes que tiene la palabra.

Ejemplo:

```
>> print(cuenta_letra('programacion'))  
>> [5, 7]
```

## Ejercicio 7

Escriba una función llamada division\_correo(correo) que reciba una dirección de correo electrónico y entregue una lista con la parte antes del arroba (@) y luego la parte después del arroba.

Ejemplo:

```
>> print(division_correo('programacion_algoritmos@duoc.cl'))  
>> ['programacion_algoritmos', 'duoc.cl']
```