

# Funciones

## Funciones de uno o varios parámetros

Jonatan Gómez Perdomo, Ph. D.

[jgomezpe@unal.edu.co](mailto:jgomezpe@unal.edu.co)

Arles Rodríguez, Ph.D.

[aerodriguezp@unal.edu.co](mailto:aerodriguezp@unal.edu.co)

Camilo Cubides, Ph.D. (c)

[eccubidesg@unal.edu.co](mailto:eccubidesg@unal.edu.co)

Carlos Andres Sierra, M.Sc.

[casierrav@unal.edu.co](mailto:casierrav@unal.edu.co)

Research Group on Artificial Life – Grupo de investigación en vida artificial – (Alife)

Computer and System Department

Engineering School

Universidad Nacional de Colombia

# Agenda

## 1 Funciones de un parámetro de entrada

- Cuadrado de un número
- Área de un círculo
- Conversión de tipos de datos numéricos (*typecasting*)

## 2 Funciones con más de un parámetro de entrada

- Área de un rectángulo
- Ley de Coulomb

## 3 Problemas



# Definición de funciones de un parámetro

En programación, así como en matemáticas, para las funciones definidas como  $f : A \rightarrow B$ , al conjunto  $A$  se le denomina dominio y al conjunto  $B$  como rango. A partir de estos objetos se construye el encabezado de las funciones de programación.

Sobre esta función se tiene que  $f$  corresponde al nombre de la función, el conjunto  $A$  corresponde al tipo de los argumentos de dicha función y el conjunto  $B$  que es el rango corresponderá al valor de retorno de dicha función.



# Agenda

## 1 Funciones de un parámetro de entrada

- Cuadrado de un número
- Área de un círculo
- Conversión de tipos de datos numéricos (*typecasting*)

## 2 Funciones con más de un parámetro de entrada

- Área de un rectángulo
- Ley de Coulomb

## 3 Problemas



# Cuadrado de un número I

## Ejemplo

Se definirá una función que eleve un número al cuadrado. Para expresar una función que calcule esta operación, en primera instancia se construye la expresión  $f : \mathbb{R} \rightarrow \mathbb{R}$  que define la función que tiene como entrada (dominio) un número real y como salida (rango) un número real. La declaración de la función junto con su cuerpo quedará de la siguiente forma

$$f : \mathbb{R} \rightarrow \mathbb{R}$$

$$(x) \mapsto x^2$$

Esta función también podría notarse como

$$f = \{(x, x^2) : (x \in \mathbb{R})\}.$$



# Cuadrado de un número II

## Ejemplo (continuación)

Dicha función se traduce en lenguaje Python paso a paso de la siguiente manera:

- Primero se escribe la palabra reservada `def`.

```
def
```

- Posteriormente se escribe el nombre de la función `f`.

```
def f
```

- Entre paréntesis se coloca el nombre de la variable del dominio (parámetro). En este caso solamente se tiene la variable `x` quedando el texto `(x)` y se coloca a continuación el símbolo de dos puntos `:`.

```
def f(x):
```

# Cuadrado de un número III

## Ejemplo (continuación)

- En la siguiente línea se escribe el cuerpo de la función, teniendo en cuenta que debe haber una y exactamente una tabulación aplicada con respecto a la línea superior<sup>†</sup>. Dicho cuerpo de la función corresponde al algoritmo ó computo que se desea calcule la función<sup>‡</sup>. Para retornar la imagen de la función se utiliza la palabra reservada `return`, y a continuación la operación que genera la imagen de la función, como se puede observar a continuación

```
def f(x):  
    return x * x
```

<sup>†</sup>A esto se le conoce como la *indentación* o *sangrado* del código.

<sup>‡</sup>El cuerpo de la función es el ámbito o *scope* en el cual las variables de la función pueden ser usadas, es decir, es el entorno en el cual estas existen, pues ellas están definidas localmente.



# Cuadrado de un número IV

## Ejemplo (continuación)

- Para invocar una función, simplemente se escribe el nombre de la función y a continuación dentro de un paréntesis el argumento en el cual se desea evaluar la función, como se muestra a continuación cuando el argumento es el valor 2

```
f(2)
```

- Para el caso en el que se quiera observar el resultado de evaluar la función, se puede invocar la función `print` sin tabulación y se evalúa con el valor obtenido al aplicar la función `f(2)`, como se observa a continuación

```
print(f(2))
```





# Cuadrado de un número V

## Ejemplo (continuación)

Para el caso del cuadrado, este se puede calcular directamente, pues la potencia es una operación matemática básica en Python; por lo que se puede escribir el cuadrado como el resultado de realizar la operación `x ** 2`, como se muestra a continuación

```
def f(x):  
    return x ** 2  
  
print(f(2))
```



# Cuadrado de un número VI

## Ejemplo (continuación)

Otra posible escritura de esta función podría obtenerse almacenando el valor del cálculo `x ** 2` en una variable auxiliar a la que se le aplique una y exactamente una tabulación con respecto a la línea superior, y en la siguiente línea se retorne la variable, manteniendo la alineación actual, como se muestra a continuación

```
def f(x):  
    y = x ** 2  
    return y
```

Esto es similar al caso cuando se utiliza la notación funcional

$$f(x) = y$$

donde  $x$  es la variable independiente y  $y$  es la variable dependiente.

# Agenda

## 1 Funciones de un parámetro de entrada

- Cuadrado de un número
- Área de un círculo
- Conversión de tipos de datos numéricos (*typecasting*)

## 2 Funciones con más de un parámetro de entrada

- Área de un rectángulo
- Ley de Coulomb

## 3 Problemas



# Área de un círculo I

## Ejemplo

Para el desarrollo de esta función lo primero es determinar el nombre. La función se llamará *area\_circulo* cuyo dominio es el conjunto de los números reales (para el radio) y cuyo rango pertenece al conjunto de los números reales (el valor de retorno que corresponde al área del círculo).

Teniendo en cuenta que el algoritmo para el cálculo del área de un círculo depende del valor de su radio, entonces, el área del círculo está dada por la expresión  $A_c = \pi r^2$ , donde las variables están definidas así:

$r :=$  Radio del círculo

$A_c :=$  Área del círculo de radio  $r$



# Área de un círculo II

## Ejemplo (continuación)

entonces, el planteamiento matemático de la función solicitada será el siguiente

$$\begin{aligned} \text{area\_circulo} : \mathbb{R} &\rightarrow \mathbb{R} \\ (r) &\mapsto \pi r^2 \end{aligned}$$

Para utilizar el valor de  $\pi$  se debe tomar una aproximación, ya que  $\pi$  es un número irracional, esta aproximación podría ser el valor 3.14159265.

Para el caso del cuadrado del radio, este se puede calcular como un producto o como una potencia, por lo que se puede escribir como  $r * r$  o  $r ** 2$ .



# Área de un círculo III

## Ejemplo (continuación)

Esta función se traduce al lenguaje Python paso a paso de la siguiente forma:

- Primero se escribe la palabra reservada `def`.

```
def
```

- Posteriormente se escribe el nombre de la función `area_circulo`.

```
def area_circulo
```

obsérvese que tanto las palabras `area` como `circulo` en el nombre de la función se escribieron sin tilde, se recomienda escribir los nombres de las funciones únicamente con caracteres del alfabeto inglés, esto para evitar problemas de incompatibilidad entre plataformas.



# Área de un círculo IV

## Ejemplo (continuación)

- Entre paréntesis se coloca el nombre de la variable del dominio (parámetro). En este caso solamente se tiene la variable  $r$  quedando el texto (r) y se coloca a continuación el símbolo de dos puntos :

```
def area_circulo(r):
```

- En las siguientes líneas se escribe el cálculo del área del círculo y se retorna el valor calculado.

```
    def area_circulo(r):  
        return 3.14159265 * r ** 2
```

aquí la potencia (\*\*) tiene mayor prioridad que la multiplicación (\*).



# Área de un círculo V

## Ejemplo (continuación)

- Para invocar una función, simplemente se escribe el nombre de la función y a continuación dentro de un paréntesis el argumento en el cual se desea evaluar la función, como se muestra a continuación cuando el argumento es el valor 3

```
area_circulo(3)
```

- Para el caso en el que se quiera observar el resultado de evaluar la función, se puede invocar la función `print` sin tabulación y se evalúa con el valor obtenido al aplicar la función `area_circulo(3)`, como se observa a continuación

```
print(area_circulo(3))
```





# Área de un círculo VI

## Ejemplo (continuación)

Otra posible escritura de esta función podría obtenerse almacenando el valor del área del círculo en una variable auxiliar a la que se le aplique una y exactamente una tabulación con respecto a la línea superior, y en la siguiente línea se retorne la variable, manteniendo la alineación actual, como se muestra a continuación

```
def area_circulo(r):  
    area = 3.14159265 * r ** 2  
    return area
```

Esto es similar al caso cuando se utiliza la notación funcional

$$\textit{area\_circulo}(r) = \textit{area}$$

donde  $r$  es la variable independiente y  $area$  es la variable dependiente.

# Área de un círculo VII

## Ejemplo (continuación)

Otra posible escritura de esta función en donde se utilice la aproximación a la constante  $\pi$  suministrada por Python, es la siguiente

```
import math

def area_circulo(r):
    area = math.pi * r ** 2
    return area
```

En este caso se importó el módulo `math`, y para obtener el valor aproximado de la constante matemática  $\pi$  se utiliza la instrucción `math.pi`.



# Área de un círculo VIII

## Ejemplo (continuación)

Otra posible escritura de esta función en donde se importe únicamente la aproximación a la constante  $\pi$  suministrada por Python, es la siguiente

```
from math import pi

def area_circulo(r):
    area = pi * r ** 2
    return area
```

En este caso se importó únicamente el valor aproximado de la constante matemática  $\pi$ , de esta manera la variable `pi` almacena el valor aproximado de  $\pi$ . Esto se hace para no tener que cargar todo el módulo `math` a memoria. El valor que almacena la variable  $\pi$  puede cambiar durante la ejecución del programa si se le asigna un nuevo valor.

# Agenda

## 1 Funciones de un parámetro de entrada

- Cuadrado de un número
- Área de un círculo
- Conversión de tipos de datos numéricos (*typecasting*)

## 2 Funciones con más de un parámetro de entrada

- Área de un rectángulo
- Ley de Coulomb

## 3 Problemas



# De entero a real I

**De entero a real:** dado un dato o una variable de tipo entero, si se opera o se asigna el dato o la variable con un dato o una variable de tipo real, entonces al realizar la operación o la asignación, el dato entero se convierte (se promueve) a un dato de tipo real de forma automática, simplemente agregándole la parte decimal “.0”.



# De entero a real II

## Ejemplo

Para las asignaciones

```
n = 1  
x = float(n)  
y = float(0)  
z = float(-2)
```

se tiene que las variables almacenan los valores:

n	1
x	1.0
y	0.0
z	-2.0



# De entero a real III

## Ejemplo

Las siguientes operaciones son equivalentes

- $-2 + 1.0 \Leftrightarrow -2.0 + 1.0$
- $0.0 * 5 \Leftrightarrow 0.0 * 5.0$
- $(5 // 2) * 2.0 \Leftrightarrow 2 * 2.0 \Leftrightarrow 2.0 * 2.0$
- $(5.0 / 2) * 2 \Leftrightarrow (5.0 / 2.0) * 2 \Leftrightarrow 2.5 * 2 \Leftrightarrow 2.5 * 2.0$



# De real a entero I

**De real a entero:** dado un dato o una variable de tipo real, si se aplica la función `int( )` al dato o la variable, entonces el valor del dato o la variable se convierte (se promueve) a un dato de tipo entero, simplemente elimina la parte decimal del real y dejando la parte entera.





# De real a entero II

## Ejemplo

Para las siguientes asignaciones en Python

```
x = 1.0  
y = -2.5  
n = int(x)  
m = int(y)  
p = int(3.14159265)
```

se tiene que las variables almacenan los valores:

x	1.0
y	-2.5
n	1
m	-2
p	3

# De real a entero III

## Ejemplo

Las siguientes operaciones son equivalentes

- $2 * \text{int}(2.5) \Leftrightarrow 2 * 2$
- $\text{int}(-3.14) * \text{int}(5.5) \Leftrightarrow -3 * 5$
- $2 / (2.5 - \text{int}(2.5)) \Leftrightarrow 2 / (2.5 - 2) \Leftrightarrow 2 / (0.5)$   
 $\Leftrightarrow 2 / 0.5 \Leftrightarrow 2.0 / 0.5$



# De real a entero IV

## Ejemplo

Las siguientes sentencias son equivalentes

```
x = 0.4  
y = 2.5  
n = int(x) * int(y)
```



```
x = 0.4  
y = 2.5  
n = 0 * 2
```



# Agenda

## 1 Funciones de un parámetro de entrada

- Cuadrado de un número
- Área de un círculo
- Conversión de tipos de datos numéricos (*typecasting*)

## 2 Funciones con más de un parámetro de entrada

- Área de un rectángulo
- Ley de Coulomb

## 3 Problemas



# Agenda

- 1 Funciones de un parámetro de entrada
  - Cuadrado de un número
  - Área de un círculo
  - Conversión de tipos de datos numéricos (*typecasting*)
- 2 Funciones con más de un parámetro de entrada
  - Área de un rectángulo
  - Ley de Coulomb
- 3 Problemas



# Área de un rectángulo I

Las funciones están definidas en la forma  $f : A \rightarrow B$ , siendo  $A$  y  $B$  conjuntos. Esta definición nos permite utilizar productos cartesianos generalizados como dominio en la declaración de las funciones.

## Ejemplo

La función que calcula el área de un rectángulo es una función que tiene como parámetros de entrada el ancho y el largo del rectángulo que son valores de tipo real y retorna como salida el área del rectángulo. Lo que puede escribirse como

$$area\_rectangulo : \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}.$$



# Área de un rectángulo II

## Ejemplo (continuación)

Para el cálculo del área de un rectángulo es necesario conocer el largo y el ancho del rectángulo, a partir de los cuales el área del rectángulo está dada por la expresión  $A_r = l * a$ , donde las variables están definidas así:

$l :=$  Largo del rectángulo

$a :=$  Ancho del rectángulo

$A_r :=$  Área del rectángulo de largo  $l$  y ancho  $a$

entonces, la función matemática queda definida de la siguiente forma

$$\text{area\_rectangulo} : \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}$$

$$(l, a) \mapsto l * a$$



# Área de un rectángulo III

## Ejemplo (continuación)

Para traducir a lenguaje Python esta función, se utilizan reglas similares de traducción:

- Primero se escribe la palabra reservada `def`.

```
def
```

- Posteriormente se escribe el nombre de la función `area_rectangulo`.

```
def area_rectangulo
```





# Área de un rectángulo IV

## Ejemplo (continuación)

- Entre paréntesis se colocan los nombres ordenados de las variables del dominio (parámetros) separados por comas. En este caso se tienen las variables  $l$  y  $a$  correspondientes al largo y al ancho del rectángulo respectivamente y se coloca a continuación el símbolo de dos puntos  $:$ .

```
def area_rectangulo(l, a):
```

- En la siguiente línea se escribe el cálculo del área del rectángulo y se retorna el valor calculado, aplicando una y exactamente una tabulación con respecto a la línea superior

```
    return l * a
```



# Área de un rectángulo V

## Ejemplo (continuación)

Otra posible escritura de la función puede ser

```
def area_rectangulo(l, a):  
    area = l * a  
    return area
```

Esto es similar al caso cuando se utiliza la notación funcional

$$\text{area\_rectangulo}(l, a) = \text{area}$$

para expresar la función, las variables independientes y la dependiente.



# Área de un rectángulo VI

## Ejemplo (continuación)

- Para invocar una función de más de un parámetro, simplemente se escribe el nombre de la función y a continuación dentro de un paréntesis los argumentos en los cuales se desea evaluar la función, en el mismo orden que se escribieron en la definición de la función, como se muestra a continuación cuando los argumentos son los valores  $l = 3$  y  $a = 4$ .

```
area_rectangulo(3, 4)
```

- Para el caso en el que se quiera observar el resultado de evaluar la función, se puede invocar la función `print` sin tabulación y se evalúa con el valor obtenido al aplicar la función `area_rectangulo(3, 4)`, como se observa a continuación

```
print(area_rectangulo(3, 4))
```

# Área de un rectángulo VII

## Ejemplo (continuación)

Para el llamado de la función utilizando la lectura de los argumentos por consola, se podría utilizar el siguiente código

```
def area_rectangulo(l, a):  
    area = l * a  
    return area  
  
largo = float(input("Largo del rectángulo: "))  
ancho = float(input("Ancho del rectángulo: "))  
print("El área del rectángulo es:", end = " ")  
print(area_rectangulo(largo, ancho))
```



# Agenda

- 1 Funciones de un parámetro de entrada
  - Cuadrado de un número
  - Área de un círculo
  - Conversión de tipos de datos numéricos (*typecasting*)
- 2 Funciones con más de un parámetro de entrada
  - Área de un rectángulo
  - Ley de Coulomb
- 3 Problemas



# Ley de Coulomb I

## Ejemplo

El módulo de la fuerza con que se atraen o se repelen dos cargas  $Q_1$  y  $Q_2$  (en culombios) que se encuentran separadas una distancia  $r$  (en metros), se puede calcular utilizando la ley de Coulomb que está dada por la expresión

$$F = \kappa \frac{Q_1 Q_2}{r^2}$$

donde  $\kappa$  se denomina constante eléctrica del medio y está dada por la Constante de Coulomb que en el vacío es igual a

$$\kappa = 9 \times 10^9 \frac{Nm^2}{C^2}.$$

modele matemáticamente, y luego escriba una función en Python que permita calcular el módulo de la fuerza con que se atraen o se repelen dos cargas  $Q_1$  y  $Q_2$  y que se encuentran separadas una distancia  $r$ .

# Ley de Coulomb II

## Ejemplo (continuación)

Para el cálculo de la ley de Coulomb se puede utilizar la siguiente especificación:

$Q_1$  := Valor de la primera carga

$Q_2$  := Valor de la segunda carga

$r$  := Distancia entre las cargas

$F$  := Fuerza con la que se atraen o se repelen las cargas  $Q_1$  y  $Q_2$   
que se encuentran a la distancia  $r$

entonces, el modelo matemático queda definido de la siguiente manera

$$\text{ley\_coulomb} : \mathbb{R} \times \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}$$

$$(Q_1, Q_2, r) \mapsto \kappa \frac{Q_1 Q_2}{r^2}$$

# Ley de Coulomb III

## Ejemplo (continuación)

Para el llamado de la función utilizando la lectura de los argumentos por consola, se podría utilizar el siguiente código

```
KAPPA: float = 9E+9
def ley_coulomb(Q1, Q2, r):
    modulo = KAPPA * Q1 * Q2 / r ** 2
    return modulo
carga1 = float(input("Carga 1: "))
carga2 = float(input("Carga 2: "))
distancia = float(input("Distancia entre cargas: "))
print("El módulo de la fuerza es:", end = " ")
print(ley_coulomb(carga1, carga2, distancia))
```





# Ley de Coulomb IV

## Ejemplo (continuación)

En este caso la Constante de Coulomb KAPPA se definió real y global para poderse utilizar en cualquier parte del programa.

Para las cargas  $Q_1 = 1.6\mu C$  y  $Q_2 = 0.4\mu C$ , que se encuentran a una distancia  $r$  de 30 cm, un llamado correcto de la función `ley_coulomb` puede ser

```
print(ley_coulomb(1.6e-6, 0.4e-6, 0.3))
```

y la impresión del valor obtenido es

```
El módulo de la fuerza es: 0.064
```

Obsérvese como aquí el orden en que se establezcan los argumentos es importante.

# Agenda

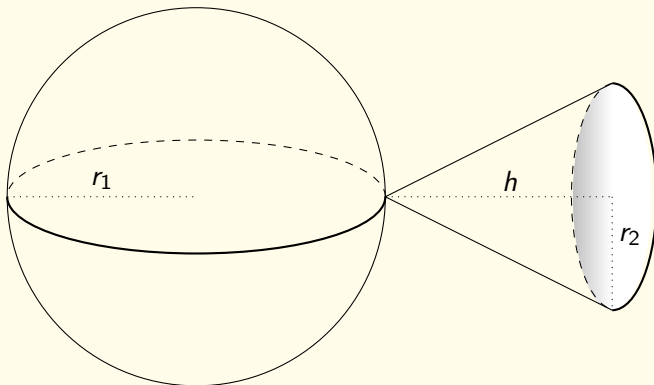
- 1 Funciones de un parámetro de entrada
  - Cuadrado de un número
  - Área de un círculo
  - Conversión de tipos de datos numéricos (*typecasting*)
- 2 Funciones con más de un parámetro de entrada
  - Área de un rectángulo
  - Ley de Coulomb
- 3 Problemas



# Volumen de un sólido I

## Problema

Para el sólido que se presenta a continuación,



# Volumen de un sólido II

## Problema (continuación)

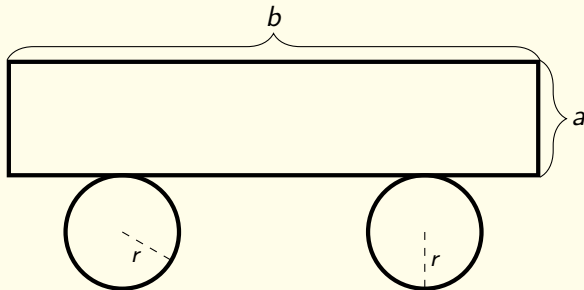
- 1 Establezca el modelo matemático (función matemática) que permita calcular el volumen del sólido anteriormente mostrado.
- 2 Escriba una función en Python que implemente la función anteriormente modelada, en la cual se invoque la constante matemática  $\pi$  del módulo `math`.
- 3 Para los valores  $r_1 = 3$ ,  $h = 9/2$  y  $r_2 = 4$ , calcule (a mano o con calculadora) el volumen del sólido y compárelo con el resultado obtenido a partir de la evaluación de la función anteriormente implementada. ¿Qué pasa si se invoca la función con los mismos valores, pero  $h$  se calcula como la expresión  $h = 9//2$ ?



# Área lateral de un vagón I

## Problema

Para el vagón que se presenta a continuación,



# Área lateral de un vagón II

## Problema (continuación)

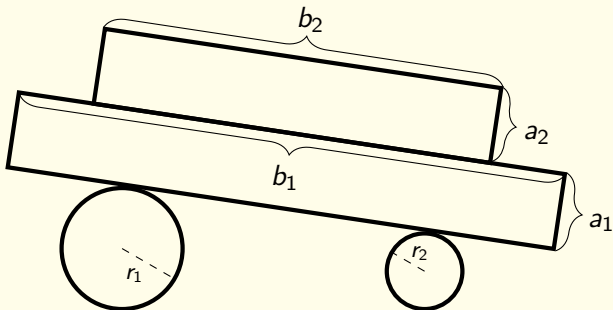
- 1 Establezca el modelo matemático (función matemática) que permita calcular el área lateral del vagón.
- 2 Escriba una función en Python que implemente la función anteriormente modelada, en la cual se invoque la constante matemática  $\pi$  del módulo `math`.



# Área lateral de un carro l

## Problema

Para el carro que se presenta a continuación,



# Área lateral de un carro II

## Problema (continuación)

- 1 Establezca el modelo matemático (función matemática) que permita calcular el área lateral del carro.
- 2 Escriba una función en Python que implemente la función matemática previamente modelada, en la cual se utilice la composición de las funciones de suma de números reales, *area\_circulo* y *area\_rectangulo* codificadas previamente.





# Problemas varios

## Problemas

- ① Diseñe una función que calcule la cantidad de carne de aves en kilos si se tienen  $N$  gallinas,  $M$  gallos y  $K$  pollitos cada uno pesando 6 kilos, 7 kilos y 1 kilo respectivamente.
- ② Mi mamá me manda a comprar  $P$  panes a \$ 300 cada uno,  $M$  bolsas de leche a \$ 3300 cada una y  $H$  huevos a \$ 350 cada uno. Hacer un programa que me diga las vueltas (o lo que quedo debiendo) cuando me da un billete de  $B$  pesos.
- ③ Si pido prestados  $P$  cantidad de pesos para pagarlos en dos meses, si el interés del préstamo es del 3%. ¿Cuánto se debe pagar al final del segundo mes si el interés es compuesto mensualmente?
- ④ El número de contagiados de Covid-19 en el país de NuncaLandia se duplica cada día. Hacer un programa que diga el número total de personas que se han contagiado cuando pasen  $D$  días a partir de hoy, si el número de contagiados actuales es  $C$ .