



# **FUNDAMENTOS DE PROGRAMACIÓN**

**Números, expresiones y programas  
simples**

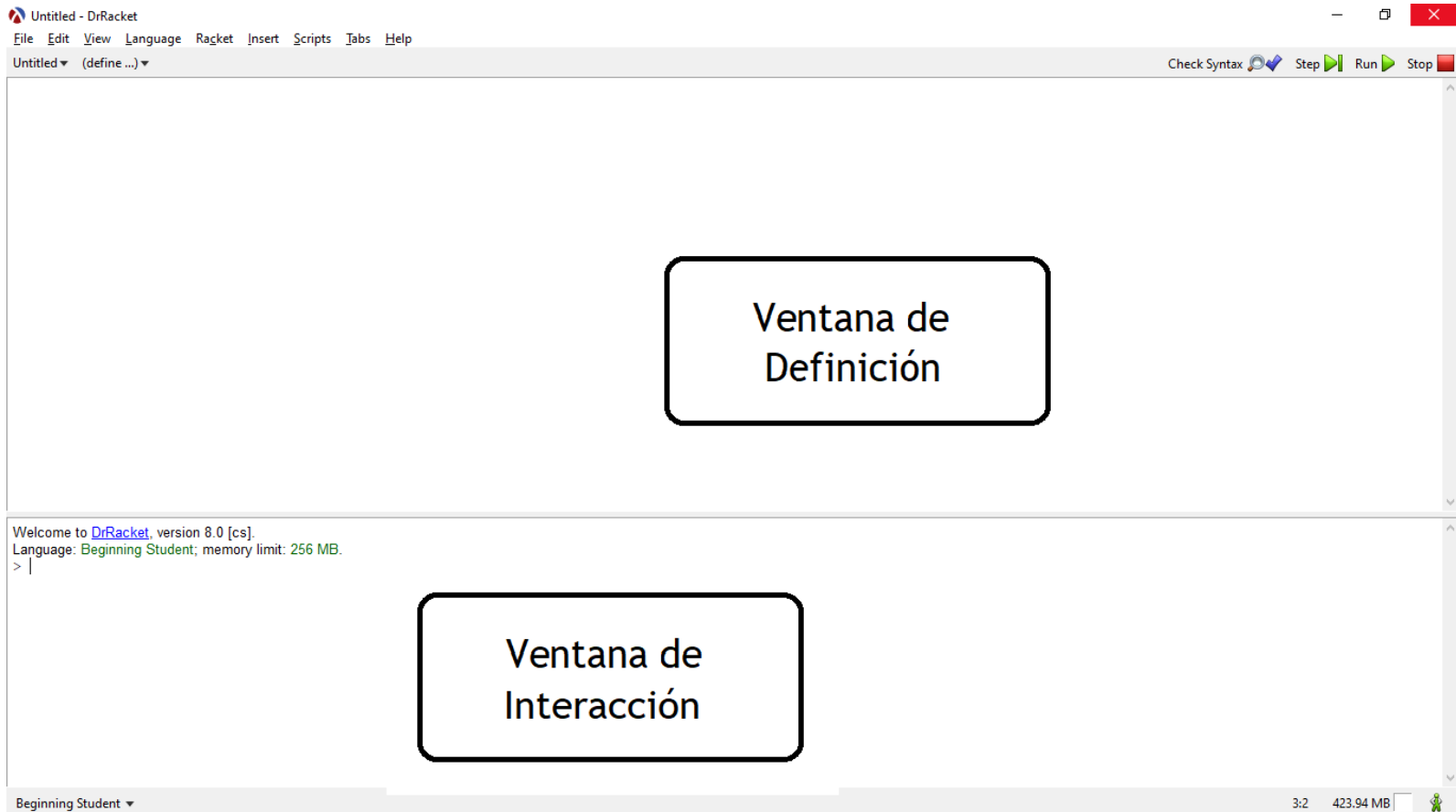
**Escuela de Ingeniería de Sistemas y Computación  
Facultad de Ingeniería  
Universidad del Valle**

# Recordemos

- Programa.
- Lenguaje de programación.
- El lenguaje que vamos a utilizar se llama Racket, es un lenguaje funcional, por tanto los programas son funciones.
- El entorno de desarrollo se llama DrRacket.



# y el entorno de desarrollo...



# Números

- En Racket se pueden representar varios tipos de números: enteros, racionales y reales. Por ejemplo:  $5$ ,  $-5$ ,  $2/3$ ,  $7/3$ , `#i1,4142135623731`.
- $5$ ,  $-5$ ,  $2/3$ ,  $7/3$ , `#i1,4142135623731` no tienen distinción de algún tipo, todos son simplemente números



# Operaciones con números

- Racket permite también sumar, restar multiplicar, dividir y en general realizar muchas operaciones con números.
- Las operaciones tienen notación prefija, esto es: el operador se escribe primero.
- Ejemplo:
  - $5 + 3$  se escribe `+ 5 3`
  - y  $3 * 5 + 15$ ??



# Expresiones

- Al igual que en la aritmética. para evitar diferentes interpretaciones usamos paréntesis:
  - $(3*5)+15$  se escribe  $+ (* 3 5) 15$
  - ¿ $((3*5)+15)$  es lo mismo que  $(3*5)+15$  ?
- **Expresión:** Conjunto de palabras que permiten expresar algo.  
Una expresión en racket es como una frase en español.
- Existen reglas para construir expresiones en racket:
  - Todas las expresiones en racket empiezan y terminan por paréntesis
  - Los operadores van primero (notación prefija)



# Ejemplos

- Expresiones que contienen operaciones numéricas:
  - $(+ 2 2)$  equivale a ???
  - $(* 3 1)$  equivale a ???
  - $(/ 10 0)$  equivale a ???
  - $(* 7 (+ 4 5))$  equivale a??
- Al igual que en álgebra se pueden escribir muchas operaciones dentro de otras usando paréntesis:
  - $((+ 2 2) (/ ((+ 3 5) (/ 30 10)) 2))$  equivale a 48
  - Los paréntesis internos deben resolverse primero!!!!



# Ejemplos

- Escribamos las expresiones en racket equivalentes a :
  - $((5+6)*4)/(17+9)$
  - $((+(4/8 - 2/8) 5)*((1-18)/(23 / 7)))$
- ¿Y cómo se escriben operaciones como potencias, raíces, logaritmos?
  - Estas operaciones se escriben siguiendo las reglas anteriormente vistas, pero no se puede escribir signos como  $3^4$  ó  $\log_{10} 2$  por tanto se usan palabras reservadas.
  - $3^4$  se escribe en racket (expt 3 4)





# Variables

- Las variables son valores que están ligados a un nombre.

$$\pi \rightarrow 3,14$$

$$e \rightarrow 2,7182$$

- Podemos definir variables usando la siguiente instrucción:

`(define <nombre> <valor>)`

## Ejemplos:

- `(define entrada 5)`
- `(define ganancia (entrada*6))`
- El valor puede ser resultado de una operación



# Ejercicio

- Escriba las siguientes instrucciones en la ventana de definiciones:
  - (define entrada 5)
  - (define ganancia (entrada\*6))
- Ahora de click en el botón ejecutar.
- Pregunta: ¿Qué pasa cuando escribe entrada, o ganancia en la ventana de interacción y luego da enter?
- escriba los siguiente en la ventana de interacción: (+ entrada ganancia), ¿cual es el resultado?



# Resumen

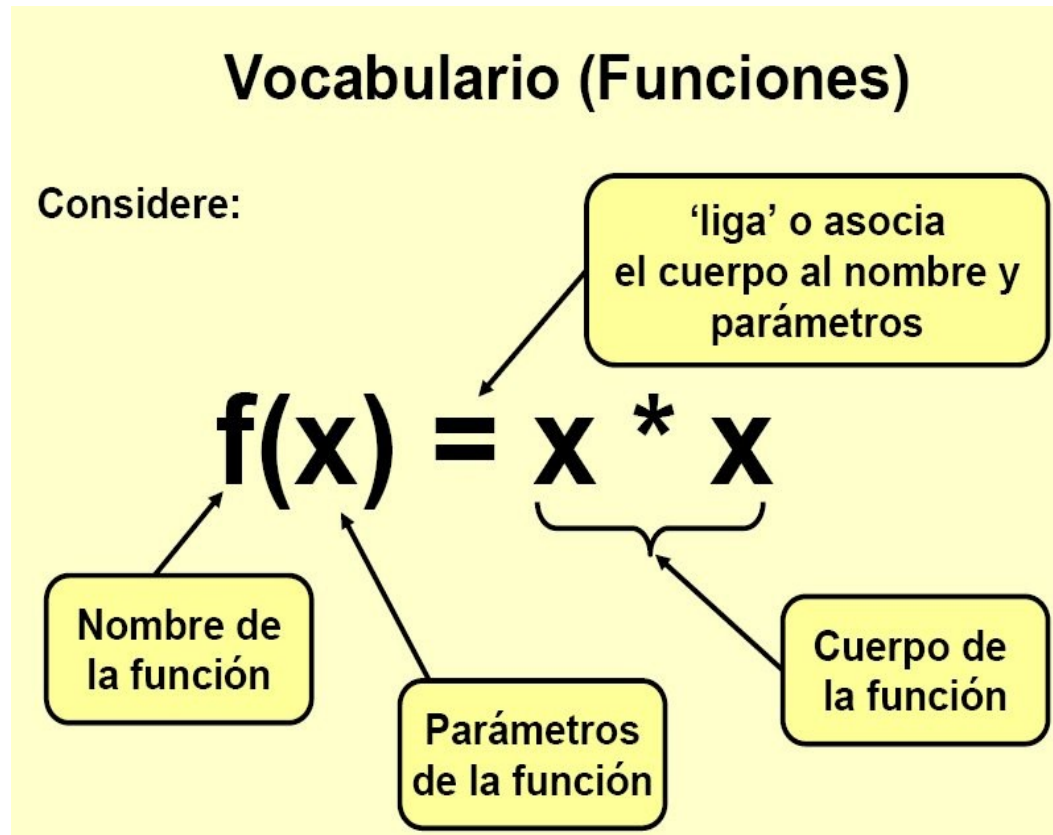
- Hasta ahora hemos visto 3 reglas para escribir expresiones en racket:
  - Todas las expresiones en racket empiezan y terminan por paréntesis
  - Los operadores van primero (notación prefija)
  - Podemos definir variables usando la siguiente instrucción:

`(define <nombre> <valor>)`



# Programas

- Recordemos que los programas en racket son funciones, por tanto para definir programas seguimos el vocabulario de funciones



# Programas

- Los programas igual que las funciones tienen nombre.
- Para relacionar un programa con su nombre es necesario definirlo.
- Un programa en racket se define de la siguiente forma:

(define (<encabezado>  
          <cuerpo>)

<encabezado> → <nombre> <parametros>

<cuerpo> → <expresión>

<nombre> → combinación de letras y numeros

<parametros> → conjunto de nombres de variables

<expresión> → Una expresión numérica bien escrita  
(siguiendo las reglas)



# Ejemplos

- Desarrolle un programa que calcule la hipotenusa de un triángulo rectángulo.
- Desarrolle un programa que calcule el perímetro de un cuadrado.
- Desarrolle un programa que determine si un número es par o no



# Errores

## ○ Errores de Sintaxis

- Comparable a los errores gramaticales en los lenguajes naturales.
- Racket verifica que todo el código este correctamente organizado de acuerdo a las reglas.

## ○ Errores en Tiempo de ejecución

- Dividir por cero, por ejemplo.
- Número incorrecto de parámetros cuando se intenta ejecutar la función. ( Error Semántico)

## ○ Errores Lógicos

- La “gramática” es correcta, pero la solución no es apropiada para el problema. Por ejemplo, se solicita una función que determine el área de un cuadrado y se realiza una función que determina la raíz cuadrada.

