

# Fundamentos de programación

## Introducción al curso

Facultad de Ingeniería. Universidad del Valle

Octubre de 2019

# Contenido

- 1 Presentación del curso
- 2 Introducción a la programación
- 3 Introducción a Dr Racket

# Contenido

- 1 Presentación del curso
- 2 Introducción a la programación
- 3 Introducción a Dr Racket

- Carlos Andres Delgado Saavedra
- Correo: carlos.andres.delgado@correounivalle.edu.co
- Ingeniero de Sistemas e Ingeniero Electrónico de la Universidad del Valle
- Magíster en Ingeniería: Énfasis en Ingeniería de Sistemas y Computación
- Docente de la sede desde el 2014
- Hoja de Vida: [Enlace CVLAC](#)

# El curso

- Código: 750080M
- Habilitable: No, Validable: Si
- Prerrequisitos: Ninguno
- El curso es prerrequisito de: (750084M) Matemáticas discretas I y (750095M) FLP
- Número de créditos: 4\*. 192 horas en total, 48 horas dictadas en 12 semanas y 144 de dedicación del estudiante en 18 semanas. Esto quiere decir que el estudiante debe dedicarse 8 horas semana de trabajo independiente.

\*Según el Decreto 1330 de julio 25 de 2019, un crédito equivale a 48 horas de trabajo incluidas las horas de contacto directo del estudiante con el docente y las horas de trabajo independiente

# Objetivos

Al final del curso usted podrá:

- Diseñar, escribir, depurar programas que resuelven problemas sencillos.
- Usar diferentes tipos de datos
- Usar diferentes tipos de datos compuestos
- Comprender el papel de la abstracción en el diseño de programas

# Objetivos

Al final del curso usted podrá:

- Diseñar y escribir programas usando la recursión generativa
- Diseñar y programar programas completamente funcionales
- Comprender las implicaciones de la asignación en el diseño de programas
- Comprender la técnica de refinamiento iterativo
- Comprender la importancia de la especificación para el trabajo en grupo

- Clases en video beam
- Realización de talleres prácticos
- Uso de una guía de diseño de programas
- Proyecto final que debe ser sustentado
- Las clases se graban y suben a Youtube



Puede ver las clases de periodos pasados

- 2019-1
- 2018-2

Puedes suscribirse a mi canal y darle clic a la campanita para recibir las mejores notificaciones de FDP:

<https://www.youtube.com/user/carlosUnivalle>

# Porcentajes

- Primer examen: 30 %
- Segundo examen: 35 %
- 6 talleres: 15 %. Los talleres serán trabajados en 1h y 30 durante la clase, los grupos de trabajo son asignados por el docente.
- Proyecto final: 20 %. Debe ser trabajado en grupos de mínimo 2 y máximo 3 integrantes. No se aceptan grupos individuales ni de más de 3 integrantes. Estos son de libre conformación.

# Contenido

- 1 Introducción al curso
- 2 Procesamiento de datos simples
  - a) Elementos de un programa
  - b) Expresiones condicionales
  - c) Datos simples
  - d) Datos compuestos I: Estructuras

A esto le dedicaremos 3 semanas.

## 3 Procesamiento de datos arbitrariamente grandes

- a) Datos compuestos II: Listas
- b) Definiciones recursivas: Lo números  
**Hasta aquí va el primer parcial.**
- c) Datos compuestos III: Árboles
- d) Definiciones locales

A esto le dedicaremos 3 semanas.

## 4 Abstracción en el diseño

- a) El problema de las múltiples funciones
- b) Desarrollo de funciones abstractas
- c) Funciones como ciudadanos de primera clase

A esto le dedicaremos 2 semanas.

## 5 Recursión Generativa

- a) Diferencias entre recursión estructural y recursión generativa.
- b) Algoritmos de ordenamiento.
- c) Diseño de funciones generativas.

A esto le dedicaremos 1 semana.

## 6 Acumulación de conocimiento

- a) La pérdida de conocimiento.
- b) Diseñando funciones que acumulan.

A esto le dedicaremos 1 semana.

## 7 Cambiando el estado de las variables

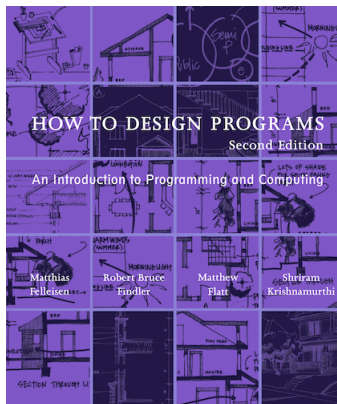
- a) La necesidad del manejo de memoria en las funciones
- b) Asignación de valores a variables.
- c) Diseño de funciones con memoria
- d) Vectores

A esto le dedicaremos 1 semana.



# Bibliografía

El curso trabaja sobre el libro How to Desing programs.



El libro está disponible gratuitamente aquí: <https://htdp.org>

# Información importante

El campus virtual es herramienta de comunicación del docente.  
Registren su correo institucional en el momento que les sea asignado para evitar los anuncios terminen en la bandeja de SPAM.



06-750080M-50-201910041

FUNDAMENTOS DE  
PROGRAMACIÓN-50

# Ten presente

- El lenguaje es un poco difícil de dominar al principio, pero se adquiere trabajando
- Es un entorno educativo para aprender a programar
- ¡Animate! Esta es tu entrada al mundo de la programación

# Ten presente

Es un curso que requiere dedicación y esfuerzo, así lo muestran las estadísticas de los años anteriores:

Año	Matriculados	Aprobaron	Promedio
2016-1	46	21	2.503
2017-1	31	20	3.0189
2018-1	29	18	2.987
2018-2	32*(18)	15	3.43
2019-1	27	13	2.81

\* Este semestre debido a la movilización estudiantil se abrieron cancelaciones al final del semestre.

# Contenido

- 1 Presentación del curso
- 2 Introducción a la programación
- 3 Introducción a Dr Racket

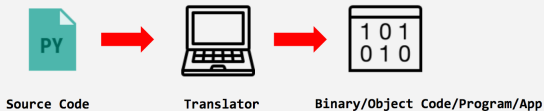
# Introducción a la programación

## Programación

- 1 Conjunto de signos y reglas que permiten la comunicación de un humano con un computador por medio de programas
- 2 El proceso que sigue un computador para ejecutar un programa es:
  - Interpretación
  - Traducción a otro lenguaje
- 3 Es un proceso de llevar conceptos del "mundo de las ideas." al "mundo real"

# Introducción a la programación

## Concepto de programación



# Introducción a la programación

## Lenguajes de programación

- 1 **Lenguaje de máquina:** Conjunto de instrucciones que una computadora puede interpretar directamente. Ejemplo:  
0001010101
- 2 **Lenguaje ensamblador:** Es muy similar a la máquina, sin embargo tiene instrucciones que pueden ser entendidas por seres humanos **SUM A B**
- 3 **Lenguaje de alto nivel:** Permiten diseñar y construir programas de tal manera se adecue a la capacidad cognitiva humana, en lugar de la capacidad ejecutora de la máquina



# Introducción a la programación

## Ejercicio

Programar no es sólo es escribir un código, ¡es volver realidad tus ideas!.

- 1 Abre esta página <https://www.lightbot.lu/>
- 2 Asume el reto, resuelve los problemas :)

# Introducción a la programación

## Pregunta

¿Que creen es la programación?

<https://www.youtube.com/watch?v=yCn0atPy3e4>

# Introducción a la programación

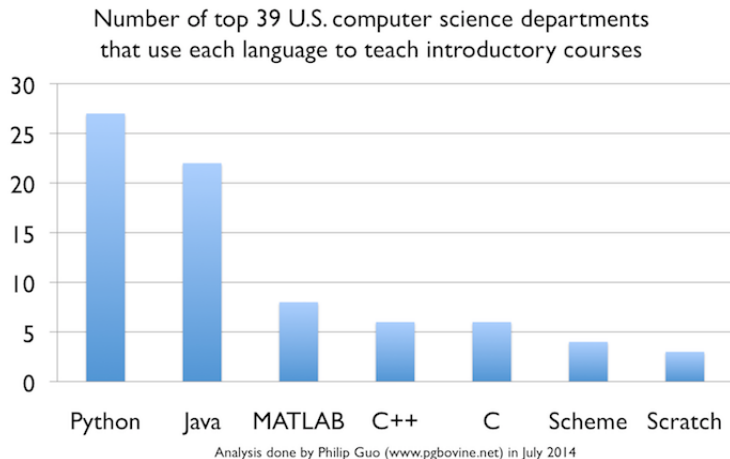
## Definición

Programar es mucho más que calcular...

- 1 Leer y entender problemas en muchos dominios
- 2 Organizar el conocimiento para aprender lo que no se sabe aún
- 3 Entender lo que realmente se desea y explorar ejemplos que fortalezcan la comprensión del problema **especificación**
- 4 Extraer los conceptos importantes y relevantes **abstracción**

# Introducción a la programación

## Lenguaje de trabajo



# Contenido

- 1 Presentación del curso
- 2 Introducción a la programación
- 3 Introducción a Dr Racket**

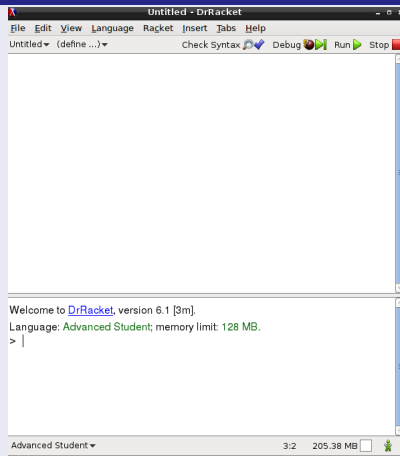
# Introducción a Dr Racket

## Interfaz de desarrollo

- El entorno de trabajo se llama Dr Racket
- Puede ser descargado desde <https://racket-lang.org>
- Está disponible para Windows, Linux y Mac
- El programa que usaremos lo encuentras en el menú como DrRacket

# Introducción a Dr Racket

## Interfaz de desarrollo



# Introducción a Dr Racket

$(+ \ 7 \ 4)$

- Las expresiones en Dr Racket son en notación prefija, por ejemplo  $(+ \ 5 \ 2)$  es equivalente a  $5+2$  en notación infija.
- Se recomienda utilizar paréntesis para evitar problemas en la interpretación de resultados por ejemplo  $(+ \ (- \ 3 \ 2) \ 4)$  es equivalente en notación infija a  $((3 - 2) + 4)$
- En seleccionar lenguaje para el curso de Fundamentos de Lenguaje de programación debe seleccionar Estudiante Principante. (¡Luego subimos el voltaje!)

$4 \times 5^2 - 8$   
 $(+ \ 5 \ (- \ 4 \ 2))$   
 $(+ \ 5 \ (- \ 4 \ 2))$



# Introducción a Dr Racket

Notación prefija:

```
( 1 - ( sqrt( 2 + sin( 3 ) ) * 4 ) )
```

```
( - 1 ( * sqrt( + 2 sin( 3 ) ) 4 ) )
```

$$1 - \sqrt{2 + \sin(3)} \times 4$$

( ? ← operación

$( - 1 ( * ( \sqrt{ + 2 ( \sin 3 ) } ) 4 ) )$

$$3^2 \pm \left(\frac{8}{4}\right)^7$$

$$\sqrt[3]{8} = 8^{\frac{1}{3}}$$

$$\left[ \sqrt[3]{1 \pm \frac{8}{\sqrt{3+2^2}}} \right]$$

$$(/ (+ (expt 3 2) (expt (/ 8 4) 7)))$$

$$(expt (+ 1 (/ 8 (expt (+ 3 (* 2 2) (/ 1 2)))))$$

$$) \quad (/ 1 3))$$

# Introducción a Dr Racket

- Se utiliza la palabra reservada **define** para la definición de variables por ejemplo

```
(define perrito 5)
```

```
(define gatito (* 2 perrito))
```

- Las variables son útiles para almacenar información

# Introducción a Dr Racket

```
(+ 2 2)      ;; Resultado 4
(* 3 3)      ;; Resultado 9
(- 4 2)      ;; Resultado 2
(/ 6 2)      ;; Resultado 3
( sqr 3)     ;; Resultado 9
( expt 2 3)   ;; Resultado 8
( sin 0)     ;; Resultado 0
( cos pi )   ;; Resultado # i-1.0
(+ (* 5 5) (+ (* 3 (/ 12 4) ) 4) ) ;; ??
(+ 1 2 3 4 5 6 7 8 9 0) ;; ??
(* 1 2 3 4 5 6 7 8 9 0) ;; ??
( string-append "hello" "world" ) ;; ??
(+ ( string-length "hello world" ) 20) ;; ??
( > 10 9)    ;; ??
( = 10 9)    ;; ??
( <= 10 9)   ;; ??
```

# Introducción a Dr Racket

## Ejercicios

Tu primer reto: Calcule la siguientes operaciones en el Dr Racket:

- $2 * 2 + 3 * 5 + (\frac{1}{4})^2$  Resultado: 19.0625
- $2 * (1 + 3^2 + \frac{4}{4}) + 3 * (5 - 3) + \frac{12}{4} - 3 + 4 * 5^3$  Resultado: 528
- $2 * (1 + \frac{7}{4}) + (1 - 2) * (5 - 3) + \frac{1+34}{4} - \frac{8}{9} + 4 * 5^3$  Resultado: 511.361

Para potenciación  $2^3$  se escribe (expt 2 3)

$$\sqrt[x]{y} = (\text{exp+ } y \text{ (1 } x))$$

$$\sqrt[3]{8} = (\text{root of } 8 \text{ (13)})$$

$$Q) \frac{\sqrt{8^2 + \frac{7}{4}} + \sqrt[3]{8^3 + \frac{1}{6^3}}}{(1 + \sin(\sqrt{3^2 + \sqrt{3}}))^8}$$

$$b) \sqrt[3]{\sqrt[4]{6 + \frac{8}{\sqrt{5^3}} + 7\sqrt{6} - \frac{3}{\sqrt{7^2}}}}$$

# Introducción a Dr Racket

## Ejercicios

$$2 * 2 + 3 * 5 + \left(\frac{1}{4}\right)^2$$

```
(+ (* 2 2) (* 3 5) (expt (/ 1 4) 2))
```

Resultado: 19.0625

# Introducción a Dr Racket

## Ejercicios

$$2 * (1 + 3^2 + \frac{4}{4}) + 3 * (5 - 3) + \frac{12}{4} - 3 + 4 * 5^3$$

```
(+ (* 2 (+ 1 (expt 3 2) (/ 4 4))) (* 3 (- 5 3)) (/ 12 4) -3  
  (* 4 (expt 5 3)))
```

Resultado: 528



# Introducción a Dr Racket

## Ejercicios

$$2 * (1 + \frac{7}{4}) + (1 - 2) * (5 - 3) + \frac{1+34}{4} - \frac{8}{9} + 4 * 5^3$$

```
(+ (* 2 (+ 1 (/ 7 4))) (* (- 1 2) (- 5 3)) (/ (+ 1 34) 4) (/
  -8 9) (* 4 (expt 5 3)))
```

Resultado: 511.361

# Introducción a Dr Racket

## Ejercicios en clase

Ahora intenta los siguientes ejercicios:

■  $\frac{(1+4/7+8)^2}{5-3+\frac{8^3}{7+3-8^3}}$  Resultado: 93.4742823959

■  $\frac{(2-7)^3+\frac{8^2+2*(6-3)}{(7-3)*(9-4)}}{8^2+6*4+7}$  Resultado: -1.27894736842

■  $\frac{(4-5)^2}{7^2+6*9+\frac{8+6}{5+6*7-2^2}}$  Resultado: 0.00967814539

$$a) \frac{7^2 + \sqrt[5]{8^3 + 9^6}}{2 + \sqrt{7}}$$

(/ (+ (sqr 7) (expt (+ (expt 8 3) (expt 9 6))  
(/ 1 5))))

(+ 2 (sqrt 7))

)

$$b) \sqrt{8^2 + \sqrt{\frac{9}{4}}} - \frac{\sqrt{\frac{1}{3}}}{2}$$

(sqrt (+ (expt 8 2) (sqrt (/ 9 4))  
(- (/ (sqrt (/ 1 3)) 2))))

# Introducción a Dr Racket

$$(+ (\text{expt } 2 \ 3) \ 8 \ 7) \quad 2^3 + 8 + 7$$

## Ejercicios en clase

Ahora, evalúa las siguientes expresiones en DrRacket y transformarlas a notación infija:

- $(- (+ (+ 10 \ 29) (- 32 \ 3)) (+ 34 (- 2 \ 1)))$
- $(+ (\text{expt } 2 (/ 2 \ 4)) (\text{expt } 3 \ 2) (/ (+ 1 \ 44) (* 2 (+ 4 \ 3))))$
- $(\text{expt } (+ 3 (* 4 \ 5) (/ 7 \ 6)) (/ 4 (* 4 (\text{expt } 2 \ 3))))$

$$(- (+ (+ 10 29) (- 32 3)) (+ 34 (- 2 1)))$$

$$((10 + 29) + (32 - 3)) - (34 + (2 - 1))$$

input

$(\text{expt } (+ 3 (* 4 5) (/ 7 6)) (/ 4 (* 4 (\text{expt } 2 3))))$   
 base                      exponento

$$\left( 3 + 4 \times 5 + \frac{7}{6} \right)^{\frac{4}{4 \times 2^3}}$$

$$(\text{expt } a b)$$

$a^b$

# A tener en cuenta

- Aplicar la metodología propuesta en el curso
- Explorar ejemplos
- Hacer ejercicios por fuera de clase

# ¿Preguntas?

