

# Laboratorio de Programación (300CIP007)

Luis Garreta

[luis.garreta@javerianacali.edu.co](mailto:luis.garreta@javerianacali.edu.co)

Ingeniería de Sistemas y Computación  
Pontificia Universidad Javeriana – Cali

25 de enero de 2017

# Información Básica

- Créditos: 2
- Horas de Clase: 3 / semana
- Horas de trabajo independiente: 4 / semana
- Prerequisitos: Introducción a la Programación (300CIP001)

# Objetivos Instruccionales

Al finalizar el curso los participantes podrán:

- Diseñar y describir apropiadamente la solución de problemas básicos de la computación.
- Utilizar diferentes técnicas de programación para solucionar problemas.
- Reconocer la necesidad y utilidad de documentar apropiadamente el código fuente.
- Usar herramientas útiles para desarrollo de proyectos.
- Identificar las principales características prácticas de los lenguajes de programación compilados.

# Competencias que se Desarrollan

- Habilidades específicas: Uso del lenguaje C, compilador Gcc y herramienta Make.
- Conceptos Fundamentales de Computación: Diseño de algoritmos, programación modular y manejo de interprete de comandos y documentación de código fuente.
- Competencias Generales:
  - Desarrollo modular de programas.
  - Análisis y diseño de soluciones con pseudocódigo.
  - Lectura y escritura de archivos.
  - Trabajo en grupo, escritura de reportes.

# Metodología

- El curso es presencial y con alta dosis de participación y trabajo en clase.
- Para algunas sesiones de clase se asignarán investigaciones, ejercicios y lecturas.
- Durante la sesión se expondrán los conceptos acompañados de ejemplos
- Se fomentará la participación de los estudiantes.
- Se realizarán talleres en el que se ponga en práctica algunos conceptos de computación y exámenes de seguimiento de avance.

# Contenido

- **Introducción** : Pseudocódigo, lenguajes compilados, lenguaje C, compilador gcc, entradas y salidas.
- Operadores, asignación, lógicas, prescendencia.
- **Lenguaje C**: tipos de errores, manejo de cadenas, operadores, prescendencia, incrementos y decrementos, prioridad y asociatividad.
- División entre cabeceras e implementación, directivas.
- **Sentencias**: if, if anidadas, if else, ciclos while, switch, do-while, for, break, continue.
- **Codificación**: Herramienta make, Documentación de código.
- **Manejo de memoria**: Punteros, Arreglos, Cadenas, Matrices
- Memoria dinámica, typedef, structs, enums.
- **Archivos**: Lectura/escritura de archivos de texto y binarios.
- **Algoritmos de ordenamiento**: Burbuja, Selection-Insertion, Quick sort, Merge sort.
- **Estructuras de Datos Lineales**: listas sencillas, listas doblemente enlazadas, pilas y colas.
- **Estructuras de Datos Tipo Red**: Árboles binarios, Grafos.

# Evaluación

- Parcial 1: 30 %
- Parcial 2: 30 %
- Proyecto: 20 %
- Quizes, talleres, otros: 20 %

# Bibliografía

- ① Programación en C.  
[https://es.wikibooks.org/wiki/Programación\\_en\\_C](https://es.wikibooks.org/wiki/Programación_en_C)
- ② Herbert Schildt, Programación en lenguaje C. McGraw-Hill, 1988.
- ③ Harvey M. Deitel, Paul J. Deitel, Como programar en C/C++. 6th Edicion. Prentice Hall Hispanoamericana, 2010.
- ④ Brian W. Kernighan, Dennis M. Ritchie, El lenguaje de programación C. Prentice Hall Hispanoamericana, 1985.
- ⑤ James F. Peters, Hamed M. Sallam, Compleat C. Prentice Hall, 1986.
- ⑥ Aaron M. Tenenbaum, Langsam Yedidiah, Moshe J. Augenstein, Data structures using C. Prentice Hall, 1990.
- ⑦ Visualising data structures and algorithms through animation  
<http://visualgo.net/>



# Material de este semestre

- Editor  
online:[http://www.tutorialspoint.com/compile\\_c\\_online.php](http://www.tutorialspoint.com/compile_c_online.php)
- Curso en el moodle
- ...Otros para más adelante.