



PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE  
ESCUELA DE INGENIERÍA  
DEPARTAMENTO DE CIENCIA DE LA COMPUTACIÓN

## IIC3800 — Tópicos Avanzados en Ciencia de la Computación

### Programa de Curso

Semestre 2/2018 – Profesor: Cristian Ruz – [cruz@ing.puc.cl](mailto:cruz@ing.puc.cl)

Horario: M-J:3, Sala Javier Pinto.

Este curso estudia el proceso de compilación, y sus fases de análisis léxico, sintáctico y semántico. Además, estudia el manejo de memoria, la optimización de código y la generación de código objeto.

## 1. Objetivos

Al finalizar el curso el alumno será capaz de:

- Identificar las fases del proceso de compilación, y las formas en que ellas se comunican.
- Adaptar gramáticas independientes del contexto a gramáticas que puedan usarse en lenguajes de programación.
- Diseñar y construir analizadores léxicos, sintácticos y semánticos para lenguajes independientes del contexto, y los traductores de estos lenguajes a lenguajes intermedio, objeto o de alto nivel, incluyendo técnicas para encontrar y corregir errores.
- Aplicar técnicas de optimización, para código intermedio y código objeto.
- Explicar el trabajo realizado y las técnicas usadas por el manejador de memoria en un compilador.
- Explicar algoritmos de recolección de basura.

## 2. Contenido

El curso abordará los siguientes temas:

- |  |                                      |
|--|--------------------------------------|
| 0. El proceso de compilación                   | 3.2) Tablas de símbolos              |
| 0.1) Etapas de compilación                     | 3.3) Chequeo de tipos                |
| 0.2) Estructura de un compilador               | 4. Representación Intermedia         |
| 1. Análisis Léxico                             | 4.1) Entornos de ejecución           |
| 1.1) Lexemas y tokens                          | 4.2) Árboles de activación           |
| 1.2) Expresiones regulares y autómatas finitos | 4.3) Objetos, herencia e interfaces  |
| 1.3) LEX (flex)                                | 4.4) Generación de código intermedio |
| 2. Análisis Sintáctico                         | 4.5) Semántica operacional           |
| 2.1) Parsing y gramáticas libres de contexto   | 5. Optimización de código            |
| 2.2) AST: <i>Abstract Syntax Tree</i>          | 5.1) Optimización local              |
| 2.3) Top-Down parsing                          | 5.2) Optimización global             |
| 2.4) Bottom-Up parsing                         | 6. Administración de memoria         |
| 2.5) YACC (bison)                              | 6.1) Asignación de registros         |
| 3. Análisis Semántico                          | 6.2) Recolección de basura           |
| 3.1) Scope                                     |                                      |

### **3. Metodología**

Una clase expositiva semanal separada en 2 módulos de 80 min. de duración. Un proyecto semestral a realizar en grupos de dos.

### **4. Bibliografía**

El curso está preparado de manera que el material de clases y las referencias que se hagan durante ellas sean suficiente para comprender los contenidos. Sin embargo, aquellos que desean profundizar más en los contenidos están invitados a hacerlo. El siguiente libro es una referencia general para los temas de compilación:

1. Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman. *Compilers: Principles, Techniques, and Tools*, 2nd Edition. Addison Wesley. Sept 2006. ISBN 978-0321486813. <http://dragonbook.stanford.edu/>
2. Cursos online de Compiladores de Stanford. Alex Aiken.



PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE  
ESCUELA DE INGENIERÍA  
DEPARTAMENTO DE CIENCIA DE LA COMPUTACIÓN

## IIC3800 — Construcción de Compiladores

### Aspectos Administrativos

Semestre 2/2018 – Profesor: Cristian Ruz – [cruz@ing.puc.cl](mailto:cruz@ing.puc.cl)

Horario: M-J:3, Sala Javier Pinto

## 1. Horario, página web y atención a alumnos

Cátedra: martes y jueves, módulo 3. La página web del curso estará ubicada en la plataforma IngCursos (SIDING).

Atención de alumnos en Oficina P17, DCC. De preferencia enviar email previamente. Para cualquier tema relacionado con el curso, favor de iniciar el subject con [iic3800] para facilitar el *parsing*.

## 2. Evaluación

La evaluación del curso incluye 4 controles en horario de clase, y un proyecto desarrollado en 4 etapas.

Los controles comprenderán cada etapa del curso (uno por mes).

$N_C$  será el promedio aritmético de los 3 mejores controles:  $N_C = (C_1 + C_2 + C_3 + C_4 + \min(C_1, C_2, C_3, C_4))/3$ .

El proyecto consistirá en el desarrollo de un compilador, en 4 etapas de desarrollo.  $N_P$  será el promedio ponderado de las 4 etapas considerando:

|      |                                     |                           |
|------|-------------------------------------|---------------------------|
| 15 % | Etapa 1: Selección de lenguaje      | Martes 28-Agosto-2018     |
| 15 % | Etapa 2: Análisis Léxico/Sintáctico | Jueves 27-Septiembre-2018 |
| 35 % | Etapa 3: Análisis Semántico         | Jueves 25-October-2018    |
| 35 % | Etapa 4: Generación de código       | Jueves 22-Noviembre-2018  |

La nota final,  $N_F$ , se calcula como:

$$N_F = \begin{cases} 20 \% \cdot N_C + 80 \% \cdot N_P & \text{si } N_P \geq 4 \\ N_P & \text{en otro caso} \end{cases}$$

Los controles se efectuarán en horario de clases (viernes, 8:30) en las fechas:

|       |                         |
|-------|-------------------------|
| $C_1$ | Jueves 30 de Agosto     |
| $C_2$ | Martes 25 de Septiembre |
| $C_3$ | Martes 30 de Octubre    |
| $C_4$ | Martes 20 de Noviembre  |

## 3. Política de Integridad Académica

Los alumnos de la Escuela de Ingeniería de la Pontificia Universidad Católica de Chile deben mantener un comportamiento acorde a la Declaración de Principios de la Universidad. En particular, se espera que mantengan altos estándares de honestidad académica. Cualquier acto deshonesto o fraude académico está prohibido; los alumnos que incurran en este tipo de acciones se exponen a un Procedimiento Sumario. Es responsabilidad de cada alumno conocer y respetar el documento sobre Integridad Académica publicado por la Dirección de Docencia de la Escuela de Ingeniería.

Específicamente, para los cursos del Departamento de Ciencia de la Computación, rige obligatoriamente la siguiente política de integridad académica. Todo trabajo presentado por un alumno para los efectos de la evaluación de un curso debe ser hecho individualmente por el alumno, sin apoyo en material de terceros. Por trabajo se entiende en general las interrogaciones escritas, las tareas de programación u otras, los trabajos de laboratorio, los proyectos, el examen, entre otros. Si un alumno copia un trabajo, obtendrá nota final 1.1 en el curso y se solicitará a la Dirección de Docencia de la Escuela de Ingeniería que no le permita retirar el curso de la carga académica semestral. Por copia se entiende incluir en el trabajo presentado como propio partes hechas por otra persona.

Obviamente, está permitido usar material disponible públicamente, por ejemplo, libros o contenidos tomados de Internet, siempre y cuando se incluya la referencia correspondiente.

Lo anterior se entiende como complemento al Reglamento del Alumno de la Pontificia Universidad Católica de Chile. Por ello, es posible pedir a la Universidad la aplicación de sanciones adicionales especificadas en dicho reglamento.