

PRONTUARIO

TÍTULO:	Estructuras de compiladores
CODIFICACIÓN:	CCO 320
PRERREQUISITO:	Análisis comparativo de lenguajes de programación (CCO 215)
CRÉDITOS:	3 créditos 24 horas de reunión presencial 21 horas en línea 1 semestre

DESCRIPCIÓN

Se estudian las etapas básicas del proceso de compilación: análisis léxico, análisis sintáctico, codificación y optimización, y las estructuras de datos y algoritmos necesarios para implantar estas etapas. Se requerirá un proyecto de programación y un proyecto de investigación. Este curso es una electiva para estudiantes del Programa de Ciencias de Cómputos. Curso dictado parcialmente en línea.

JUSTIFICACIÓN

Un programador utiliza traductores constantemente. El compilador es uno de los componentes de más valor en el sistema de información, pues sirve de puente entre el programador y la computadora. El conocimiento de la organización y mecanismos utilizados por el compilador ayudan al programador a lograr una mejor y más eficiente aplicación de los distintos lenguajes de programación, a hacer una implantación mejor de los algoritmos que deba desarrollar, y a seleccionar correctamente el compilador apropiado para la aplicación a ser desarrollada.

COMPETENCIAS

El curso desarrolla en el o la estudiante las siguientes competencias:

- Cuestionamiento crítico
- Emprendimiento e innovación
- Investigación y exploración
- Comunicación

- Sentido ético y justicia social

OBJETIVOS

Al finalizar el curso el o la estudiante será capaz de:

1. Identificar los conceptos básicos de la teoría de los traductores.
2. Explicar la diferencia entre compiladores, intérpretes y ensambladores.
3. Describir la estructura de un compilador.
4. Explicar el análisis léxico y los métodos usados para su implantación.
5. Explicar el análisis de sintaxis y los métodos usados para su implantación.
6. Explicar el generador de código y los métodos usados para su implantación.
7. Analizar y comparar diferentes tipos de compiladores.
8. Discutir e implantar algunos componentes de un compilador.
9. Desarrollar destrezas de autoaprendizaje.
10. Expresar sus ideas de forma clara y coherente en forma oral y escrita, especialmente en la producción de documentación de sistemas.
11. Utilizar múltiples fuentes de información integradamente en el desarrollo de una investigación sobre un compilador disponible comercialmente.
12. Integrar la teoría y la práctica a través de proyectos de investigación y de programación.

CONTENIDO

- I. Introducción a la traducción de lenguajes de alto nivel.
 - A. Objetivos de la traducción
 - B. Tipos de traductores
 1. Intérpretes
 2. Ensambladores
 3. Compiladores
 4. "High Level Translators"

- C. Organización general de un compilador
- II. Análisis léxico
 - A. Conjunto de "tokens"
 - B. Analizador léxico o "scanner"
 - C. Tablas de símbolos
 - D. Diseño
 - E. Máquinas de estado finito
- III. Gramática de contexto libre
 - A. Definición
 - B. Producciones
 - C. Árboles
- IV. Análisis de Sintaxis
 - A. Gramática con atributos
 - B. "Top-Down Parsing"
 - C. "Bottom-Up Parsing"
 - D. Diseño
 - E. "Syntax-directed translation"
- V. Generador de Código
 - A. Tabla de símbolos
 - B. Distribución de memoria
 - C. Manejo de registros
 - D. Diseño
 - E. Optimización de código

METODOLOGÍA

Se recomiendan las siguientes estrategias de la metodología de aprendizaje activo:

Conferencia

Análisis crítico y discusión de lecturas del libro de texto y de otras fuentes

Proyecto de programación (diseño e implantación de una parte de un compilador)

Proyecto de investigación

Trabajo en equipo

Presentación oral y con recursos multimedia (presentación electrónica)

Uso de sistema de educación a distancia para acceder al componente en línea del curso

Uso de recursos disponibles en la Internet

RECURSOS

Acceso a Internet y al World Wide Web.

EVALUACIÓN

Exámenes parciales	40%
Programas	35%
Proyecto final y su presentación	<u>25%</u>
Total	100%

AVALÚO DEL APRENDIZAJE

Se aplica la rúbrica de avalúo institucional a la actividad central del curso.

BIBLIOGRAFÍA

TEXTO

Aho, Alfred V., et al. Compilers Principles, Techniques, and Tools 2^{da} edición. Reading: Addison Wesley Longman, 2007.

REFERENCIAS

Beck, Leland. Systems Software: And Introduction to Systems Programming 3^{era} edición. Reading: Addison Wesley Longman, 1997.

Binstock, Andrew and Rex, John. Practical Algorithms for Programmers. Reading: Addison Wesley Longman, 1995.

Cooper, Keith and Torczon, Linda. Engineering a Compiler. Morgan Kauffman, 2003.

Deitel, Harvey and Deitel, Paul. C How to Program 5^{ta} edición. Upper Saddle River: Prentice Hall, 2006.

Fischer, Charles N. and LeBlanc, Richard J. Crafting a Compiler with C. Reading: Addison Wesley Longman, 1991.

DIRECCIONES ELECTRÓNICAS

<http://www.cs.rice.edu/~keith/512/Lectures/>

<http://www.ucse.edu.ar/fma/compiladores/>
<http://webdiis.unizar.es/~ezpeleta/COMPI/compiladoresI.htm>
<http://www.compilers.net/>
<http://www.personal.kent.edu/~rmuhamma/Compilers/compiler.html>
<http://www.idiom.com/free-compilers/>

Puede encontrar más recursos de información relacionados a los temas del curso en la página de la biblioteca <http://biblioteca.sagrado.edu/>

Las bases de datos electrónicas a las cuales la Biblioteca Madre María Teresa Guevara está suscrita directamente y a través del Consorcio COBIMET, incluyen libros, documentos, artículos de revistas y periódicos y otros recursos de información relacionados con los temas del curso. Al utilizarlas siga los siguientes pasos:

Para acceder desde cualquier lugar en la Universidad

- escriba la dirección <http://biblioteca.sagrado.edu/>,
- seleccione **Biblioteca Virtual** y aparecerá la página en donde podrá acceder a las bases de datos, por disciplina o en orden alfabético.

Para acceder fuera de la Universidad

- escriba la dirección <http://biblioteca.sagrado.edu/>,
- seleccione **Biblioteca Virtual** y aparecerá la página en donde podrá acceder a las bases de datos, por disciplina o en orden alfabético.
- escriba el nombre del usuario y la contraseña

El nombre de usuario y la contraseña, los solicita personalmente en la Biblioteca.

ACOMODO RAZONABLE

Para obtener información detallada del proceso y la documentación requerida, debe visitar la oficina correspondiente. Para garantizar igualdad de condiciones, en cumplimiento de la Ley ADA (1990) y el Acta de Rehabilitación (1973), según enmendada, todo estudiante que necesite servicios de acomodo razonable o asistencia especial deberá completar el proceso establecido por la Vicepresidencia de Asuntos Académicos.

INTEGRIDAD ACADÉMICA

Esta política aplica a todo estudiante matriculado en la Universidad del Sagrado Corazón para tomar cursos con o sin crédito académico. Una falta de integridad académica es todo acto u omisión que no demuestre la honestidad, transparencia y responsabilidad que debe caracterizar toda actividad académica. Todo estudiante que falte a la política de honradez, fraude y plagio se expone a las siguientes sanciones: recibirá nota de cero en la evaluación y/o repetición del trabajo en el seminario, nota de F(*) en el seminario: suspensión o expulsión según se establece en el documento de Política de Integridad Académica con fecha de efectividad de noviembre 2022.

CURSOS DE INVESTIGACIÓN

“Este curso puede requerir que los estudiantes practiquen tareas relacionadas al proceso de investigación, tales como: toma de consentimiento o asentimiento informado, administración de instrumentos, realización de entrevistas, observaciones o grupos focales, entre otros. Estas tareas son parte de un ejercicio académico y no se utilizará la información recopilada para compartirla con terceros o divulgar en otros escenarios que no sean el salón de clases junto al profesor que enseña el curso. Todo estudiante que vaya a interactuar con sujetos humanos como parte de su práctica en investigación tiene que estar certificado en ética con sujetos humanos en la investigación por el *Collaborative Institutional Training Initiative (CITI Program)*, al igual que su profesor”(Los cursos de investigación deben incluir este texto)

Derechos reservados | Sagrado | Noviembre, 2022 (2008)