

Curso Académico: (2023 / 2024)

Fecha de revisión: 19-05-2023

Departamento asignado a la asignatura: Departamento de Informática

Coordinador/a: GARCIA HERRERO, JESUS

Tipo: Obligatoria Créditos ECTS : 6.0

Curso : 4 Cuatrimestre : 2

REQUISITOS (ASIGNATURAS O MATERIAS CUYO CONOCIMIENTO SE PRESUPONE)

Programación (1º, 1C)

Teoría de Autómatas y Lenguajes Formales (2º, 1C)

OBJETIVOS

- Conocimiento

1. Conocimiento sobre la teoría formal de autómatas y lenguajes formales
2. Conocimiento sobre las técnicas de análisis léxico, sintáctico y semántico
3. Técnicas para la generación de código
4. Técnicas para la recuperación de errores
5. Conocimiento de los métodos de optimización de código

- Procedimentales/Instrumentales (Saber hacer)

1. Diseño de una gramática formal
2. Diseñar un analizador léxico y un analizador sintáctico
3. Usar herramientas automáticas de generación de analizadores

- Actitudinales (Ser)

1. Capacidad para generar nuevas ideas (creatividad)
2. Preocupación por la calidad
3. Motivación de logro
4. Interés por investigar y buscar soluciones a nuevos problemas

Competencia de conocimiento:

CGB5 - Conocimiento de la estructura, organización, funcionamiento e interconexión de los sistemas informáticos, los fundamentos

de su programación, y su aplicación para la resolución de problemas propios de la ingeniería.

Resultados de aprendizaje:

R1. Conocimiento y comprensión: Tener conocimientos básicos y la comprensión de los fundamentos científicos y tecnológicos de la Ingeniería Informática, así como un conocimiento específicos de las ciencias de la computación, la ingeniería de computadores y sistemas de información.

R2 Análisis de la Ingeniería: Ser capaces de identificar problemas de Ingeniería Informática, reconocer sus especificaciones, establecer diferentes métodos de resolución y seleccionar el más adecuado para su solución, teniendo en cuenta las limitaciones sociales, salud humana, Medio Ambiente, y comerciales aplicables en cada caso.

R5 Aplicaciones de la Ingeniería: Los egresados serán capaces de aplicar su conocimiento y comprensión para resolver problemas, dirigir investigaciones y diseñar dispositivos o procesos del ámbito de la Ingeniería Informática de acuerdo con criterios de coste, calidad, seguridad, eficiencia, respeto por el medioambiente e implicaciones éticas. Estas habilidades incluyen el conocimiento, uso y limitaciones de sistemas informáticos, ingeniería de procesos, arquitecturas de computadores, modelos computacionales, equipos, trabajo práctico, bibliografía técnica y fuentes de información.

DESCRIPCIÓN DE CONTENIDOS: PROGRAMA

Se presentan los descriptores asociados a la asignatura:

Representación de lenguajes, Análisis léxico, Análisis sintáctico, Análisis Semántico, Generación de código, Recuperación de errores, Optimización de código

ACTIVIDADES FORMATIVAS, METODOLOGÍA A UTILIZAR Y RÉGIMEN DE TUTORÍAS

Clases Teóricas: 1.5 ECTS. Tienen por objetivo alcanzar las competencias específicas cognitivas de la asignatura. (PO a, c)

Clases Prácticas: 1.5 ECTS. Desarrollan las competencias específicas instrumentales y la mayor parte de las transversales, como son la de trabajo en equipo, capacidad de aplicar los conocimientos a la práctica, de planificar y organizar y de análisis y síntesis. También tienen por objetivo desarrollar las capacidades específicas actitudinales. Consisten en el diseño y desarrollo de un proyecto de compilador/intérprete elaborado en grupos de trabajo (PO c, d, e, g, k)

Realización de Actividades Académicas Dirigidas

- Con presencia del profesor: 1 ECTS Planteamiento de un estudio, orientado por el profesor pero propuesto por el alumno, donde profundiza sobre algún aspecto de la materia, realizando una exposición pública del mismo (PO c, d, g).

- Sin presencia del profesor: 1.5 ECTS. Ejercicios y lecturas complementarias propuestas por el profesor (PO a, c).

Ejercicios y Examen: 0.5 ECTS. Tienen por objeto incidir y complementar en el desarrollo de las capacidades específicas cognitivas y procedimentales (PO a, c).

SISTEMA DE EVALUACIÓN

Los ejercicios y exámenes además de servir como actividad formativa tienen el doble objetivo de ser medida para el sistema de evaluación. El sistema de evaluación incluye la valoración de las actividades académicas dirigidas y prácticas según la siguiente ponderación. (No se especifica la relación con las competencias dado que las actividades formativas ya han sido relacionadas con ellas.)

Ejercicios y Examen: 40%

Práctica: 40%

Actividades Académicas Dirigidas:

Con presencia del profesor: 15%

Sin presencia del profesor: 5%

Peso porcentual del Examen Final: 40

Peso porcentual del resto de la evaluación: 60

BIBLIOGRAFÍA BÁSICA

- A. V. Aho and Ravi Sethi and J. D. Ullman Compiladores: Principios, Técnicas y Herramientas, Addison-Wesley Iberoamericana, 1990.
- Kenneth C. Loudon Construcción de Compiladores. Principios y práctica, Thomson, 2004.

BIBLIOGRAFÍA COMPLEMENTARIA

- C. N. Fisher, R. J. Leblanc Crafting a Compiler with C, Addison-Wesley, 1991.
- Dick Grune, Henri E. Bal, Criel J.H. Jacobs, Koen G. Langendoen Modern Compiler Design, John Wiley & Sons, 2000.
- Doug Brown, John Levine, Tony Mason Lex & Yacc, O'Reilly Media, Inc., 1995.
- F. J. Sanchis and C. Galán Compiladores: Teoría y Construcción, Paraninfo, 1986.
- Garrido, Iñesta, Moreno, Pérez Diseño de Compiladores, Publicaciones Universidad de Alicante, 2002.
- K. A. Lemone Fundamentals of Compilers: An Introduction to Computer Language Translation, CRC Press, 1992.
- T. Pittman and J. Peters The Art of Compiler Design: Theory and Practice, Prentice-Hall, Englewood Cliffs, New Jersey, 1992.