Procesadores del Lenguaje

Curso Académico: (2019 / 2020) **Fecha de revisión: 20-04-2020**

Departamento asignado a la asignatura: Departamento de Informática

Coordinador/a: GARCIA HERRERO, JESUS

Tipo: Obligatoria Créditos ECTS : 6.0

Curso : 5 Cuatrimestre : 2

REQUISITOS (ASIGNATURAS O MATERIAS CUYO CONOCIMIENTO SE PRESUPONE)

Programación
Teoría de Autómatas y Lenguajes Formales

OBJETIVOS

Competencias Transversales/Genéricas:
- Capacidad de análisis y síntesis (PO a)
- Capacidad de organizar y planificar (PO c, e)
- Resolución de problemas (PO c)
- Trabajo en equipo (PO d)
- Capacidad de aplicar los conocimientos en la práctica (PO a, c)

Competencias Específicas:
- Cognitivas (Saber)

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

- Procedimentales/Instrumentales (Saber hacer)

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

- Actitudinales (Ser) (PO a, c, d)

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

Competencias básicas establecidas en el artículo 3 sobre competencias RD 1393/2007 modificado por el RD 861/2010 en el que se indica que se garantizarán, como mínimo las siguientes competencias básicas, en el caso del Grado, y aquellas otras que figuren en el Marco Español de Cualificaciones para la Educación Superior, MECES

CECC2. Capacidad para conocer los fundamentos teóricos de los lenguajes de programación y las técnicas de procesamiento léxico, sintáctico y semántico asociadas, y saber aplicarlas para la creación, diseño y procesamiento de lenguajes.

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% (PO a, c)
Práctica: 40% (PO c, d, e, g, k)
Actividades Académicas Dirigidas:
Con presencia del profesor: 15% (PO a, c)
Sin presencia del profesor: 5% (PO a, c, k)

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. Louden 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, Ceriel 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.