Guia docente					
Datos Identificativos					2019/20
Asignatura (*)	Procesamiento de Lenguajes			Código	614G01067
Titulación	Grao en Enxeñaría Informática				
Descriptores					
Ciclo	Periodo	Curso		Tipo	Créditos
Grado	1º cuatrimestre	Cuarto		Optativa	6
Idioma	CastellanoGallego				
Modalidad docente	Presencial				
Prerrequisitos					
Departamento	Ciencias da Computación e Tecnoloxías da InformaciónComputación				
Coordinador/a	Dafonte Vazquez, Jose Carlos		Correo electrónico	carlos.dafonte@udc.es	
Profesorado	Arcay Varela, Bernardino Dafonte Vazquez, Jose Carlos Gomez Garcia, Angel Martinez Perez, Maria		Correo electrónico	bernardino.arcay@udc.es carlos.dafonte@udc.es angel.gomez@udc.es maria.martinez@udc.es	
Web	http://estudos.udc.es/es/study/detail/614G01V01#plan-structure				
Descripción general	Compiladores; traductores e intérpretes; etapas de un compilador; optimización de código; macroprocesadores. El objetivo e familiarizar al alumno con el funcionamiento de los reconocedores del lenguaje y los compiladores como caso particular, el entorno en el que trabajan así como algunas herramientas software para la construcción de los mismos. Se debe asumir la característica interdisciplinar de la asignatura. Adquirir los conocimientos necesarios para diseñar e implementar las diferentes etapas necesarias para el desarrollo de un reconocedor del lenguaje: análisis (léxico, sintáctico y semántico) y síntesis (generación de código intermedio, optimización de código y geración de código objeto).				

Competencias del título	
Código	Competencias del título
A40	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.
B1	Capacidad de resolución de problemas
B3	Capacidad de análisis y síntesis
C2	Dominar la expresión y la comprensión de forma oral y escrita de un idioma extranjero.
C4	Desarrollarse para el ejercicio de una ciudadanía abierta, culta, crítica, comprometida, democrática y solidaria, capaz de analizar la realidad, diagnosticar problemas, formular e implantar soluciones basadas en el conocimiento y orientadas al bien común.
C6	Valorar críticamente el conocimiento, la tecnología y la información disponible para resolver los problemas con los que deben enfrentarse.

Resultados de aprendizaje			
Resultados de aprendizaje	Competencias del título		
Conocer los conceptos teóricos básicos en los que se basan los compiladores. Diseñar e implementar cada una de las fases precisas para la implementación de un compilador. Conocer las distintas herramientas disponibles para la implementación de compiladores y manejar algunas de las más habituales. El estudiane también sabrá verificar el funcionamiento del procesador del lenguaje. En relación a las competencias tipo C, será preciso manejar material en inglés y castellano y se orientará el trabajo de prácticas para obtener un diagnóstico y una solución a los problemas asociados con la temática de la asignatura.	A40	B3	C2 C4
Partiendo de la base teórica de la computación, se plantean diferentes técnicas y herramientas (con una orientación muy práctica) para el procesamiento del lenguaje. En relación a las competencias tipo C, también será necesario manejar documentación en ambos idiomas y se tratará de valorar las diferentes soluciones y planteamientos posibles con sus ventajas e inconvenientes.	A40		C2 C6

Entender las razones y saber optimizar el código objeto en función de la plataforma que lo va a ejecutar. Saber elegir el tipo de optimización según la máquina objeto, sus capacidades y su arquitectura. Con respecto a las competencias tipo C, una vez más precisaremos manejar documentación en varios idiomas y en las prácticas se tratará de proporcionar las herramientas para que el estudiante pueda descubrir por sí mismo las diferentes posibilidades y sus ventajas según el entorno.		B1	C2 C6
Saber analizar y diagnosticar el caso concreto, el objetivo, para el que se precisa un reconocedor del lenguaje. El estudiante también será capaz de documentarse, sintetizar y adaptar la solución con un enfoque totalmente práctico pero basado en conocimiento. Sobre las competencias tipo C, y dado que principalmente este resultado se obtendrá a partir del proyecto propuesto por el estudiante, es obvio que precisamos utilizar técnicas avanzadas basadas en conocimiento, diagnóstico de problemas así como una valoración crítica de la propuesta que él mismo realiza.		B3	C4 C6

Contenidos	
Tema	**Subtema**
Tema I. Introducción	1.1 Estructura de un compilador. 1.2 Ejemplo de las fases de un compilador
Tema II. Lenguajes y gramáticas	2.1 Notación y clasificación de Chomsky. 2.2 Gramáticas de contexto libre (GCL) y notación BNF. 2.3 Mejora y simplificación de gramáticas.
Tema III. Análisis Léxico	3.1 Tipos de máquinas reconocedoras o autómatas. 3.2 Autómatas Finitos. 3.3 Conversión entre gramática regular, expresión regular y autómata finito. 3.4 Implementación de autómatas.
Tema IV. Análisis Sintáctico (Parsing)	4.1 Máquinas teóricas, mecanismos con retroceso 4.2 Algoritmos sin retroceso. 4.2.1 Análisis sintáctico ascendente por precedencia simple. 4.2.2 Analizadores descendentes LL(K). 4.2.3 Analizadores ascendentes LR(k). 4.2.4 Generador de analizadores sintácticos YACC.
Tema V. Análisis Semántico	5.1 Definiciones dirigidas por la sintáxis. 5.2 Esquema de traducción. 5.3 Comprobaciones en tiempo de compilación.
Tema VI. Generación de Código	6.1 Lenguajes intermedios. 6.2 Generación de código intermedio. 6.3 Generación de código desde lenguaje intermedio.
Tema VII. Optimización de Código	7.1 Algoritmo de Nakata. 7.2 Lazos en los grafos de flujo. 7.3 Análisis global del flujo de datos. 7.4 Solución iterativa de las ecuaciones de flujo de datos.
Tema VIII. Errores	8.1 Tipos de errres 8.2 Recuperación de errores léxico-gráficos.
Tema IX. Intérpretes y Complementos	9.1 Estructura de un intérprete actual. 9.2 Arquitectura neutral de Java.

Planificación				
Metodologías / pruebas	**Competéncias**	**Horas presenciales**	**Horas no presenciales / trabajo autónomo**	**Horas totales**
Sesión magistral	A40 B3 C2 C4	21	31.5	52.5
Prueba objetiva	A40 B1 C2 C6	3	18	21

Trabajos tutelados	B1 B3 C2 C4 C6	7	17.5	24.5
Prácticas de laboratorio	A40 B1 C2	14	14	28
Atención personalizada		24	0	24
(*)Los datos que aparecen en la tabla de planificación són de carácter orientativo, considerando la heterogeneidad de los alumnos				

Metodologías	
Metodologías	**Descripción**
Sesión magistral	En estas sesiones se analizarán en profundidad los conceptos teóricos y principales técnicas para la implementación de un reconocedor del lenguaje.
Prueba objetiva	Se realizará un examen al final del cuatrimestre y su correspondiente 2ª oportunidad. Este examen se centrará principalmente en los aspectos teóricos y ejercicios asociados. Sin embargo, podrá preguntarse alguna cuestión relacionada con las prácticas de implementación realizadas.
Trabajos tutelados	Se llevará a cabo un proyecto a propuesta de un grupo de 2 estudiantes del que se realizará un seguimiento en el horario de trabajo en grupo.
Prácticas de laboratorio	Se llevarán a cabo pequeñas prácticas sobre los diferentes temas de la asignatura que se irán desarrollando en las sesiones magistrales. Algunas de ellas serán ejercicios de desarrollo y otras de implementación.

Atención personalizada	
Metodologías	**Descripción**
Sesión magistral Prácticas de laboratorio Trabajos tutelados	Especialmente en el caso del proyecto a desarrollar, organizado por grupos de alumnos, se realizará un seguimiento. A ello se dedicarán principalmente las horas de trabajo en grupo reducido.

Evaluación			
Metodologías	**Competéncias**	**Descripción**	**Calificación**
Prácticas de laboratorio	A40 B1 C2	Se llevarán a cabo pequeñas prácticas sobre los diferentes temas de la asignatura que se irán desarrollando en las sesiones magistrales. Algunas de ellas serán ejercicios de desarrollo y otras de implementación. El porcentaje por asistencia y entrega de estas pequeñas prácticas es testimonial (10%). El 20% restante corresponde al desarrollo del proyecto.	30
Prueba objetiva	A40 B1 C2 C6	Se realizará un examen al final del cuatrimestre y su correspondiente 2ª oportunidad. Este examen se centrará principalmente en los aspectos teoricos y ejercicios asociados. Sin embargo, podrá preguntarse alguna cuestión relacionada con las prácticas de implementación realizadas.	50
Trabajos tutelados	B1 B3 C2 C4 C6	Se llevará a cabo un proyecto a propuesta de un grupo de 2 estudiantes del que se realizará un seguimiento en el horario de trabajo en grupo.	20
Otros			

Observaciones evaluación
En cualquier caso, es preciso aprobar las dos partes (prueba objetiva y prácticas-trabajos). En caso contrario, la máxima nota que se podrá conseguir es un 4.5. IMPORTANTE: El 40% de la nota lo aporta el proyecto que propondrán los grupos, el 20% corresponde al seguimiento y el 20% a la entrega final. ESTUDIANTES CON MATRÍCULA A TIEMPO PARCIAL O CON DISPENSA ACADÉMICA DE EXENCIÓN DE DOCENCIADeberán ponerse en contacto con los profesores de la asignatura para posibilitar la realización de las tareas fuera de la organización habitual de la materia.

Fuentes de información	
Básica	

Complementária	"Compiladores: Principios, técnicas y herramientas", Aho, A.V.; Lam M.; Sethi, R. ; Ullman, J.D., Addison-Wesley, Reading, Massachussetts 2008."Construcción de compiladores. Principios y Práctica", Louden D. K., Paraninfo Thomson Learning, 2004.Garrido, A. ; Iñesta J.M. ; Moreno F. ; Pérez J.A. [2004] Diseño de compiladores, Publicaciones Universidad de Alicante."Compiladores, teoría y construcción", Sanchis, F.J.; Galán, J.A., Ed. Paraninfo, 1987."The theory of parsing, translation and compiling" (I y II), Aho, A.V.; Ullman, J.D., Prentice-Hall, 1972."Principles of compiler design", Aho, A.V.; Ullman J.D., Addison-Wesley, 1977."ntroducción a la teoría de autómatas, lenguajes y computación", Hopcroff, J.E. ; Motwani R. ; Ullman, J. D. [2002] I, Addison-Wesley, 2002"Compiler design in C", Allen I.; Holub, Prentice-Hall, 1991."Compiladores e Intérpretes", Sánchez, G.; Valverde J.A., Ed. Díaz de Santos, 1984.?Languages and machines?, Sudkamp T.A., Addison-Wesley, 1994

Recomendaciones
Asignaturas que se recomienda haber cursado previamente
Teoría de la computación/614G01039
Asignaturas que se recomienda cursar simultáneamente
Asignaturas que continúan el temario
Diseño de los Lenguajes de Programación/614G01065
Otros comentarios
Aunque tiene relación con otra asignaturas, está fuertemente relacionada con "Diseño de Lenguajes de Programación", que la complementa, y "Teoría de la Computación", que es la base inicial para poder afrontar los contenidos.

(*) La Guía Docente es el documento donde se visualiza la propuesta académica de la UDC. Este documento es público y no se puede modificar, salvo cosas excepcionales bajo la revisión del órgano competente de acuerdo a la normativa vigente que establece el proceso de elaboración de guías