



Guía Docente				
Datos Identificativos				2014/15
Asignatura (*)	Teoría de Autómatas e Linguaxes Formais		Código	614311302
Titulación	Enxeñeiro Técnico en Informática de Sistemas			
Descriptorios				
Ciclo	Período	Curso	Tipo	Créditos
1º e 2º Ciclo	2º cuatrimestre	Terceiro	Troncal	8
Idioma	CastelánGalego			
Prerrequisitos				
Departamento	Computación			
Coordinación	Graña Gil, Jorge	Correo electrónico	jorge.grana@udc.es	
Profesorado	Graña Gil, Jorge	Correo electrónico	jorge.grana@udc.es	
Web	campusvirtual.udc.es			
Descrición xeral	Se trata de una asignatura troncal, que se imparte de manera cuatrimestral en el tercer curso de la titulación de Ingeniero en Informática. Destaca el carácter integrador de su contenido, ya que sirve de puente entre lo que podemos denominar una "visión de usuario" de los lenguajes informáticos, representada por la programación estándar, y una "visión generativa" de éstos, en la que el alumno construye y adecúa un lenguaje de programación en atención a sus requerimientos. Finalmente, se transmite también al alumno una visión formal de los fundamentos propios de la ciencia de la computación.			

Competencias da titulación	
Código	Competencias da titulación
A1	Analizar novas técnicas e ferramentas do mercado estudando a súa viabilidade e necesidade. Posibilidade de contratar recursos externos.
A3	Interpretar as especificacións funcionais encamiñadas ao desenvolvemento das aplicacións informáticas.
B1	Aprender a aprender.
B2	Resolver problemas de forma efectiva.
B3	Aplicar un pensamento crítico, lóxico e creativo.
B4	Aprendizaxe autónoma.
B5	Traballar de forma colaborativa.
B8	Traballar en equipos de carácter interdisciplinar.
C6	Valorar criticamente o coñecemento, a tecnoloxía e a información dispoñible para resolver os problemas cos que deben afrontarse.

Resultados da aprendizaxe			
Competencias de materia (Resultados de aprendizaxe)	Competencias da titulación		
Conocer en profundidad la estructura y función de los sistemas de descripción y reconocimiento de lenguajes formales.	A1	B1 B4	
Estudiar los conceptos, modelos y técnicas relacionados con estas cuestiones.	A1	B1 B4	
Conocer las estructuras de datos y los algoritmos utilizados para implementar los distintos modelos de reconocimiento de lenguajes formales, así como sus posibles dominios de aplicación práctica.	A1 A3	B1 B4	C6
Realizar implementaciones de estos modelos en alguno de esos dominios.	A1 A3	B2 B3 B5	C6
Sintetizar todos los conceptos estudiados en ideas concretas que permitan comprender mejor los fundamentos de la computación.	A1	B1 B4	
Perfeccionar las habilidades para realizar futuros trabajos de análisis, diseño y programación.	A1 A3	B2 B3 B5	C6



Considerar la integración de las técnicas y estructuras estudiados aquí en otros dominios de aplicación.	A1	B2	C6
	A3	B3	
		B5	
		B8	

Contidos	
Temas	Subtemas
Preliminares matemáticos	Lógica elemental Teoría de conjuntos Relaciones y funciones Inducción matemática Cardinalidad
Lenguajes formales	Alfabetos, palabras y lenguajes Operaciones con palabras Operaciones con lenguajes
Lenguajes regulares y autómatas finitos	Lenguajes sobre alfabetos Lenguajes regulares y expresiones regulares Autómata finito determinista (AFD) Autómata finito no determinista (AFN) Equivalencia entre AFNs y AFDs Autómata finito con epsilon transiciones Autómatas finitos y expresiones regulares Aplicaciones prácticas de las expresiones regulares y de los autómatas finitos
Lenguajes independientes del contexto y autómatas de pila	Gramáticas regulares Gramáticas regulares y lenguajes regulares Gramáticas independientes del contexto Arboles de derivación y ambigüedad Simplificación de gramáticas independientes del contexto Propiedades de los lenguajes independientes del contexto Algoritmos de análisis sintáctico Autómatas de pila Forma normal de Greibach
Máquinas de Turing	Definiciones básicas Máquinas de Turing como aceptadoras de lenguajes Construcción de máquinas de Turing Modificaciones de las máquinas de Turing Máquina de Turing universal
Lenguajes recursivamente enumerables	Lenguajes aceptados por máquinas de Turing Lenguajes regulares e independientes del contexto como lenguajes recursivos Propiedades de los lenguajes recursivos y recursivamente enumerables Gramáticas no restringidas y lenguajes recursivamente enumerables Lenguajes sensibles al contexto y la jerarquía de Chomsky
Resolubilidad	El problema de la parada El problema de correspondencia de Post Problemas no decidibles en lenguajes independientes del contexto
Computabilidad	Fundamentos de la teoría de funciones recursivas Alcance de las funciones recursivas primitivas Funciones recursivas parciales El poder de los lenguajes de programación



Introducción a la teoría de la complejidad computacional	<p>Complejidad algorítmica</p> <p>Modelo general de cómputo y complejidad computacional</p> <p>Tiempo y espacio en máquinas de Turing</p> <p>Las distintas clases de complejidad</p> <p>Los problemas tratables y no tratables</p> <p>Reducibilidad en tiempo polinómico</p> <p>Problemas NP-completos</p>
--	--

Planificación			
Metodoloxías / probas	Horas presenciais	Horas non presenciais / traballo autónomo	Horas totais
Sesión maxistral	30	60	90
Prácticas de laboratorio	12	24	36
Proba de resposta múltiple	4	4	8
Traballos tutelados	1	6	7
Seminario	3	0	3
Aprendizaxe colaborativa	4	4	8
Solución de problemas	3	16	19
Proba de ensaio	3	16	19
Atención personalizada	10	0	10

\*Os datos que aparecen na táboa de planificación son de carácter orientativo, considerando a heteroxeneidade do alumnado

Metodoloxías	
Metodoloxías	Descrición
Sesión maxistral	La técnica que mejor se adapta a la impartición de los contenidos teóricos de esta asignatura está constituida por las clases magistrales. En ellas, haremos un uso intensivo de la pizarra y de las transparencias, de modo que el ritmo de exposición de conceptos por parte del profesor y el de asimilación de los mismos por parte del alumno sean lo más acordes posible.
Prácticas de laboratorio	Las prácticas de laboratorio tendrán horas de laboratorio reservadas, con ordenadores a disposición de los alumnos. Estas horas serán utilizadas para implementar en algún lenguaje de programación los algoritmos más destacados, de entre todos aquéllos que hayan sido presentados en las sesiones teóricas.
Proba de resposta múltiple	Se realizarán controles tipo test al final de cada bloque temático, que permitirán al profesor conocer el grado de asimilación de la materia por parte de los alumnos, y modificar la estrategia docente si es necesario.
Traballos tutelados	Los trabajos de grupos autónomos tutelados se realizarán a lo largo de todo el cuatrimestre. El profesor elegirá un tema de trabajo que será asignado por igual a todos los grupos. El tema será presentado por el profesor en una sesión en el aula, será desarrollado por los alumnos en horas no presenciales, y será supervisado y evaluado por el profesor en las tutorías en grupo. La evaluación se realizará a partir de la exposición de una memoria final por parte de los alumnos.
Seminario	Los seminarios se implementarán bajo la forma de un ciclo de charlas cortas o conferencias, sobre aplicaciones prácticas relacionadas con la materia de la asignatura. El objetivo de estas charlas es el de completar la percepción general que el alumno tiene sobre cómo los conceptos vistos en clase son puestos en práctica en la vida real.
Aprendizaxe colaborativa	Los trabajos de grupos cooperativos se realizarán utilizando la &quot;técnica puzzle&quot; de Aronson. Según esta técnica, el profesor elige previamente un tema de trabajo y divide a los alumnos en varios grupos. A cada miembro de cada grupo se le asigna una sección de dicho tema, para que la estudie y la comente con los miembros de los otros grupos a los que les ha sido asignada la misma sección. Posteriormente, cada alumno regresa a su grupo y explica al resto de miembros su sección. Finalmente, el profesor realiza a los alumnos un test general sobre el tema elegido. Dicho test permitirá al profesor no sólo evaluar el grado de comprensión de los nuevos conocimientos adquiridos, sino también el nivel de cooperación que ha tenido lugar entre los miembros de cada grupo concreto.



Solución de problemas	Se pondrán a disposición de los alumnos una serie de boletines de ejercicios, correspondientes a los bloques temáticos del programa de la asignatura. Los alumnos deberán entregar al profesor sus soluciones personales a estos ejercicios. El profesor deberá corregirlas, evaluarlas y comentarlas durante al menos una sesión en el aula.
Proba de ensaio	Se implementará bajo la forma de un examen final escrito.

### Atención personalizada

Metodoloxías	Descrición
Prácticas de laboratorio Traballos tutelados	<p>Dado el carácter personalizado de las prácticas de laboratorio, de los trabajos tutelados y de las tutorías, estas actividades no deben dedicarse a extender los contenidos con nuevos conceptos, sino a aclarar los conceptos ya expuestos.</p> <p>El profesor debe además utilizarlas como una interacción que le permita extraer conclusiones respecto al grado de asimilación de la materia por parte de los alumnos.</p> <p>De esta manera, podrá desarrollar las clases magistrales y el resto de actividades no personalizadas atendiendo al progreso de los alumnos en las capacidades de comprensión y asimilación de los contenidos impartidos, compaginando el avance general de la clase con una atención específica a aquellos alumnos que presenten mayores dificultades en la tarea del aprendizaje y con un apoyo adicional a aquellos otros que presenten mayor desenvoltura y deseen ampliar conocimientos.</p>

### Avaliación

Metodoloxías	Descrición	Cualificación
Prácticas de laboratorio	Implementación de algoritmos en algún lenguaje de programación	25
Proba de resposta múltiple	Controles tipo test	7.5
Traballos tutelados	Trabajo de grupos autónomos tutelados	5
Aprendizaxe colaborativa	Trabajo de grupos cooperativos	7.5
Solución de problemas	Boletines de ejercicios	5
Proba de ensaio	Examen final escrito	50
Outros		

### Observacións avaliación

En el examen final se requiere una nota mínima de 3 puntos (sobre 10).
--

### Fontes de información

<b>Bibliografía básica</b>	<ul style="list-style-type: none"> <li>- John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman (2002). Introducción a la teoría de autómatas, lenguajes y computación. Addison Wesley</li> <li>- Thomas A. Sudkamp (1988). Languages and machines: an introduction to the theory of computer science. Addison Wesley</li> <li>- Dean Kelley (1995). Teoría de autómatas y lenguajes formales. Prentice Hall</li> </ul>
<b>Bibliografía complementaria</b>	<ul style="list-style-type: none"> <li>- Harry R. Lewis, Christos H. Papadimitriou (1998). Elements of the theory of computation. Prentice Hall</li> <li>- Peter J. Denning, Jack B. Dennis, Joseph E. Qualitz (1978). Machines, languages and computation. Prentice Hall</li> <li>- J. Glenn Brookshear (1993). Teoría de la computación: lenguajes formales, autómatas y complejidad. Addison Wesley Iberoamericana</li> </ul>



## Recomendacións

### Materias que se recomenda ter cursado previamente

Estrutura de Datos e da Información/614111102

Álgebra/614111106

Matemática Discreta I/614111107

Programación/614111109

Algoritmos/614111206

Programación Declarativa/614111207

### Materias que se recomenda cursar simultaneamente

### Materias que continúan o temario

Intelixencia Artificial/614111404

Compiladores/614111405

Linguaxes Naturais/614111625

### Observacións

(\*A Guía docente é o documento onde se visualiza a proposta académica da UDC. Este documento é público e non se pode modificar, salvo casos excepcionais baixo a revisión do órgano competente dacordo coa normativa vixente que establece o proceso de elaboración de guías