



Teaching Guide				
Identifying Data				2016/17
Subject (*)	Theoretical Computer Science	Code	614G01039	
Study programme	Grao en Enxeñaría Informática			
Descriptors				
Cycle	Period	Year	Type	Credits
Graduate	2nd four-month period	Third	Optativa	6
Language	SpanishGalician			
Teaching method	Face-to-face			
Prerequisites				
Department	Computación			
Coordinador	Blanco Ferro, Antonio angel	E-mail	antonio.blanco.ferro@udc.es	
Lecturers	Blanco Ferro, Antonio angel Graña Gil, Jorge Novo Bujan, Jorge	E-mail	antonio.blanco.ferro@udc.es jorge.grana@udc.es j.novo@udc.es	
Web	moodle.udc.es			
General description	<p>Trátase dunha materia na que destaca o carácter integrador do seu contido, xa que serve de ponte entre o que podemos denominar unha "visión de usuario" das linguaxes informáticas, representada pola programación estándar, e unha "visión xerativa" destes, na que o alumno constrúe e adecúa unha linguaxe de programación en atención aos seus requirimentos. Finalmente, transmítese tamén ao alumno unha visión formal dos fundamentos propios da ciencia da computación.</p> <p>Se trata de una asignatura en la que 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.</p>			

Study programme competences / results	
Code	Study programme competences / results
A39	Capacidade para ter un coñecemento profundo dos principios fundamentais e modelos da computación, e saber aplicalos para interpretar, seleccionar, valorar, modelar, e crear novos conceptos, teorías, usos e desenvolvementos tecnolóxicos relacionados coa informática.
A40	Capacidade para coñecer os fundamentos teóricos das linguaxes de programación e as técnicas de procesamento léxico, sintáctico e semántico asociadas, e saber aplicalas para a creación, o deseño e o procesamento de linguaxes.
A41	Capacidade para avaliar a complexidade computacional dun problema, coñecer estratexias algorítmicas que poidan conducir á súa resolución e recomendar, desenvolver e implementar aquela que garanta o mellor rendemento de acordo cos requisitos establecidos.
B1	Capacidade de resolución de problemas
B2	Traballo en equipo
B3	Capacidade de análise e síntese
B6	Toma de decisións
B8	Capacidade de traballar nun equipo interdisciplinar
C6	Valorar criticamente o coñecemento, a tecnoloxía e a información dispoñible para resolver os problemas cos que deben enfrontarse.
C7	Asumir como profesional e cidadán a importancia da aprendizaxe ao longo da vida.

Learning outcomes			
Learning outcomes			Study programme competences / results
Coñecer en profundidade a estrutura e función dos sistemas de descrición e recoñecemento de linguaxes formais.			A39 A40
			B6
			C7



Estudar os conceptos, modelos e técnicas relacionados con estas cuestións.	A39 A40	B6	C7
Coñecer as estruturas de datos e os algoritmos utilizados para implementar os distintos modelos de recoñecemento de linguaxes formais, así como os seus posibles dominios de aplicación práctica.	A41	B6	C6 C7
Realizar implementacións destes modelos nalgún deses dominios.	A41	B1 B2 B3	C6
Sintetizar todos os conceptos estudados en ideas concretas que permitan comprender mellor os fundamentos da computación.	A39	B6	C7
Perfeccionar as habilidades para realizar futuros traballos de análises, deseño e programación.	A40 A41	B1 B2 B3	C6
Considerar a integración das técnicas e estruturas estudadas aquí noutros dominios de aplicación.	A40 A41	B1 B2 B3 B8	C6

Contents	
Topic	Sub-topic
Preliminares sobre linguaxes formais	Alfabetos, palabras e linguaxes Linguaxes regulares e expresións regulares Autómatas finitos
Linguaxes independentes do contexto e autómatas de pila	Gramáticas regulares Gramáticas regulares e linguaxes regulares Gramáticas independentes do contexto Árbores de derivación e ambigüidade Simplificación de gramáticas independentes do contexto Propiedades das linguaxes independentes do contexto Algoritmos de análise sintáctico Autómatas de pila Forma normal de Greibach
Máquinas de Turing	Definicións básicas Máquinas de Turing como aceptadoras de linguaxes Construción de máquinas de Turing Modificacións das máquinas de Turing Máquina de Turing universal
Linguaxes recursivamente enumerables	Linguaxes aceptadas por máquinas de Turing Linguaxes regulares e independentes do contexto como linguaxes recursivas Propiedades das linguaxes recursivas e recursivamente enumerables Gramáticas non restrinxidas e linguaxes recursivamente enumerables Linguaxes sensíbeis ao contexto e a xerarquía de Chomsky
Resolubidade	O problema da parada O problema de correspondencia de Post Problemas non decidibles en linguaxes independentes do contexto
Computabilidade	Fundamentos da teoría de funcións recursivas Alcance das funcións recursivas primitivas Funcións recursivas parciais O poder das linguaxes de programación

Planning



Methodologies / tests	Competencies / Results	Teaching hours (in-person & virtual)	Student?s personal work hours	Total hours
Guest lecture / keynote speech	A39 A40 B8 C6 C7	18	36	54
Laboratory practice	A40 A41 B1 B2 B3 B6 B8 C6	13	26	39
Short answer questions	A39 A40 B1 C6 C7	2	4	6
Supervised projects	A39 B2 B8 C6 C7	1	5.5	6.5
Problem solving	B1 B3 B6	4	17	21
Objective test	A39 A40 B1 C6 C7	3	16	19
Personalized attention		4.5	0	4.5

(*)The information in the planning table is for guidance only and does not take into account the heterogeneity of the students.

Methodologies	
Methodologies	Description
Guest lecture / keynote speech	A técnica que mellor se adapta á impartición dos contidos teóricos desta materia está constituída polas clases maxistras. Nelas, faremos un uso intensivo da lousa e das transparencias, de modo que o ritmo de exposición de conceptos por parte do profesor e o de asimilación dos mesmos por parte do alumno sexan o máis acordes posible.
Laboratory practice	As prácticas de laboratorio terán horas de laboratorio reservadas, con computadores a disposición dos alumnos. Estas horas serán utilizadas para implementar nalgunha linguaxe de programación os algoritmos máis destacados, de entre todos aqueles que fosen presentados nas sesións teóricas.
Short answer questions	Realizaranse controles ao final de cada bloque temático, que permitirán ao profesor coñecer o grao de asimilación da materia por parte dos alumnos, e modificar a estratexia docente se é necesario.
Supervised projects	Os traballos de grupos autónomos tutelados realizaranse ao longo de todo o cuadrimestre. O profesor elixirá un tema de traballo que será asignado por igual a todos os grupos. O tema será presentado polo profesor nunha sesión na aula, será desenvolvido polos alumnos en horas non presenciais, e será supervisado e avaliado polo profesor nas titorías en grupo. A avaliación realizarase a partir da exposición dunha memoria final por parte dos alumnos.
Problem solving	Poranse a disposición dos alumnos unha serie de boletíns de exercicios, correspondentes aos bloques temáticos do programa da materia. Os alumnos deberán entregar ao profesor as súas solucións persoais a estes exercicios. O profesor deberá corríxilas, avalialas e comentalas durante polo menos unha sesión na aula.
Objective test	Implementarase baixo a forma dun exame final escrito.

Personalized attention	
Methodologies	Description
Supervised projects Laboratory practice	Dado o carácter personalizado das prácticas de laboratorio, dos traballos tutelados e das titorías, estas actividades non deben dedicarse a estender os contidos con novos conceptos, senón a aclarar os conceptos xa expostos. O profesor debe ademais utilizalas como unha interacción que lle permita extraer conclusións respecto ao grao de asimilación da materia por parte dos alumnos. Desta maneira, poderá desenvolver as clases maxistras e o resto de actividades non personalizadas atendendo ao progreso dos alumnos nas capacidades de comprensión e asimilación dos contidos impartidos, compaxinando o avance xeral da clase cunha atención específica a aqueles alumnos que presenten maiores dificultades na tarefa da aprendizaxe e cun apoio adicional a aqueles outros que presenten maior soltura e desexen ampliar coñecementos.

Assessment			
Methodologies	Competencies / Results	Description	Qualification
Objective test	A39 A40 B1 C6 C7	Exame final escrito. (**)	0



Problem solving	B1 B3 B6	Boletíns de exercicios e controis dos mesmos.	5
Short answer questions	A39 A40 B1 C6 C7	Controles con cuestións teóricas e prácticas ao final de cada bloque temático. (*)	60
Supervised projects	A39 B2 B8 C6 C7	Traballo de grupos autónomos tutelados.	5
Laboratory practice	A40 A41 B1 B2 B3 B6 B8 C6	Implementación de algoritmos nalgunha linguaxe de programación e resolución de problemas.	30
Others			

Assessment comments

(*) A materia dividirase en tres bloques temáticos. Ao final de cada bloque temático, realizarase un control con cuestións teóricas e prácticas. Cada control poderá consolidar ata un 20% da cualificación. A porcentaxe correspondente aos controis non superados pasará a computarse na proba obxectiva (examen final). Os alumnos que superen os tres controis, non terán que realizar o exame final.

(**) No caso de ter que realizar o exame final, requírese que o alumno obtenga unha nota mínima de 3 puntos (sobre 10).

Os alumnos a tempo parcial terán consideracións adecuadas á súa situación.

Sources of information

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

Recommendations

Subjects that it is recommended to have taken before

Programming I/614G01001
 Discrete Mathematics/614G01004
 Programming II/614G01006
 Algebra/614G01010
 Algorithms/614G01011
 Programming Paradigms/614G01014

Subjects that are recommended to be taken simultaneously

Subjects that continue the syllabus

Knowledge Representation and Automatic Reasoning/614G01036
 Information Retrieval/614G01040
 Programming Language Design/614G01065
 Programming Language Design/614G01067

Other comments

(*)The teaching guide is the document in which the URV publishes the information about all its courses. It is a public document and cannot be modified. Only in exceptional cases can it be revised by the competent agent or duly revised so that it is in line with current legislation.