



DATOS IDENTIFICATIVOS

Procesadores de linguaxe

Materia	Procesadores de linguaxe			
Código	O06G150V01604			
Titulación	Grao en Enxeñaría Informática			
Descritores	Creditos ECTS	Sinale	Curso	Cuadrimestre
	6	OB	3	2c
Lingua de impartición	Castelán Galego			
Departamento	Informática			
Coordinador/a	González Dacosta, Jacinto Pavon Rial, Maria Reyes			
Profesorado	González Dacosta, Jacinto Pavon Rial, Maria Reyes			
Correo-e	jdacosta@uvigo.es pavon@uvigo.es			
Web	http://faitic.uvigo.es			
Descrición xeral	Nesta materia inclúense as competencias básicas e imprescindibles para o futuro exercicio profesional de Enxeñeiro/a Técnico/a en Informática, para entender, colaborar, avisar e decidir no complexo ámbito do deseño de software, entre outros a comprensión do funcionamento de compiladores, tradutores e o proceso de especificación formal de información.			

Competencias de titulación

Código	
A1	Capacidade para a resolución dos problemas matemáticos que poidan suscitarse na enxeñería. Aptitude para aplicar os coñecementos sobre: álgebra lineal; cálculo diferencial e integral; métodos numéricos; algorítmica numérica; estatística e optimización
A3	Capacidade para comprender e dominar os conceptos básicos de matemática discreta, lóxica, algorítmica e complexidade computacional, e a súa aplicación para a resolución de problemas propios da enxeñería
A4	Coñecementos básicos sobre o uso e programación dos ordenadores, sistemas operativos, bases de datos e programas informáticos con aplicación na enxeñería
A5	Coñecemento da estrutura, organización, funcionamento e interconexión dos sistemas informáticos, os fundamentos da súa programación, e a súa aplicación para a resolución de problemas propios da enxeñería
A7	Capacidade para deseñar, desenvolver, seleccionar e avaliar aplicacións e sistemas informáticos, asegurando a súa fiabilidade, seguridade e calidade, conforme aos principios éticos e á lexislación e normativa vixente
A12	Coñecemento e aplicación dos procedementos algorítmicos básicos das tecnoloxías informáticas para deseñar solucións a problemas, analizando a idoneidade e complexidade dos algoritmos propostos
A13	Coñecemento, deseño e utilización de forma eficiente dos tipos e estruturas de datos máis axeitados á resolución dun problema
A14	Capacidade para analizar, deseñar, construír e manter aplicacións de forma robusta, segura e eficiente, elixindo o paradigma e as linguaxes de programación máis axeitadas
A15	Capacidade de coñecer, comprender e avaliar a estrutura e arquitectura dos computadores, así como os compoñentes básicos que os conforman
A22	Coñecemento e aplicación dos principios, metodoloxías e ciclos de vida da enxeñería de software
A25	Capacidade para desenvolver, manter e avaliar servizos e sistemas software que satisfagan todos os requisitos do usuario e se comporten de forma fiable e eficiente, sexan asequibles de desenvolver e manter e cumpran normas de calidade, aplicando as teorías, principios, métodos e prácticas da Enxeñería do Software
A26	Capacidade para valorar as necesidades do cliente e especificar os requisitos software para satisfacer estas necesidades, reconciliando obxectivos en conflito mediante a procura de compromisos aceptables dentro das limitacións derivadas do custo, do tempo, da existencia de sistemas xa desenvolvidos e das propias organizacións
A27	Capacidade de dar solución a problemas de integración en función das estratexias, estándares e tecnoloxías dispoñibles

A28	Capacidade de identificar e analizar problemas e deseñar, desenvolver, implementar, verificar e documentar solucións software sobre a base dun coñecemento axeitado das teorías, modelos e técnicas actuais
A32	Capacidade para seleccionar, deseñar, despregar, integrar, avaliar, construír, xestionar, explotar e manter as tecnoloxías de hardware, software e redes, dentro dos parámetros de custo e calidade adecuados
B1	Capacidade de análise, síntese e avaliación
B2	Capacidade de organización e planificación
B3	Comunicación oral e escrita na lingua nativa
B5	Capacidade de abstracción: capacidade de crear e utilizar modelos que reflectan situacións reais
B7	Capacidade de buscar, relacionar e estruturar información proveniente de diversas fontes e de integrar ideas e coñecementos
B8	Resolución de problemas
B9	Capacidade de tomar decisións
B10	Capacidade para argumentar e xustificar lóxicamente as decisións tomadas e as opinións
B11	Capacidade de actuar autonomamente
B12	Capacidade de traballar en situacións de falta de información e/ou baixo presión
B13	Capacidade de integrarse rapidamente e traballar eficientemente en equipos unidisciplinares e de colaborar nun entorno multidisciplinar
B15	Capacidade de relación interpersoal
B16	Razoamento crítico
B17	Compromiso ético e democrático
B18	Aprendizaxe autónoma
B19	Adaptación a novas situacións
B20	Creatividade
B21	Liderado
B22	Ter iniciativa e ser resolutivo
B24	Ter motivación pola calidade e a mellora continua

Competencias de materia

Resultados previstos na materia	Resultados de Formación e Aprendizaxe	
Coñecer os fundamentos matemáticos dos autómatas e linguaxes formais.	A1	B1
	A3	B3
		B5
		B7
		B8
		B9
		B10
		B11
		B12
		B13
		B15
		B16
		B17
		B18
		B19
		B20
		B21
		B22
		B24

Coñecer os algoritmos básicos con autómatas e linguaxes formais.

A3
A12

B1
B3
B5
B7
B8
B9
B10
B11
B12
B13
B15
B16
B17
B18
B19
B20
B21
B22
B24

Coñecer o proceso de xeración de código dende a formulación formal do problema ata a execución da aplicación no ordenador.

A1
A3
A4
A5
A7
A12
A13
A15
A22
A25
A26
A27
A28
A32

B1
B2
B3
B5
B7
B8
B9
B10
B11
B12
B13
B15
B16
B17
B18
B19
B20
B21
B22
B24

Coñecer as ferramentas e as súas propiedades en uso para xerar código de máquina con linguaxes de programación de diferentes tipos.

A4
A5
A7
A12
A13
A14
A15
A22
A25
A26
A27
A28
A32

B1
B2
B3
B5
B7
B8
B9
B10
B11
B12
B13
B15
B16
B17
B18
B19
B20
B21
B22
B24

Coñecer métodos de adaptación e optimización de código.

A4
A5
A7
A12
A13
A15
A22
A25
A26
A27
A28
A32
B1
B2
B3
B5
B7
B8
B9
B10
B11
B12
B13
B15
B16
B17
B18
B19
B20
B21
B22
B24

Contidos

Tema

Fundamentos da teoría de autómatas e linguaxes (*) (*)
formais

Conceptos da teoría (sintaxis e semántica) en linguaxes de programación de diferentes tipos (*) (*)

Funcionamento de procesadores de linguaxes (*) (*)

Xeración de código e a súa optimización (*) (*)

Planificación

	Horas na aula	Horas fóra da aula	Horas totais
Sesión maxistral	20	10	30
Presentacións/exposicións	4	26	30
Prácticas de laboratorio	30	30	60
Resolución de problemas e/ou exercicios	6	24	30

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

Metodoloxía docente

	Descrición
Sesión maxistral	Exposición dos contidos teóricos da materia. Co fin de facilitar a comprensión da materia e aumentar o interese do alumno, inclúense diversos exemplos e exercicios nos que se pode requirir a participación activa do alumno.
Presentacións/exposicións	Preparación e presentación en pequenos grupos de un tema, a súa exposición oral, planteamento de aplicacións e avaliación das mesmas.
Prácticas de laboratorio	Realización de actividades de carácter práctico que inclúen exercicios, investigacións, resolución de problemas e desenvolvemento de aplicacións relacionadas cos contidos da materia.

Atención personalizada

Metodoloxías	Descrición
Presentacións/exposicións	Atención ás preguntas e dúbidas dos alumnos que poidan surxir ó longo do traballo a realizar nas clases. No caso das presentacións/exposicións ó profesor guiará ós alumnos propondo unha bibliografía básica inicial coa que iniciar o traballo de investigación e supervisando o desenvolvemento do mesmo.
Prácticas de laboratorio	Atención ás preguntas e dúbidas dos alumnos que poidan surxir ó longo do traballo a realizar nas clases. No caso das presentacións/exposicións ó profesor guiará ós alumnos propondo unha bibliografía básica inicial coa que iniciar o traballo de investigación e supervisando o desenvolvemento do mesmo.

Avaliación

	Descrición	Cualificación
Presentacións/exposicións	Preparación e presentación en pequenos grupos de un tema, a súa exposición oral, planteamento de aplicacións e avaliación das mesmas.	20

Prácticas de laboratorio	Realización dos exercicios propostos	40
Resolución de problemas e/ou exercicios	Realización de diferentes actividades (p.ex. lectura de bibliografía, elaboración de esquemas, estudio de manuais) ao longo do curso que recollerán contidos de carácter teórico e práctico correspondentes á materia impartida durante as clases da aula. Realización de un exame escrito final.	40

Outros comentarios sobre a Avaliación

Bibliografía. Fontes de información

Kenneth C. Louden, **Construcción de Compiladores: Principios y Práctica**, Primera,
A. Aho, M.S. Lam, R. Sethi, J.D. Ullman e S. Fuenlabrada Velázquez, **Compiladores : principios, técnicas y herramientas**, Segunda,
D. Grune, H.E. Bal e C.J.H. Jacobs, **Diseño de Compiladores Modernos**, Primera,
D. Grune, H.E. Bal e C.J.H. Jacobs, **Modern Compiler Design**, Primera,
K.C. Louden, **Compiler Construction: Principles and Practice**, Primera,

Recomendacións

Materias que se recomenda ter cursado previamente

Informática: Algoritmos e estruturas de datos I/O06G150V01201
Matemáticas: Fundamentos matemáticos para a informática/O06G150V01103
Informática: Programación I/O06G150V01104
Programación II/O06G150V01205
Algoritmos e estruturas de datos II/O06G150V01302
Linguaxes de programación/O06G150V01504