



Guía Docente				
Datos Identificativos				2016/17
Asignatura (*)	Validación e Verificación do Software	Código	614G01225	
Titulación	Grao en Enxeñaría Informática			
Descritores				
Ciclo	Período	Curso	Tipo	Créditos
Grao	2º cuatrimestre	Curso de Adaptación Enxeñeiros Téc. en Informática	Obrigatoria	6
Idioma	Castelán			
Modalidade docente	Presencial			
Prerrequisitos				
Departamento	Computación			
Coordinación		Correo electrónico		
Profesorado		Correo electrónico		
Web	<a href="http://guiadocente.udc.es/guia_docent/index.php?centre=614&amp;ensenyament=614G01&amp;assignatura=614G01053&amp;any_academic=2016_17">guiadocente.udc.es/guia_docent/index.php?centre=614&amp;ensenyament=614G01&amp;assignatura=614G01053&amp;any_academic=2016_17</a>			
Descrición xeral	Consultar:  614G01053 Cuarto 1º cuatrimestre Validación y Verificación del Software 3 3 Grao en Enxeñaría Informática Obrigatoria Computación  Castro Souto, Laura Milagros			

Competencias / Resultados do título	
Código	Competencias / Resultados do título
A28	Capacidade de identificar e analizar problemas, e deseñar, desenvolver, implementar, verificar e documentar solucións sóftware sobre a base dun coñecemento adecuado das teorías, modelos e técnicas actuais.
B1	Capacidade de resolución de problemas
B3	Capacidade de análise e síntese
C2	Dominar a expresión e a comprensión de forma oral e escrita dun idioma estranxeiro.
C3	Utilizar as ferramentas básicas das tecnoloxías da información e as comunicacións (TIC) necesarias para o exercicio da súa profesión e para a aprendizaxe ao longo da súa vida.
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.
C8	Valorar a importancia que ten a investigación, a innovación e o desenvolvemento tecnolóxico no avance socioeconómico e cultural da sociedade.

Resultados da aprendizaxe			
Resultados de aprendizaxe	Competencias / Resultados do título		
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.	A28	B1 B3	C2 C3 C6 C7 C8

Contidos
----------



Temas	Subtemas
Bloque I: Validación de software	I.1 Especificación, deseño e execución de probas I.1.1. Niveis e tipos de probas I.1.2. Propiedades e trazabilidade de requerimentos I.2 Xestión de probas: planificación, avaliación, métricas e revisións
Bloque II: Métodos formais e razonamento automático	II.1 Introducción: Dedución natural e cálculo de secuentes II.2 Proba automática utilizando PVS II.3 Que é e para que se utiliza un probador de teoremas? II.4 Linguaxe de especificación de PVS: Tipos, expresións, teorías, subtipado II.5 Probador PVS: tácticas, recursión, razonamento ecuacional
Bloque III: Comprobación por modelos (model checking)	III.1 Introducción ás lóxicas modais temporais III.2 Especificación de propiedades: deadlocks, safety, liveness, fairness III.3 Funcionamento dun comprobador por modelos III.4 Introducción ao manexo dunha ferramenta de model checking

Planificación				
Metodoloxías / probas	Competencias / Resultados	Horas lectivas (presenciais e virtuais)	Horas traballo autónomo	Horas totais
Sesión maxistral	B3 C2 C7 C8	21	26.25	47.25
Prácticas de laboratorio	A28 B1 B3 C2 C3 C6	14	35	49
Traballos tutelados	A28 B1 B3 C2 C3 C6	7	7	14
Proba obxectiva	B1 B3 C6	3	31.5	34.5
Atención personalizada		5.25	0	5.25

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

Metodoloxías	
Metodoloxías	Descrición
Sesión maxistral	Clases de teoría onde se imparten os contidos do temario.
Prácticas de laboratorio	Elaboración de traballos prácticos no laboratorio.
Traballos tutelados	Resolución de traballos tutelados prantexados e resoltos en horario de titorías de grupos reducidos.
Proba obxectiva	Exame escrito.

Atención personalizada	
Metodoloxías	Descrición
Sesión maxistral Prácticas de laboratorio Traballos tutelados Proba obxectiva	Resolución de dúbidas de teoría e/ou prácticas, traballos tutelados, etc. en horario de titorías de cada profesor/a.

Avaliación			
Metodoloxías	Competencias / Resultados	Descrición	Cualificación
Prácticas de laboratorio	A28 B1 B3 C2 C3 C6	Entrega e defensa de traballos prácticos de laboratorio. Computa ata un máximo de 4 puntos da nota final. A súa realización non é obrigatoria para superar a asignatura.	40



Traballos tutelados	A28 B1 B3 C2 C3 C6	Resolución e participación en traballos tutelados en horario de titorías de grupos reducidos. Computa un máximo de 2 puntos da nota final. A súa realización non é obrigatoria para superar a asignatura.	20
Proba obxectiva	B1 B3 C6	Exame escrito que computa un máximo de 4 puntos sobre a nota final. Require un mínimo de 2 puntos para superar a asignatura.	40

### Observacións avaliación

En caso de non acadar o mínimo na proba obxectiva, a nota final será a obtida na proba obxectiva.

Na convocatoria de segunda oportunidade, a proba obxectiva poderá ir acompañada dunha actividade de avaliación das prácticas de laboratorio. Dacordo coa normativa da UDC en relación ao estudantado matriculado a tempo parcial, o réxime de asistencia a clase non afectará negativamente ao proceso de avaliación, admitíndose nesta asignatura a dispensa académica para a asistencia solicitada polas canles institucionais habilitadas ao efecto. Porén, esta flexibilidade asistencial non eximirá da entrega de traballos tutelados e prácticas nos mesmos prazos fixados para o estudantado a tempo completo.

### Fontes de información

<b>Bibliografía básica</b>	<ul style="list-style-type: none"><li>- Mordechai Ben-Ari (2012). Mathematical Logic for Computer Science. Springer</li><li>- Ron Patton (2001). Software testing. Sams</li><li>- Peter Farrell-Vinay (2008). Manage software testing. Auerbach</li><li>- Kent Beck (2002). Test Driven Development (By Example). Addison-Wesley</li><li>- Gerard J. Holzmann (2003). The SPIN model checker: primer and reference manual. Addison-Wesley</li><li>- Mordechai Ben-Ari (2001). Mathematical Logic for Computer Science. Springer</li><li>- Zohar Manna and Amir Pnueli (1991). The Temporal Logic of Reactive and Concurrent Systems. Specification. Springer</li><li>- Zohar Manna and Amir Pnueli (1995). The Temporal Logic of Reactive and Concurrent Systems. Safety. Springer</li></ul>
----------------------------	--

### Bibliografía complementaria

### Recomendacións

#### Materias que se recomenda ter cursado previamente

Deseño Software/614G01015  
Concurrencia e Paralelismo/614G01018  
Proceso Software/614G01019  
Arquitectura do Software/614G01221  
Enxeñaría de Requisitos/614G01222  
Aseguramento da Calidade/614G01223

#### Materias que se recomenda cursar simultaneamente

Representación do Coñecemento e Razoamento Automático/614G01036  
Teoría da computación/614G01039  
Metodoloxías de Desenvolvemento/614G01051

#### Materias que continúan o temario

Proxectos de Desenvolvemento Software/614G01226

### 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