



| Guía Docente | | | | |
|-----------------------|---|--|---|-----------|
| Datos Identificativos | | | | 2014/15 |
| 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 | | | |
| Prerrequisitos | | | | |
| Departamento | Computación | | | |
| Coordinación | Castro Souto, Laura Milagros | Correo electrónico | laura.milagros.castro.souto@udc.es | |
| Profesorado | Cabalar Fernandez, Jose Pedro Castro Souto, Laura Milagros Perez Vega, Gilberto | Correo electrónico | pedro.cabalar@udc.es laura.milagros.castro.souto@udc.es gilberto.pvega@udc.es | |
| Web | moodle.udc.es | | | |
| Descrición xeral | <p>Esta asignatura busca dominar as alternativas actuais da enxeñaría de software para a validación e verificación do software, mediante o:</p> <ul style="list-style-type: none"> - coñecemento de técnicas e ferramentas de validación de software funcionais e non funcionais a todos os niveles (unidade, integración, sistema); - coñecemento de técnicas e ferramentas de razonamiento automático; e - o coñecemento de técnicas e ferramentas de verificación formal. | | | |

| Competencias da titulación | |
|----------------------------|--|
| Código | Competencias da titulación |
| A28 | Capacidade de identificar e analizar problemas, e deseñar, desenvolver, implementar, verificar e documentar solucións s'oftware sobre a base dun coñecemento adecuado das teorías, modelos e técnicas actuais. |
| B1 | Capacidade de resolución de problemas |
| B2 | Traballo en equipo |
| B3 | Capacidade de análise e síntese |
| B4 | Capacidade para organizar e planificar |
| 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. |
| C4 | Desenvolverse para o exercicio dunha cidadanía aberta, culta, crítica, comprometida, democrática e solidaria, capaz de analizar a realidade, diagnosticar problemas, formular e implantar solucións baseadas no coñecemento e orientadas ao ben común. |
| C6 | Valorar criticamente o coñecemento, a tecnoloxía e a información dispoñible para resolver os problemas cos que deben afrontarse. |
| 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 | |
|---|----------------------------|
| Competencias de materia (Resultados de aprendizaxe) | Competencias da titulación |
| | |



| | | | |
|--|-----|----------------------|----------------------------------|
| 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 B2 B3 B4 | C2 C3 C4 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 | Horas presenciais | Horas non presenciais / traballo autónomo | Horas totais |
| Sesión maxistral | 21 | 26.25 | 47.25 |
| Prácticas de laboratorio | 14 | 35 | 49 |
| Traballos tutelados | 7 | 7 | 14 |
| Proba obxectiva | 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. |



Avaliación

| Metodoloxías | Descrición | Cualificación |
|--------------------------|---|---------------|
| Prácticas de laboratorio | 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 | 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 | 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

| |
|--|
| Na convocatoria de segunda oportunidade, a proba obxectiva irá acompañada dunha actividade de avaliación das prácticas de laboratorio. |
|--|

Fontes de información

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

Recomendacións

Materias que se recomenda ter cursado previamente

Proxectos de Desenvolvemento Software/614G01226

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

Deseño Software/614G01015

Concorrencia e Paralelismo/614G01018

Proceso Software/614G01019

Arquitectura do Software/614G01221

Enxeñaría de Requisitos/614G01222

Aseguramento da Calidade/614G01223

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