



Universitat de Lleida

GUÍA DOCENTE
**SISTEMAS DE VALIDACIÓN DE
HARDWARE Y SOFTWARE**

Coordinación: BEJAR TORRES, RAMON

Año académico 2022-23

Información general de la asignatura

Denominación	SISTEMAS DE VALIDACIÓN DE HARDWARE Y SOFTWARE			
Código	102044			
Semestre de impartición	2o Q(SEMESTRE) EVALUACIÓN CONTINUADA			
Carácter	Grado/Máster	Curso	Carácter	Modalidad
	Grado en Ingeniería Informática	4	OBLIGATORIA	Presencial
	Grado en Ingeniería Informática	4	OPTATIVA	Presencial
Número de créditos de la asignatura (ECTS)	6			
Tipo de actividad, créditos y grupos	Tipo de actividad	PRALAB	TEORIA	
	Número de créditos	3	3	
	Número de grupos	1	1	
Coordinación	BEJAR TORRES, RAMON			
Departamento/s	INFORMATICA E INGENIERIA INDUSTRIAL			
Distribución carga docente entre la clase presencial y el trabajo autónomo del estudiante	6 ECTS = 25x6 = 150 - 60 horas de actividades presenciales - 90 horas de actividades no presenciales			
Información importante sobre tratamiento de datos	Consulte este enlace para obtener más información.			
Idioma/es de impartición	Catalan Todo el material de aprendizaje y ejercicios están desarrollados en inglés			
Distribución de créditos	3 créditos teoría 3 créditos práctica			

Profesor/a (es/as)	Dirección electrónica\nprofesor/a (es/as)	Créditos impartidos por el profesorado	Horario de tutoría/lugar
BEJAR TORRES, RAMON	ramon.bejar@udl.cat	6	

Información complementaria de la asignatura

Para poder abordar con éxito la asignatura, es recomendable haber cursado previamente asignaturas con contenidos básicos sobre:

- Lógica computacional (obligatoria en este grado de Informática).
- Inteligencia artificial (obligatoria en este grado de Informática).

Objetivos académicos de la asignatura

Resultados esperados del aprendizaje ligados a competencias estratégicas y transversales:

- Sabe preparar documentos técnicos con diferentes herramientas de presentación para documentos digitales (CT3).
- Sabe trabajar con documentación técnica y científica escrita en inglés (CT2).
- Comprende los principales problemas que encuentran en el diseño de herramientas automáticas para la verificación de programas y sabe analizar las propiedades formales verificar por la verificación completa o parcial de un algoritmo (EPS6).

Resultados esperados del aprendizaje ligados a competencias específicas:

- Conoce y comprende los problemas fundamentales sobre la especificación formal del comportamiento de programas (GII-C5).
- Conoce la especificación formal de programas basada en lenguajes formales (GII-C5).
- Sabe utilizar herramientas fundamentales para la verificación semi-automática de software, que necesitan de la interacción con personas para finalizar pruebas que no se pueden automatizar al 100%, y comprende las limitaciones que los resultados teóricos sobre computabilidad imponen sobre estas herramientas (GII- C5, GII-C6).

Competencias

- Estratégicas:

CT2. Adquirir un dominio significativo de una lengua extranjera, especialmente del inglés.

CT3. Adquirir capacitación en el uso de las nuevas tecnologías y de las tecnologías de la información y la comunicación.

- Transversales:

EPS6. Capacidad de análisis y síntesis.

- Específicas:

GII-C5. Capacidad para adquirir, obtener, formalizar y representar el conocimiento humano en una forma computable para la resolución de problemas mediante un sistema informático en cualquier ámbito de aplicación, particularmente los relacionados con aspectos de computación, percepción y actuación en ambientes o entornos inteligentes.

GII-C6. Capacidad para desarrollar y evaluar sistemas interactivos y de presentación de información compleja y su aplicación a la resolución de problemas de diseño de interacción persona computadora.

Contenidos fundamentales de la asignatura

1. Introducción a la verificación de software y hardware
2. Verificación formal de algoritmos
3. Verificación completa
 - Hoare Logic Calculus
 - Verificación de programas While-Do
 - Verificación parcial/completa
 - Forward Verification con actualización de estados
 - Verificación de programas con la herramienta Key-Hoare
4. Verificación parcial y búsqueda de errores
 - Verificación de sistemas mediante Bounded Model Checking (BMC)
 - Verificación de programas ANSI-C mediante la herramienta CBMC
 - Verificación de Hardware mediante BMC

Ejes metodológicos de la asignatura

Habrán tres tipos de actividades:

- 1) Clases magistrales on-line con videos y con la herramienta de video-conferencia del campus virtual.
- 2) Clases de problemas y de laboratorio.
- 3) Trabajo autónomo fuera de clase para finalizar ejercicios y llevar a cabo los trabajos obligatorios a realizar.

Plan de desarrollo de la asignatura

Week	Description	Classroom/Virtual Campus Activity	Autonomous Activity	Hours (F and A)
1	Presentation and introduction to formal software verification	Lectures		4 -

2	Formal specification and Symbolic Execution	Lectures and problem solving laboratory	Solve Exercises	4 6
3	Formal specification and Symbolic Execution	Lectures and problem solving laboratory	Study Solve Exercises	4 7
4	Hoare Logic with states – assignments and conditionals	Lectures and problem solving laboratory	Solve Exercises	4 6
5	Hoare Logic with states - loops	Lectures and problem solving laboratory	Study Solve Exercises	4 7
6	Hoare Logic with states - loops	Lectures and problem solving laboratory	Solve Exercises	4 6
7	Hoare Logic with states – loop termination	Lectures and problem solving laboratory	Work on 1st problem set assignment Solve Exercises	4 8
8	Hoare Logic with states – loop termination	Lectures and problem solving laboratory	Study Work on 1st problem set assignment	4 8
9	Evaluation	Written Exam on complete verification with Hoare Logic with states	Study Work on 1st problem set assignment	4 8
10	Introduction to BMC for digital software and hardware systems	Lectures and programming laboratory	Solve Exercises	4 6
11	Static Analysis in CBMC and formula building	Lectures and programming laboratory	Solve Exercises	4 6
12	The CBMC verification tool	Lectures and programming laboratory	Study Solve Exercises	4 6
13	CBMC user assertions	Lectures and programming laboratory	Study Solve Exercises	4 10
14	CBMC automatic assertions and code coverage	Lectures and programming laboratory	Study Work on 2nd problem set assignment	4 8
15	Verification of Verilog Hardware Designs	Lectures and programming laboratory	Study Work on 2nd problem set assignment	4 8
16			Study Work on 2nd problem set assignment	- 6
17	Evaluation	Written Exam on BMC and CBMC	Study Work on 2nd problem set assignment	2 6
18				
19				

Sistema de evaluación

Tabla. Actividades de evaluación

Acr.	Actividad de Evaluación	Peso	Nota Mínima	En grupo	Obligatoria
------	-------------------------	------	-------------	----------	-------------

P1	Problemas de verificación formal	24%	NO	SÍ	NO
P2	Problemas de verificación formal	21%	NO	SÍ	NO
PR	Ejercicios en clase	5%	NO	NO	NO
E1	Examen escrito (1)	28%	NO	NO	NO
E2	Examen escrito (1)	27%	NO	NO	NO

Nota final = $0,24 \cdot P1 + 0,21 \cdot P2 + 0,05 \cdot PR + 0,28 \cdot E1 + 0,27 \cdot E2$

(1): Los exámenes se pueden recuperar.

Bibliografía y recursos de información

Todo el material de aprendizaje será proporcionado durante el curso en forma de transparencias, apuntes y manuales de los diferentes programas que se utilizarán. No hay ningún libro de texto adecuado para seguir los contenidos de esta asignatura.