



Guía docente				
Datos Identificativos				2015/16
Asignatura (*)	Arquitectura e Ingeniería de Computadores	Código	614111401	
Titulación	Enxeñeiro en Informática			
Descriptorios				
Ciclo	Periodo	Curso	Tipo	Créditos
1º y 2º Ciclo	Anual	Cuarto	Troncal	9
Idioma	CastellanoGallego			
Modalidad docente	Presencial			
Prerrequisitos				
Departamento	Electrónica e Sistemas			
Coordinador/a	Doallo Biempica, Ramon	Correo electrónico	ramon.doallo@udc.es	
Profesorado	Doallo Biempica, Ramon	Correo electrónico	ramon.doallo@udc.es	
Web	campusvirtual.udc.es/moodle/			
Descripción general	- Arquitecturas microprocesador (microprocesadores comerciales, multithreading, multicore, streaming) - Arquitecturas multiprocesador			

Competencias / Resultados del título	
Código	Competencias / Resultados del título
A1	Aprender de manera autónoma nuevos conocimientos y técnicas avanzadas adecuadas para la investigación, el diseño y el desarrollo de sistemas y servicios informáticos.
A2	Concebir y desarrollar nuevas arquitecturas de computación, en especial para sistemas multiprocesadores, analizando y adaptando diversas alternativas tecnológicas a cada problema concreto.
A3	Concebir y planificar el desarrollo de aplicaciones informáticas complejas o con requisitos especiales.
A4	Conocer y aplicar diferentes protocolos de comunicación y sistemas de gestión de red.
A6	Evaluar, definir, seleccionar y auditar plataformas hardware y software para la ejecución y desarrollo de aplicaciones y servicios informáticos.
A8	Concebir, desplegar, organizar y gestionar un servicio informático complejo.
B2	Resolver problemas de forma efectiva.
B3	Aplicar un pensamiento crítico, lógico y creativo.
B5	Trabajar de forma colaborativa.
B8	Trabajar en equipos de carácter interdisciplinar.
B9	Capacidad para tomar decisiones.
B11	Razonamiento crítico.
B12	Capacidad para el análisis y la síntesis.
B13	Capacidad de comunicación.
B15	Motivación por la calidad.
C6	Valorar críticamente el conocimiento, la tecnología y la información disponible para resolver los problemas con los que deben enfrentarse.

Resultados de aprendizaje			
Resultados de aprendizaje		Competencias / Resultados del título	
Utilizar nuevas arquitecturas de computación, tanto microprocesador como multiprocesador, en soluciones en ingeniería informática.	A1	B2	C6
	A2		
	A3		
	A6		
	A8		
Configurar sistemas para servidores y sistemas de alto rendimiento.	A2	B3	
	A3		
	A8		



<p>Analizar de forma crítica parámetros de rendimiento de sistemas hardware y aplicaciones específicas con necesidades especiales.</p>	<p>A2 A3 A4 A6 A8</p>	<p>B9 B11 B12</p>	<p>C6</p>
<p>Desarrollar aplicaciones paralelas en equipo siguiendo el método de ingeniería informática.</p>	<p>A2 A6</p>	<p>B2 B5 B8 B11 B12 B13 B15</p>	<p>C6</p>

Contenidos	
Tema	Subtema
<p>1. Introducción al procesamiento paralelo</p>	<p>1.1. Procesamiento paralelo 1.2. Condiciones de paralelismo - Concepto de dependencia - Condiciones de Bernstein - Paralelismo hardware y software 1.3. Niveles de paralelismo 1.4. Importancia del procesamiento paralelo 1.5. Clasificaciones de arquitecturas paralelas - Taxonomía de Flynn - Organización del sistema memoria: clasificación tradicional de sistemas MIMD. 1.6. Medidas de rendimiento</p>
<p>2. Paralelismo a nivel de instrucción</p>	<p>2.1. Rendimiento de un procesador 2.2. Soluciones básicas para la mejora del rendimiento: Evolución 2.3. Técnicas para el aumento del paralelismo a nivel de instrucción 2.4. Planificación estática/software - Planificación estática básica - Desenrollamiento de bucles - Planificación estática superescalar - Planificación estática VLIW - Segmentación software - Detección y eliminación de dependencias 2.5. Planificación dinámica/hardware - Planificación dinámica superescalar 2.6. Renombre de registros - Renombre de registros software - Renombre de registros hardware: Buffer de reordenamiento 2.7. Tratamiento de operaciones load/store: buffer de almacenamiento 2.8. Tratamiento de riesgos de control - Predicción de salto estática, salto retardado - Predicción dinámica de saltos 2.9. Especulación 2.10. Ejemplos comerciales</p>



<p>3. Procesamiento vectorial</p>	<p>3.1. ¿Por qué máquinas vectoriales?</p> <p>3.2. Arquitecturas vectoriales</p> <p>3.3. Arquitectura vectorial básica</p> <ul style="list-style-type: none">- Componentes principales de la U.V.- Máquinas vectoriales comerciales- Repertorio de instrucciones de la U.V.- Ejemplo de bucle vectorial <p>3.4. Vectorización</p> <p>3.5. Tiempo de ejecución vectorial</p> <p>3.6. Organización de memoria en un procesador vectorial</p> <ul style="list-style-type: none">- Entrelazamiento de memoria- Técnicas de estructuración de datos para máquinas vectoriales <p>3.7. Control de la longitud del vector</p> <p>3.8. Rendimiento vectorial</p> <ul style="list-style-type: none">- Efectos de la vectorización.- Mejorando el rendimiento vectorial:<ul style="list-style-type: none">? Encadenamiento de U.F. vectoriales? Transformación de código para evitar conflictos de acceso a memoria <p>3.9. Ejemplos comerciales:</p> <ul style="list-style-type: none">- Earth Simulator- Cray X1
<p>4. Computación paralela: arquitecturas y paradigmas de programación</p>	<p>4.1. Introducción</p> <p>4.2. Arquitecturas paralelas modernas:</p> <ul style="list-style-type: none">* Arquitecturas paralelas vectorial, multimedia y GPU* Arquitecturas paralelas TLP: multiprocesador y multicomputador* Arquitecturas paralelas RLP: cluster y WSC* Arquitecturas paralelas híbridas* Comparación entre arquitecturas paralelas modernas <p>4.4. Paradigmas de programación</p> <ul style="list-style-type: none">* Paradigma de memoria compartida, memoria distribuida, PGAS, streaming y MapReduce.* Paradigmas híbridos* Comparación entre herramientas disponibles en el mercado* Comparación entre soluciones tecnológicas disponibles en el mercado
<p>5. Arquitecturas multiprocesador: coherencia caché y consistencia de memoria</p>	<p>5.1. Introducción</p> <p>5.1. Sincronización</p> <p>5.2. Modelos de consistencia de memoria</p> <p>5.2. Protocolos de coherencia caché</p> <ul style="list-style-type: none">* Protocolo de snooping* Protocolo basado en directorio
<p>6. Redes de interconexión para arquitecturas paralelas</p>	<p>6.1. Introducción</p> <p>6.2. Redes de interconexión estáticas</p> <p>6.3. Estrategias de conmutación</p> <p>6.4. Redes de interconexión dinámicas</p> <p>6.5. Funciones de encaminamiento de datos</p> <p>6.6. On-chip networks (OCN)</p>



Metodologías / pruebas	Competencias / Resultados	Horas lectivas (presenciales y virtuales)	Horas trabajo autónomo	Horas totales
Prueba objetiva	A1 A2 A3 A4 A6 A8 B2 B3 B5 B8 B9 B11 B12 B13 B15 C6	4	0	4
Atención personalizada		3	0	3

(*)Los datos que aparecen en la tabla de planificación són de carácter orientativo, considerando la heterogeneidad de los alumnos

Metodologías	
Metodologías	Descripción
Prueba objetiva	Prueba sobre los conceptos teóricos presentados en las sesiones magistrales.

Atención personalizada	
Metodologías	Descripción
Prueba objetiva	Prueba objetiva: - Atención personal en horas de tutoría para aclarar dudas sobre los contenidos teórico/prácticos de la materia. - Revisión de los ejercicios realizados.

Evaluación			
Metodologías	Competencias / Resultados	Descripción	Calificación
Prueba objetiva	A1 A2 A3 A4 A6 A8 B2 B3 B5 B8 B9 B11 B12 B13 B15 C6	Prueba sobre los contenidos trabajados en las sesiones magistrales.	100
Otros			

Observaciones evaluación
- La evaluación de los alumnos consistirá en una prueba escrita en las fechas indicadas en el calendario académico oficial. - En los exámenes de todas las convocatorias se aplicará el mismo criterio en cuanto a que el alumno debe obtener al menos la mitad de la nota para superar la asignatura.

Fuentes de información	
Básica	- Ortega, J., Anguita, M. y Prieto, A. (2005). Arquitectura de Computadores. Thomson - Hennessy, J. L. y Patterson, D. A. (2012). Computer Architecture: A Quantitative Approach (5th edition). Morgan Kaufmann - Patterson, D. A. y Hennessy, J. L. (2000). Estructura y Diseño de Computadores. Reverté - Stallings, W. (2006). Organización y arquitectura de computadores. Reverté
Complementaria	- Sima, D. , Fountain,T. y Kacsuk, P. (1997). Advanced Computer Architecture. Addison-Wesley - Culler, D. E. y Singh,J. P. (1999). Parallel Computer Architecture: a Hardware/Software Approach. Morgan Kaufmann - Hwang, K. y Xu, Z. (1998). Scalable Parallel Computing. McGraw-Hill - Tannebaum, A. S. (1999). Structured Computer Organization. Prentice Hall

Recomendaciones
Asignaturas que se recomienda haber cursado previamente



Estructura de Computadores I/614111208
Sistemas Operativos II/614111302
Estructura de Computadores II/614111306
Redes de Comunicaciones/614111307

Asignaturas que se recomienda cursar simultáneamente

Arquitecturas Tolerantes a Fallos/614111605
Optimización del Procesamiento Paralelo/614111630

Asignaturas que continúan el temario

Arquitecturas Tolerantes a Fallos/614111605
Optimización del Procesamiento Paralelo/614111630

Otros comentarios

(*) La Guía Docente es el documento donde se visualiza la propuesta académica de la UDC. Este documento es público y no se puede modificar, salvo cosas excepcionales bajo la revisión del órgano competente de acuerdo a la normativa vigente que establece el proceso de elaboración de guías