



Guía Docente

Datos Identificativos					2013/14
Asignatura (*)	Arquitectura e Enxeñaría de Computadores		Código	614111401	
Titulación					
Descritores					
Ciclo	Período	Curso	Tipo	Créditos	
1º e 2º Ciclo	Anual	Cuarto	Troncal	9	
Idioma	CastelánGalego				
Prerrequisitos					
Departamento	Electrónica e Sistemas				
Coordinación	Doallo Biempica, Ramon	Correo electrónico	ramon.doallo@udc.es		
Profesorado	Doallo Biempica, Ramon	Correo electrónico	ramon.doallo@udc.es		
Web	campusvirtual.udc.es/moodle/				
Descrición xeral	- Arquitecturas microprocesador (microprocesadores comerciais, multithreading, multicore, streaming) - Arquitecturas multiprocesador				

Competencias da titulación

Código	Competencias da titulación

Resultados da aprendizaxe

Competencias de materia (Resultados de aprendizaxe)	Competencias da titulación		
Utilizar novas arquitecturas de computación, tanto microprocesador como multiprocesador, en solucións en enxeñaría informática.	A1 A2 A3 A6 A8	B2	C6
Configurar sistemas para servidores e sistemas de alto rendemento.	A2 A3 A8	B3	
Analizar de forma crítica parámetros de rendemento de sistemas hardware e aplicacións específicas con necesidades especiais.	A2 A3 A4 A6 A8	B9 B11 B12	C6
Desarrollar aplicacións paralelas en equipo seguindo o método de enxeñaría informática.	A2 A6	B2 B5 B8 B11 B12 B13 B15	C6

Contidos

Temas	Subtemas



1. Introducción al procesamiento paralelo	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
2. Paralelismo a nivel de instrucción	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>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>



Metodoloxías / probas	Horas presenciais	Horas non presenciais / traballo autónomo	Horas totais
Proba obxectiva	4	0	4
Atención personalizada	3	0	3

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

Metodoloxías	
Metodoloxías	Descrición
Proba obxectiva	Prueba sobre los conceptos teóricos presentados en las sesiones magistrales.

Atención personalizada	
Metodoloxías	Descrición
Proba obxectiva	<p>Prueba objetiva:</p> <ul style="list-style-type: none"> - Atención personal en horas de tutoría para aclarar dudas sobre los contenidos teórico/prácticos de la materia. - Revisión de los ejercicios realizados.

Avaliación		
Metodoloxías	Descrición	Cualificación
Proba obxectiva	Prueba sobre los contenidos trabajados en las sesiones magistrales.	100
Outros		

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

Fontes de información	
Bibliografía básica	<ul style="list-style-type: none"> - 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
Bibliografía complementaria	<ul style="list-style-type: none"> - 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

Recomendacións
Materias que se recomenda ter cursado previamente
Arquitecturas Tolerantes a Fallos/614111605
Optimización do Procesamento Paralelo/614111630
Materias que se recomenda cursar simultaneamente



Arquitecturas Tolerantes a Fallos/614111605

Optimización do Procesamento Paralelo/614111630

Materias que continúan o temario

Estrutura de Computadores I/614111208

Sistemas Operativos II/614111302

Estrutura de Computadores II/614111306

Redes de Comunicacións/614111307

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