Guía Docente					
Datos Identificativos					2012/13
Asignatura (*)	Arquitectura e Enxeñaría de Computadores			Código	614111401
Titulación	Enxeñeiro en Informática				
Descriptores					
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	Arenaz Silva, Manuel Carlos Doallo Biempica, Ramon		Correo electrónico	manuel.arenaz@udc.es ramon.doallo@udc.es	
Web	campusvirtual.udc.es/moodle/				
Descrición xeral	- Arquitecturas microprocesador (microprocesadores comerciales, multithreading, multicore, streaming) - Arquitecturas multiprocesador				

Competencias da titulación	
Código	Competencias da titulación
A1	Aprender de maneira autónoma novos coñecementos e técnicas avanzadas axeitadas para a investigación, o deseño e o desenvolvemento de sistemas e servizos informáticos.
A2	Concibir e desenvolver novas arquitecturas de computación, en especial para sistemas multiprocesadores, analizando e adaptando diversas alternativas tecnolóxicas a cada problema concreto.
A3	Concibir e planificar o desenvolvemento de aplicacións informáticas complexas ou con requisitos especiais.
A4	Coñecer e aplicar diferentes protocolos de comunicación e sistemas de xestión de rede.
A6	Avaliar, definir, seleccionar e auditar plataformas hardware e software para a execución e desenvolvemento de aplicacións e servizos informáticos.
A8	Concibir, despregar, organizar e xestionar un servizo informático complexo.
B2	Resolver problemas de forma efectiva.
B3	Aplicar un pensamento crítico, lóxico e creativo.
B5	Traballar de forma colaborativa.
B8	Traballar en equipos de carácter interdisciplinar.
B9	Capacidade para tomar decisións.
B11	Razoamento crítico.
B12	Capacidade para a análise e a síntese.
B13	Capacidade de comunicación.
B15	Motivación pola calidade.
C6	Valorar criticamente o coñecemento, a tecnoloxía e a información dispoñible para resolver os problemas cos que deben enfrontarse.

Resultados da aprendizaxe			
Competencias de materia (Resultados de aprendizaxe)	Competencias da titulación		
Utilizar nuevas arquitecturas de computación, tanto microprocesador como multiprocesador, en soluciones en ingeniería informática.	A1 A2 A3 A6 A8	B2	C6
Configurar sistemas para servidores y sistemas de alto rendimiento.	A2 A3 A8	B3	

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

Contidos	
Temas	**Subtemas**
4. Computación paralela: arquitecturas y paradigmas de programación	4.1. Clasificación de arquitecturas MIMD - Organización del sistema memoria: clasificación tradicional de sistemas MIMD ? Comparación multiprocesador-multicomputador - Escalabilidad: clasificación de multiprocesadores basada en la latencia de acceso a memoria ? Clasificación de sistemas NUMA 4.2. Otras denominaciones usuales 4.3. Lista Top500 4.4. Modelos de programación paralela
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
5. Arquitecturas multiprocesador: coherencia caché y consistencia de memoria	5.1. Introducción 5.2. Protocolos de coherencia caché 5.3. Protocolo de snooping - Ejemplo de protocolo de snooping 5.4. Protocolo basado en directorio - Ejemplo de protocolo basado en directorio 5.5. Modelos de consistencia de memoria
3. Procesamiento vectorial	3.1. ¿Por qué máquinas vectoriales? 3.2. Arquitecturas vectoriales 3.3. Arquitectura vectorial básica - Componentes principales de la U.V. - Máquinas vectoriales comerciales - Repertorio de instrucciones de la U.V. - Ejemplo de bucle vectorial 3.4. Vectorización 3.5. Tiempo de ejecución vectorial 3.6. Organización de memoria en un procesador vectorial - Entrelazamiento de memoria - Técnicas de estructuración de datos para máquinas vectoriales 3.7. Control de la longitud del vector 3.8. Rendimiento vectorial - Efectos de la vectorización. - Mejorando el rendimiento vectorial: ? Encadenamiento de U.F. vectoriales ? Transformación de código para evitar conflictos de acceso a memoria 3.9. Ejemplos comerciales: - Earth Simulator - Cray X1

6. Redes de interconexión para arquitecturas paralelas	6.1. Introducción 6.2. Redes de interconexión estáticas 6.3. Estrategias de conmutación 6.4. Redes de interconexión dinámicas 6.5. Funciones de encaminamiento de datos 6.6. On-chip networks (OCN)
7. Cluster computing	7.1. Configuración de un clúster - Introducción a las arquitecturas clúster - Elementos de un clúster ? Administrador, S.O. ? Nodos de cómputo ? Almacenamiento ? Redes de interconexión de clusters 7.2. Administración de un clúster - Software de administración y monitorización de clusters - Gestión de recursos distribuidos - Balanceadores de carga: linux virtual server - Clusters en aplicaciones científicas - Clusters en aplicaciones empresariales

Planificación			
Metodoloxías / probas	Horas presenciais	Horas non presenciais / traballo autónomo	Horas totais
Prácticas de laboratorio	39	39	78
Sesión maxistral	52	78	130
Proba obxectiva	4	0	4
Seminario	5	5	10
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
Prácticas de laboratorio	Realización de prácticas de arquitecturas microprocesador (extensiones SSE2) e introducción a la programación paralela con PVM (Parallel Virtual Machine) y OpenMP.
Sesión maxistral	Exposición oral complementada con el uso de las TIC, acompañada de preguntas dirigidas a los estudiantes para afianzar la discusión y conocimientos con el fin de mejorar el aprendizaje. Puede haber debates sobre cuestiones del ámbito de trabajo.
Proba obxectiva	Prueba sobre los conceptos teóricos presentados en las sesiones magistrales.
Seminario	Seminarios orientado al trabajo práctico de la asignatura, como introducción a la programación paralela con PVM, e introducción a la programación con SSE2.

Atención personalizada	
Metodoloxías	Descrición

Prácticas de laboratorio Proba obxectiva	Prácticas de laboratorio: Análisis con el estudiante del trabajo que está realizando, y la defensa de su práctica una vez ha terminado. Sesión maxistral: Atención personal en horas de tutoría para aclarar dudas surgidas en el seguimiento durante el curso de la materia. Prueba objetiva: Revisión de los ejercicios realizados.

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

Observacións avaliación
*Consideraciones generales: - La evaluación de los alumnos consistirá enuna prueba escrita al final del curso. Es necesario entregar y aprobarlas prácticas para superar la asignatura. La evaluación de las prácticasse realiza a lo largo del curso (en evaluación continua). - Laevaluación del profesorado se realizará a través de las encuestas deevaluación docente, a cubrir por todos los alumnos de la asignatura.*Aspectos y criterios de evaluación: Se realizará un examen parcial conla materia explicada durante el primer cuatrimestre: los alumnos queaprueben este parcial, realizarán en el examen final de junio sólo laparte del examen correspondiente al segundo cuatrimestre; los alumnosque no aprueben el examen del primer cuatrimestre, realizarán en elexamen final de junio las dos partes correspondientes a amboscuatrimestres debiendo obtener la mitad de la nota correspondiente a lacalificación total de cada cuatrimestre para poder obtener el aprobado.Con respecto a los exámenes de septiembre y diciembre se aplicará elmismo criterio en cuanto a que el alumno debe obtener al menos la mitadde la nota correspondiente a la calificación total de cada cuatrimestre.También es condición necesaria para aprobar tener una calificaciónmínima de 5 sobre 10 en la valoración de las prácticas. El aprobado enlas prácticas se conserva para cursos posteriores siempre y cuando semantengan las prácticas. Para el curso 2011/2012 se mantienen lasprácticas del curso 2010/2011. El aprobado en el examen de teoría en lasconvocatorias de Junio o Septiembre se guarda como máximo hasta laconvocatoria extraordinaria de diciembre del mismo año.*Pautas para la mejora y la recuperación: - Pruebas objetivas enSeptiembre y Diciembre. Si las prácticas no han sido superadas a lolargo del curso podría realizarse un examen de prácticas en el queademás se entregarían y defenderían todas las prácticas de la asignatura.

Fontes de información	
Bibliografía básica	- Ortega, J., Anguita, M. y Prieto, A. (2005). Arquitectura de Computadores. Thomson - Hennessy, J. L. y Patterson, D. A. (1996). Computer Architecture: A Quantitative Approach. 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	- 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