



Ficha del curso: 2017-2018

|  |  |   |               |
|--|--|---|---------------|
| <b>Grado:</b> GRADO EN INGENIERÍA INFORMÁTICA  |  | <b>Curso:</b> 4º ( 1C )                   |               |
| <b>Asignatura:</b> 803289 - Arquitectura de Computadores                                   |  | <b>Abrev:</b> AC                          | <b>6 ECTS</b> |
| <b>Asignatura en Inglés:</b> Computer Architecture   |  | <b>Carácter:</b> Obligatoria              |               |
| <b>Materia:</b> Tecnología y Arquitectura de Computadores                                  |  | <b>12 ECTS</b>                            |               |
| <b>Otras asignaturas en la misma materia:</b><br>Tecnología y Organización de Computadores |  | <b>6 ECTS</b>                             |               |
| <b>Módulo:</b> Complementario  |  |   |               |
| <b>Departamento:</b> Arquitectura de Computadores y Automática                             |  | <b>Coordinador:</b> Hermida Correa, Román |               |

**Descripción de contenidos mínimos:**

Paralelismo a nivel de instrucción y a nivel thread.  
Introducción a los multiprocesadores: Problemas de sincronización.  
E/S y sistemas de almacenamiento.

**Programa detallado:**

Módulo 1. Introducción y tendencias en arquitectura de computadores.

- Contexto de la asignatura
- Evolución tecnológica, binomio arquitectura-tecnología
- Consumo de energía
- Factores determinantes del coste
- Medidas de rendimiento

Módulo 2. Paralelismo a nivel de instrucción y multithreading.

- Técnicas de compilación básicas.
- Planificación dinámica de instrucciones: dependencias y renombramiento de registros
- Técnicas de predicción de saltos.
- Ejecución especulativa.
- Técnicas de lanzamiento múltiple de instrucciones
- Arquitectura de procesadores superescalares fuera-de-orden
- Límites del paralelismo a nivel de instrucción.
- Ejemplos: Evolución de arquitecturas Intel
- Multithreading: concepto y tipos
- Ejemplos de arquitecturas multithread

Módulo 3. Paralelismo a nivel de datos.

- Concepto de arquitectura vectorial
- Instrucciones SIMD para procesamiento multimedia
- Unidades para procesamiento gráfico (GPUs)
- Paralelismo a nivel bucle: vectorización

Módulo 4. Multiprocesadores

- Conceptos básicos de multiprocesamiento
- La red de interconexión
- Arquitecturas de memoria compartida centralizada.
- Coherencia de cache: protocolos.
- Arquitectura de memoria compartida distribuida.
- Coherencia basada en directorio.
- Sincronización. Primitivas de sincronización.
- Concepto de consistencia de memoria: modelos.

**Programa detallado en inglés:**

- o Module 1. Introduction and trends in computer architecture  
Context of the course  
Technological evolution. The technology -architecture interaction.  
Energy consumption  
Key components of cost.  
Measuring performance
- o Module 2. Instruction-level parallelism (ILP) and multithreading.  
Basic compilation techniques  
Dynamic instruction scheduling: dependences and register renaming  
Branch prediction

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



- Speculative execution
- Multiple issue techniques
- Limits of ILP
- Architecture of superscalar out-of-order processors
- Examples: Evolutions of Intel architectures
- Multithreading: concept and types
- Examples of multithread architectures
- o Module 3. Data-level parallelism
  - Vector architecture
  - SIMD instruction set extensions for multimedia
  - Graphics processing units (GPUs)
  - Loop-level parallelism: vectorization
- o Module 4. Multiprocessors
  - Basic concepts of multiprocessing
  - The interconnection network
  - Centralized shared memory architectures
  - Cache coherence: protocols.
  - Distributed shared memory architectures
  - Directory-based cache coherence
  - Synchronization: primitives
  - Concept of memory consistency: models

**Competencias de la asignatura:**

**Generales:**

No tiene

**Específicas:**

CE\_GIC3-Capacidad de analizar y evaluar arquitecturas de computadores, incluyendo plataformas paralelas y distribuidas, así como desarrollar y optimizar software para las mismas.

**Básicas y Transversales:**

CT1-Capacidad de comunicación oral y escrita, en inglés y español utilizando los medios audiovisuales habituales, y para trabajar en equipos multidisciplinares y en contextos internacionales.

CT2-Capacidad de análisis y síntesis en la resolución de problemas.

CT3-Capacidad para gestionar adecuadamente la información disponible integrando creativamente conocimientos y aplicándolos a la resolución de problemas informáticos utilizando el método científico.

**Resultados de aprendizaje:**

Analizar las limitaciones de paralelismo a nivel de instrucciones y estudiar alternativas arquitectónicas para superar dichas limitaciones (CE\_GIC3)

Comprender el paralelismo a nivel de datos y las arquitecturas que lo explotan, así como las instrucciones vectoriales y los mecanismos de vectorización (CE\_GIC3)

Comprender la influencia mutua entre tecnología de circuitos integrados y diseño arquitectónico, así como sus repercusiones sociales y económicas (CE\_GIC3)

Comprender la organización de las arquitecturas con paralelismo a nivel de "thread", y analizar los diferentes mecanismos de cambio de "thread" (CE\_GIC3)

Comprender los mecanismos para gestionar la ejecución de instrucciones en desorden y la especulación de saltos, así como el incremento de la capacidad de procesamiento que se deriva de dichas técnicas (CE\_GIC3)

Conocer la organización de sistemas multiprocesador, y comprender los mecanismos para el intercambio de información entre los diferentes procesadores, así como analizar las alternativas para el diseño de su jerarquía de memoria (CE\_GIC3)

Conocer la terminología propia de la arquitectura de computadores en lengua inglesa y manejar fuentes bibliográficas en dicha lengua (CT1, CE\_GIC3)

Conocer las medidas para expresar el rendimiento de computadores en diferentes contextos, con inclusión de los aspectos térmicos y económicos (CE\_GIC3)

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento:



Resolver problemas de Arquitectura de Computadores seleccionando la solución más adecuada entre las posibles. (CT2, CT3)

**Evaluación:**

Todas las pruebas realizadas en cada asignatura serán comunes a todos los grupos de la misma.  
 La calificación final tendrá en cuenta:  
 Exámenes sobre la materia: 60-90%  
 Otras actividades: 10-40%  
 En el apartado "Otras actividades" se podrá valorar la participación activa en el proceso de aprendizaje, la realización de prácticas y ejercicios y la realización de otras actividades dirigidas. La realización de las prácticas de laboratorio y del resto de las actividades evaluables será obligatoria.  
 Antes del comienzo de cada curso escolar se concretarán en las fichas docentes los porcentajes exactos que se utilizarán durante ese curso para la evaluación de la materia, siendo comunes estos criterios para todos los grupos de una misma asignatura.  
 La calificación reflejará los resultados de aprendizaje de las diferentes competencias que se adquieren en el módulo o materia.

**Evaluación detallada:**

A mitad del cuatrimestre habrá una prueba de clase opcional y no liberatoria, cuyo peso en la nota de la asignatura será del 20% para los alumnos que deseen realizarla.  
  
 Convocatoria de junio: Examen final obligatorio y escrito, formado por teoría y problemas. La nota de esta convocatoria será la mayor de la dos siguientes:  
 - Nota de la prueba de clase x 0,2 + Nota examen x 0,7 + Nota entrega ejercicios x 0,1  
 - Nota examen x 0,9 + Nota entrega ejercicios x 0,1.  
  
 Convocatoria de septiembre: Examen final escrito que incluye teoría y problemas. La nota de esta convocatoria será Nota examen x 0,9 + Nota entrega ejercicios x 0,1  
  
 La nota entrega de ejercicios será la obtenida en la convocatoria de junio.  
  
 Mismo examen (prácticas y otros elementos de evaluación, en su caso) en todos los grupos y criterios detallados de puntuación comunes.

**Exámenes:**

|   |   |
|---|---|
| <input checked="" type="checkbox"/> En Aula   | <input type="checkbox"/> En Lab                     |
| <b>Parciales</b><br>(solo anuales)            | <b>Finales</b>                                      |
| <input type="checkbox"/> 1er Cuat.            | <input checked="" type="checkbox"/> Final Ordinaria |
| <input checked="" type="checkbox"/> 2do Cuat. | <input type="checkbox"/> Final Extraordinaria       |
| <input type="checkbox"/> Sin Examen           |   |

**Actividades formativas:**

Las actividades formativas que se van a realizar para esta materia se dividen en tres grupos:  
 Actividades presenciales: 30-40% de la dedicación del alumno. Estas actividades podrán incluir:  
 Clases teóricas magistrales.  
 Clases de problemas.  
 Laboratorios.  
 Seminarios.  
 Actividades dirigidas: 10-15% de la dedicación del alumno. Estas actividades podrán incluir:  
 Trabajos dirigidos.  
 Tutorías dirigidas.  
 Trabajo personal: 50-55% de la dedicación del alumno. Estas actividades podrán incluir:  
 Trabajo personal no dirigido: Estudio, preparación de exámenes, realización de ejercicios.  
 Realización de exámenes.

**Actividades docentes:**

|                             |   |
|-----------------------------|---|
| <b>Reparto de créditos:</b> | <b>Otras actividades:</b>   |
| Teoría: 4,50                | Clases teóricas: 3 horas de clases teóricas a la semana en aula.                            |
| Problemas: 1,50             | Clases prácticas: 1 hora de clase práctica (problemas, discusión, ...) a la semana en aula. |
| Laboratorios: 0,00          | Total horas presenciales: 4h. (40%)   |

**Bibliografía:**

Básica:  
 - Hennessy, J. L., Patterson, D.A., "Computer Architecture: A Quantitative Approach", 5ª Ed., Morgan-Kaufmann, 2012.  
 - Culler, D., Singh, J., "Parallel Computer Architecture: A Hardware/Software Approach", Morgan Kaufmann, 1999

Complementaria:  
 - Baer, J.-L., "Microprocessor Architecture", Cambridge University Press, 2010  
 - Shen, J.P., Lipasti, M.H., "Modern Processor Design", McGraw Hill, 2005  
 - Sima, D., Fountain, T., Kacsuk, P., "Advanced Computer Architectures: A Design Space Approach", Addison Wesley, 1997.

|                                      |
|--------------------------------------|
| Fecha: ____ de _____ de ____         |
| Firma del Director del Departamento: |



**UNIVERSIDAD COMPLUTENSE DE MADRID**  
**FACULTAD DE INFORMATICA**

Fecha: \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_

Firma del Director del Departamento: