



## Ficha del curso: 2015-2016

<b>Grado:</b> GRADO EN INGENIERÍA INFORMÁTICA		<b>Curso:</b> Optativas generales 3º y 4º ( 1C )
<b>Asignatura:</b> 803302 - Programación paralela para móviles y multicores	<b>Abrev:</b> PP	<b>6 ECTS</b>
<b>Asignatura en Inglés:</b>		
<b>Materia:</b> Complementos de computadores		<b>12 ECTS</b>
<b>Otras asignaturas en la misma materia:</b>		
Seguridad en redes		6 ECTS
<b>Módulo:</b> Optativo		
<b>Departamento:</b> Arquitectura de Computadores y Automática		<b>Coordinador:</b> Prieto Matias, Manuel

<b>Descripción de contenidos mínimos:</b> Procesamiento Paralelo
<b>Programa detallado:</b> Tema 1: Introducción a la Programación paralela 1.1 Motivación. ¿por qué la computación paralela?. Prestaciones 1.2 Memoria Compartida vs Memoria Distribuida. Paradigmas de programación. 1.3 Sistemas multicore y dispositivos móviles. Arquitecturas. Tema 2: Programación paralela en dispositivos móviles 2.1 Introducción a la plataforma Android. 2.2 Hilos, controladores. 2.3 Render Script 2.4 MARE (Multicore Asynchronous Runtime Environment) 2.5 OpenCL para Android 2.6 Casos Prácticos (Prácticas tema 2) Tema 3- Programación paralela con OpenMP. 3.1 Introducción a OpenMP. Evolución de las diferentes versiones. 3.2 Directivas OpenMP 3.3 Clausulas y Funciones de OpenMP 3.4 Ejecución múltiples hebras. 3.5 Variables de Entorno 3.6 Dependencia de datos, Interferencia. Sincronización. 3.7 Planificación y Transformación de bucles. 3.8 Rendimiento en OpenMP 3.9 Casos prácticos (Prácticas Tema 3) Tema 4 Nuevos paradigmas y herramientas de programación para Multicores 4.1 Entorno Intel Parallel Composer 4.2 Intel TBB/ ArBB 4.3 Cilk, Cilk+ . 4.4 Comparación de estos paradigmas con OpenMP (Prácticas Tema 4)
<b>Programa detallado en inglés:</b> No tiene
<b>Competencias de la asignatura:</b>
<b>Generales:</b> CG16-Conocimiento y aplicación de las características, funcionalidades y estructura de los Sistemas Distribuidos, las Redes de Computadores e Internet y diseñar e implementar aplicaciones basadas en ellas.  CG19-Conocimiento y aplicación de los principios fundamentales y técnicas básicas de la programación paralela, concurrente, distribuida y de tiempo real.
<b>Específicas:</b> No tiene
<b>Básicas y Transversales:</b> 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.

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

Firma del Director del Departamento:



# UNIVERSIDAD COMPLUTENSE DE MADRID

## FACULTAD DE INFORMATICA

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.

CT4-Capacidad de organización, planificación, ejecución y dirección de recursos humanos.

CT5-Capacidad para valorar la repercusión social y medioambiental de las soluciones de la ingeniería, y para perseguir objetivos de calidad en el desarrollo de su actividad profesional.

### Resultados de aprendizaje:

No tiene

### Evaluación:

Todas las pruebas realizadas en cada asignatura serán comunes a todos los grupos de la misma.

Al tener las materias optativas muy diversas características la calificación de las mismas podrá ser muy variada, por lo que los rangos se dejan muy abiertos:

- Exámenes sobre la materia: 0-60%
- Otras actividades: 100-40%

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:

Dos modalidades

Evaluación Continua: Prácticas (50%) + Exposición y entrega de trabajos (50%) o bien

Calificación Examen final: 100% examen final.

### Exámenes:

- |   |                                      |
|---|--------------------------------------|
| <input checked="" type="checkbox"/> En Aula   | <input type="checkbox"/> En Lab      |
| <input checked="" type="checkbox"/> Final Feb | <input type="checkbox"/> Parcial Feb |
| <input type="checkbox"/> Final Jun            | <input type="checkbox"/> Parcial Jun |
| <input checked="" type="checkbox"/> Final Sep | <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:

Reparto de créditos:

Teoría: 3,00

Problemas: 0,00

Laboratorios: 3,00

Otras actividades:

No tiene

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

Firma del Director del Departamento:



**Bibliografía:**

- <http://developer.android.com/guide/topics/renderscript/index.html>
- Pro Android Apps Performance Optimization, Herv Guihot, Apress. ISBN-10: 1430239999
- <https://code.google.com/p/aopencl/>
- <https://developer.qualcomm.com/mobile-development/maximize-hardware/parallel-computing-mare>
- Introduction to Parallel Programming, Peter Pacheco (Autor)
- Using OpenMP: Portable Shared Memory Parallel Programming (Scientific and Engineering Computation) Barbara Chapman (Autor), Gabriele Jost (Autor), Ruud van der Pas (Autor), MIT press 2008.
- Structured Parallel Programming: Patterns for Efficient Computation. Michael McCool (Author), James Reinders (Author), Arch Robison, 2012.
- Intel Threading Building Blocks : outfitting C++ for multi-core processor parallelism. James Reinders (Author)

Ficha docente guardada por última vez el 09/07/2014 19:55:00 por el departamento: Arquitectura de Computadores y Automática

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

Firma del Director del Departamento:



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

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

Firma del Director del Departamento: