

## Proyecto/Guía docente de la asignatura “Computación Paralela y Modelos Emergentes”

<b>Asignatura</b>	COMPUTACIÓN PARALELA Y MODELOS EMERGENTES	
<b>Materia</b>	SISTEMAS Y SERVICIOS EMPOTRADOS, UBICUOS Y DE ALTAS PRESTACIONES	
<b>Módulo</b>	(vacío)	
<b>Titulación</b>	MASTER EN INGENIERÍA INFORMÁTICA	
<b>Plan</b>	510	<b>Código 53170</b>
<b>Periodo de impartición</b>	2º CUATRIMESTRE	<b>Tipo/Carácter Obligatoria</b>
<b>Nivel/Ciclo</b>	MASTER	<b>Curso 2019-2020</b>
<b>Créditos ECTS</b>	6 ECTS	
<b>Lengua en que se imparte</b>	CASTELLANO	
<b>Profesor/es responsable/s</b>	BENJAMÍN SAHELICES FERNÁNDEZ (responsable teoría y prácticas) ARTURO GONZÁLEZ ESCRIBANO	
<b>Datos de contacto (E-mail, teléfono...)</b>	E-MAIL: <a href="mailto:benja@infor.uva.es">benja@infor.uva.es</a> E-MAIL: <a href="mailto:arturo@infor.uva.es">arturo@infor.uva.es</a>	
<b>Horario de tutorías</b>	Véase <a href="http://www.uva.es">www.uva.es</a> → Grados → Grado en Ingeniería Informática → Tutorías	
<b>Departamento</b>	DEPARTAMENTO DE INFORMÁTICA	

### 1. Situación / Sentido de la Asignatura

#### 1.1 Contextualización

Dentro de la perspectiva de los estudios de un máster sobre informática de perfil profesionalizante, en esta asignatura se pretende, en su primer bloque temático, aportar conocimientos sobre conceptos de programación para la obtención de altas prestaciones y paradigmas de programación en plataformas de computo paralelo. En el segundo bloque temático el objetivo es que el estudiante adquiera competencias sobre nuevas tecnologías relacionadas con sistemas distribuidos, cluster escalables de computación, virtualización en clústers y centros de datos y arquitecturas para plataformas cloud.

#### 1.2 Relación con otras materias

La materia guarda relación con asignaturas de programación, sistemas operativos, administración de sistemas, seguridad, evaluación, rendimiento y arquitectura de computadores.

### 1.3 Prerrequisitos

Conocimientos sobre programación, sistemas operativos y arquitectura de computadores a nivel de Grado en Informática.

## 2. Competencias

### 2.1 Generales

Código	Descripción
CG1	Capacidad para proyectar, calcular y diseñar productos, procesos e instalaciones en todos los ámbitos de la ingeniería informática
CG2	Capacidad para la dirección de obras e instalaciones de sistemas informáticos, cumpliendo la normativa vigente y asegurando la calidad del servicio.
CG4	Capacidad para el modelado matemático, cálculo y simulación en centros tecnológicos y de ingeniería de empresa, particularmente en tareas de investigación, desarrollo e innovación en todos los ámbitos relacionados con la Ingeniería en Informática.
CG6	Capacidad para la dirección general, dirección técnica y dirección de proyectos de investigación, desarrollo e innovación, en empresas y centros tecnológicos, en el ámbito de la Ingeniería Informática.
CG7	Capacidad para la puesta en marcha, dirección y gestión de procesos de fabricación de equipos informáticos, con garantía de la seguridad para las personas y bienes, la calidad final de los productos y su homologación.
CG8	Capacidad para la aplicación de los conocimientos adquiridos y de resolver problemas en entornos nuevos o poco conocidos dentro de contextos más amplios y multidisciplinares, siendo capaces de integrar estos conocimientos.

### 2.2 Específicas

Código	Descripción
CET1	Capacidad para modelar, diseñar, definir la arquitectura, implantar, gestionar, operar, administrar y mantener aplicaciones, redes, sistemas, servicios y contenidos informáticos.
CET6	Capacidad para diseñar y evaluar sistemas operativos y servidores, y aplicaciones y sistemas basados en computación distribuida.
CET7	Capacidad para comprender y poder aplicar conocimientos avanzados de computación de altas prestaciones y métodos numéricos o computacionales a problemas de ingeniería.
CET8	Capacidad de diseñar y desarrollar sistemas, aplicaciones y servicios informáticos en sistemas empotrados y ubicuos.
CET11	Capacidad para conceptualizar, diseñar, desarrollar y evaluar la interacción persona-ordenador de productos, sistemas, aplicaciones y servicios informáticos.

## 3. Objetivos

Comprender los principios de la programación de altas prestaciones, en plataformas de computo no convencionales o con nuevos modelos de programación que explotan las capacidades de paralelismo de los

sistemas actuales y futuros. Asimilar los problemas y retos de la programación paralela y de alto rendimiento en diferentes niveles de abstracción y detalle. Desarrollar habilidades y dominar técnicas de programación para el desarrollo de aplicaciones y servicios con necesidades de alto rendimiento en arquitecturas reales. Conocer las tecnologías emergentes en el ámbito de los clúster de computación, la virtualización y los centros de datos. Comprender los principios de funcionamiento de los cluster implantados sobre centros de datos virtualizados.

#### 4. Bloques temáticos

---

##### Bloque 1: Modelos de computación paralela

---

Carga de trabajo en créditos ECTS: 2

##### a. Contextualización y justificación

---

El desarrollo de aplicaciones eficientes, en cualquier campo de la ciencia y la ingeniería, que exploten el potencial de cómputo de las nuevas plataformas y arquitecturas actuales, implica la utilización de técnicas y modelos de programación que exploten el paralelismo. En este primer bloque se expondrá una visión general de los diferentes modelos y aproximaciones de programación sobre nuevas plataformas paralelas desde una perspectiva multinivel. Se describirán y explorarán técnicas de programación paralela en diferentes niveles para arquitecturas modernas.

##### b. Objetivos de aprendizaje

---

Comprender los problemas inherentes a la programación paralela en diferentes niveles de abstracción. Entender los problemas conceptuales y las soluciones aportadas por diferentes modelos de programación. Asimilar técnicas y paradigmas básicos de programación de propósito general en arquitecturas paralelas. Entender las interrelaciones entre el sistema de memoria, los paradigmas de programación y el rendimiento obtenido en estos dispositivos.

##### c. Contenidos

---

###### TEMA 1: Modelos de computación paralela

- 1.1. Arquitecturas y algoritmos paralelos
- 1.2 Modelos de programación paralela
- 1.3 Estructura de aplicaciones paralelas

###### TEMA 2: Paradigmas de programación paralela

- 2.1 Explotación de sistemas multicore

2.2 Paradigmas y patrones de explotación de paralelismo

2.3 Técnicas de partición, planificación y programación de tareas

---

#### **d. Métodos docentes**

---

Actividades presenciales. En cada sesión se realizarán diferentes actividades dirigidas a facilitar la adquisición de conocimientos, habilidades y competencias:

- Clase magistral participativa
- Resolución de problemas de forma individual y por grupos
- Ejercicios de programación

Actividades no presenciales: El alumno realizará una serie de actividades fuera del aula:

- Preparación de sesiones presenciales: Lectura de bibliografía, búsquedas de información, preparación de dudas.
- Resolución de ejercicios y actividades de programación

---

#### **e. Plan de trabajo**

---

Se organizará en unas 20 horas en sesiones presenciales en aula y a realizar en el laboratorio. En cada sesión se realizarán diferentes actividades.

---

#### **f. Evaluación**

---

Véase el punto 7 sobre evaluación

---

#### **g. Bibliografía básica**

---

- Introducción a la Programación Paralela. Francisco Almeida et.al. Paraninfo 2008, ISBN 978-84-9732-674-2
- Multicore and GPU Programming: An Integrated Approach. Gerassimos Barlas. Morgan Kaufmann, 2015. ISBN: 978-0-12-417137-4

---

#### **h. Bibliografía complementaria**

---

- Introduction to Parallel Computing (2nd Edition), Grama et.al. 2003, ISBN 0-201-64865-2
- Programming Models for Parallel Computing. Pavan Balaji. The MIT Press, 2015. ISBN: 978-0262528818
- High Performance Computing: Programming and Applications. John Levesque. Chapman and Hall/CRC, 2010. ISBN: 978-1420077056

### **i. Recursos necesarios**

---

Laboratorio de ordenadores del centro. Cada ordenador tiene el software y recursos necesarios para la ejecución de los ejercicios y consultas a través de la red.

Plataforma de docencia virtual Moodle: Campus virtual de la UVa (<http://campusvirtual.uva.es/>)

---

**Bloque 2: Computación distribuida y cloud**

---

**Carga de trabajo en créditos ECTS: 4**

---

**a. Contextualización y justificación**

---

En el ámbito de la computación distribuida actualmente están asentadas tecnologías que permiten una gestión virtualizada de los medios de computación y almacenamiento unificada de centros de datos y clusters de computadoras. Esta gestión permite ofrecer a los usuarios servicios flexibles a nivel de infraestructura, plataforma y aplicación con grandes ventajas en ahorro energético, ahorro de recursos y flexibilidad en los servicios proporcionados. El estudiante adquirirá los conocimientos necesarios para comprender, diseñar y administrar sistemas distribuidos tipo cluster, servicios de virtualización y computación cloud.

---

**b. Objetivos de aprendizaje**

---

- Comprender las tecnologías que soportan los modelos de computación distribuida y sus implicaciones en la escalabilidad, las redes, la computación cloud, las prestaciones, la seguridad y la eficiencia energética.
- Conocer la arquitectura básica y los conceptos fundamentales de los clústers, las arquitecturas MPP y sus principios de diseño. Asimismo comprender las técnicas básicas de gestión de trabajos.
- Comprender los principios de la virtualización, los niveles de virtualización, las principales herramientas disponibles, los conceptos básicos de virtualización de CPU, memoria y E/S, la virtualización de clústers y de centros de datos.
- Conocer las arquitecturas de plataformas cloud sobre centros de datos virtualizados. Entender los principios básicos de diseño y las implicaciones de interconexión, seguridad, fiabilidad y de almacenamiento junto con los tipos de plataformas cloud existentes.

---

**c. Contenidos**

---

**TEMA 3. Distributed Systems, Cluster, Virtualization and Clouds****3.1 Distributed Systems**

- Internet Computing
- Scalable computing trends
- Service-Oriented Architecture (SOA)
- Parallel and Distributed Programming Models
- Performance

**3.2 Clusters**

- Clusters, grids, peer-to-peer, cloud
- Computer clusters. Design principles.
- MPP clusters

- Job scheduling
  - Top500
- 3.3 Virtualization
- Virtual machines
  - Abstraction levels of virtualization
  - Hardware virtualization
  - OS-level virtualization
  - Hypervisor architecture
  - Host-based architecture
  - Para-virtualization
  - Hardware support for virtualization
  - CPU virtualization
  - Memory virtualization
  - I/O virtualization
  - Physical vs. virtual clusters
  - Fast deployment, migration and scheduling
  - Server consolidation in data centers
  - Virtual storage management
  - Cloud for virtualized data centers
- 3.4 Clouds
- Data center virtualization for cloud computing
  - Public, private and hybrid clouds
  - Network structure of data centers
  - Cloud ecosystem
  - IaaS
  - PaaS
  - SaaS
  - Warehouse-scale data center design
  - Data center interconnection network
  - Cloud architecture design and development
  - Virtualization support for the cloud
  - Architectural design challenges
  - Cloud security

#### **TEMA 4 Cloud Technology and Services**

- 4.1 Cloud technology
- Cloud computing + virtualization
  - MapReduce. Hadoop. Pig.
- 4.2 Cloud services
- Web services
  - SOA and REST
  - The cloud stack
  - SaaS
  - PaaS
  - IaaS
- 4.3 IBM Bluemix (IBM Cloud)
- Introduction

#### **TEMA 5 Data and Intelligence in the Cloud**

### 5.1 Data in the Cloud

- DBaaS
- BASE
- Cassandra

### 5.2 Intelligence in the Cloud

- Natural language processing
- Searching
- Collective intelligence

---

## d. Métodos docentes

---

- Actividades presenciales:
  - Sesiones de aula: discusión y explicación con los estudiantes sobre los contenidos teóricos y las dudas y aportaciones que planteen según su preparación previa de la clase. Estudios del caso. Presentación y discusión de problemas y trabajos realizados por los estudiantes.
  - Sesiones de laboratorio: análisis e implementación de supuestos prácticos usando los recursos informáticos de la Escuela de Ingeniería Informática.
- Actividades no presenciales:
  - El estudiante deberá prepararse previamente las clases mediante la lectura de la bibliografía, la ampliación de la información y la preparación de dudas.
  - El estudiante tendrá que completar el trabajo de aula y laboratorio. En concreto seguirá trabajando e investigando sobre los trabajos prácticos que se le encarguen.

---

## e. Plan de trabajo

---

Semana 6:

Aula: capítulo 3

Laboratorio: trabajo práctico y estudio del caso sobre clusters de computación

Semana 7:

Aula: capítulo 3

Laboratorio: creación de un clúster de computación utilizando máquinas virtuales. Configuración de la interconexión.

Semana 8:

Aula: capítulo 3

Laboratorio: instalación de un sistema de gestión de cluster (MOSIX)

Semana 9:

Aula: presentación de orientación y desarrollo del trabajo teórico. Bibliografía. Asignación de trabajos.

Laboratorio: introducción a Hadoop YARN

Semana 10:

Aula: introducción a Hadoop YARN

Laboratorio: presentación trabajo práctico MOSIX en el laboratorio.

Semana 11:

Aula: capítulo 4

Laboratorio: prácticas Hadoop YARN

Semana 12:

Aula: capítulo 5

Laboratorio: prácticas Hadoop YARN

Semana 13:

Aula: introducción a OpenStack

Laboratorio: trabajo práctico con Openstack

Semana 14:

Aula: sesión de orientación y control de progreso del trabajo teórico

Laboratorio: defensa en aula del trabajo teórico

Semana 15:

Aula: trabajo práctico Hadoop YARN

Laboratorio: presentación en el laboratorio del trabajo práctico Hadoop YARN

---

#### **f. Evaluación**

Véase el punto 7 sobre evaluación

---

#### **g. Bibliografía básica**

- "Distributed and Cloud Computing. From Parallel Processing to the Internet of Things", Kai Hwang, Geoffrey C. Fox, Jack J. Dongarra, Morgan Kaufmann, 2012. ISBN: 978-0-12-385880-1
- "Guide to Cloud. Principles and Practice", Richard Hill, Laurie Hirsch, Peter Lake, Siavash Moshiri, Springer, 2013. ISBN: 978-1-4471-4602-5. DOI 10.1007/978-1-4471-4603-2
- "Apache Hadoop YARN. Moving beyond MapReduce and Batch Processing with Apache Hadoop 2", Arun C. Murthy, Vinod K. Vavilapalli, Addison Wesley Data and Analytics Series, 2014. ISBN 978-0-321-93450-5
- "The Datacenter as a Computer. An Introduction to the Design of Warehouse-Scale Machines. Second Edition.", Luiz André Barroso, Jimmy Clidaras, Urs Holzle, Synthesis Lectures on Computer Architecture, Morgan&Claypool Publishers, 2013. ISBN print: 978-0-270-50098. ISBN ebook: 9781627050104
- "Virtual Machines. Versatile Platforms for Systems and Processes", James E. Smith and Ravi Nair, Morgan Kaufmann, 2005, ISBN-13: 978-1-55860-910-5. ISBN-10: 1-55860-910-5

---

#### **h. Bibliografía complementaria**

Los recursos adicionales serán accesibles desde la plataforma Moodle.

**i. Recursos necesarios**

Laboratorio de ordenadores del centro. Cada ordenador tiene el software y recursos necesarios para la ejecución de los ejercicios y consultas a través de la red. Es muy recomendable que el estudiante disponga de un portátil

Plataforma de docencia virtual Moodle: Campus virtual de la Escuela de Ingeniería Informática (<http://aulas.inf.uva.es/>)

**5. Métodos docentes y principios metodológicos**

Actividad	Metodología
<b>Clase de teoría</b>	En estas sesiones, de forma motivadora y que mueva al alumno a su implicación personal, se tratará de dirigir a éste hacia los conceptos claves y se le iniciará en el planteamiento de los principales problemas. En todo momento se tratará de utilizar un enfoque de conocimiento de concepto orientado a la resolución de problemas prácticos. Se fomentará la participación del estudiante y se evaluará la preparación que realice de las clases.
<b>Clase práctica</b>	En las prácticas de laboratorio supervisadas se pretende que el alumno realice un primer contacto directo con los conceptos abordados en la materia tal como éstos aparecen en un entorno profesional. En las sesiones prácticas se le plantearán al alumno casos concretos que debe resolver haciendo uso de herramientas profesionales. Fundamentalmente estarán orientadas a la familiarización con herramientas orientadas a posibilitar la medida del rendimiento de la ejecución de las aplicaciones informáticas bajo sus diversos aspectos.
<b>Tutoría</b>	En las sesiones de tutoría cada alumno podrá plantear personalmente al profesor de la asignatura cuantas cuestiones le hayan surgido durante su trabajo con la materia. Asimismo, el profesor dará a cada estudiante los consejos que considere oportunos para ayudarle a obtener un aprovechamiento óptimo en el proceso de adquisición de competencias que corresponde a esta materia.

**6. Tabla de dedicación del estudiante a la asignatura**

ACTIVIDADES PRESENCIALES	HORAS	ACTIVIDADES NO PRESENCIALES	HORAS
Clases teórico-prácticas (T/M)	22	Estudio y trabajo autónomo individual	90
Clases prácticas de aula (A)	8	Estudio y trabajo autónomo grupal	0
Laboratorios (L)	30		
Prácticas externas, clínicas o de campo	0		
Seminarios (S)	0		
Tutorías grupales (TG)	0		

Evaluación (fuera del periodo oficial de exámenes)	0		
<b>Total presencial</b>	<b>60</b>	<b>Total no presencial</b>	<b>90</b>

## 7. Sistema y características de la evaluación

INSTRUMENTO/PROCEDIMIENTO	PESO EN LA NOTA FINAL	OBSERVACIONES
Bloque 1: Evaluación continua en convocatoria ordinaria	35 %	No se realizará examen escrito en la convocatoria ordinaria. La nota final del estudiante estará formada por las notas de tests, tareas y presentaciones sobre la parte teórica y la parte práctica de la asignatura.
Bloque 1: Examen final escrito en convocatoria extraordinaria	35%	Los estudiantes que no hayan superado la asignatura en la convocatoria ordinaria realizarán un examen escrito en la convocatoria extraordinaria
Bloque 2: Evaluación continua en convocatoria ordinaria	65 %	No se realizará examen escrito en la convocatoria ordinaria. La nota final del estudiante estará formada por las discusiones con los estudiantes y la presentación de trabajos sobre la parte teórica y la parte práctica de la asignatura.
Bloque 2: Examen final escrito en convocatoria extraordinaria	65 %	Los estudiantes que no hayan superado la asignatura en la convocatoria ordinaria realizarán un examen escrito en la convocatoria extraordinaria

**Bloque 1.** Convocatoria extraordinaria. Examen escrito con una duración total de 45 minutos.

Problemas cortos y cuestiones sobre los contenidos teóricos y prácticos. Valoración: 100% de la nota del bloque 1.

**Bloque 2.** Convocatoria extraordinaria. Examen escrito con dos partes con una duración total de 2,5 horas:

- Problemas cortos y cuestiones sobre los contenidos teóricos. Valoración: 50 %. En concreto 5 puntos como máximo de un total de 10.
- Cuestiones cortas sobre los tratado en las prácticas de laboratorio. Valoración: 50 %. En concreto 5 puntos como máximo de un total de 10.

### CRITERIOS DE CALIFICACIÓN

- **Bloque 1 y 2. Convocatoria ordinaria: la calificación final tendrá en cuenta los siguientes aspectos**
  - Preparación previa de las clases, conocimiento de conceptos, definiciones y propiedades de lo

explicado en clase. Capacidad para razonar y responder sobre estos conceptos.

- Comprensión correcta de los trabajos realizados sobre cada capítulo. Correcta explicación de lo realizado.
- Comprensión de los conceptos que soportan los trabajos prácticos. Capacidad de razonamiento sobre el trabajo realizado, sobre las posibles modificaciones, errores cometidos y forma de realizar otros sistemas de similares características.
- Claridad y coherencia en las explicaciones y respuestas a las cuestiones planteadas por el profesor

- **Bloques 1 y 2. Convocatoria extraordinaria:**

- **A la hora de calificar las pruebas se considerarán fundamentalmente los siguientes aspectos**

- Correcta utilización de conceptos, definiciones y propiedades relacionadas con la naturaleza de la situación que se trata de resolver o explicar.
- Justificaciones teóricas que se aporten para el desarrollo de las respuestas. Se penalizará la no justificación, ausencia de explicaciones o explicaciones incorrectas.
- Claridad y coherencia en la exposición.

**8. Consideraciones finales****Distribución temporal de los bloques temáticos**

BLOQUE TEMÁTICO	CARGA ECTS	PERIODO PREVISTO DE DESARROLLO
Bloque 1: Modelos de computación paralela	2,0 ECTS	Semanas 1 a 5
Bloque 2: Computación Distribuida y Cloud	4,0 ECTS	Semanas 6 a 15