

**ANX-PR/CL/001-02**  
**GUÍA DE APRENDIZAJE**

**ASIGNATURA**

Programacion paralela y distribuida

**CURSO ACADÉMICO - SEMESTRE**

2015-16 - Primer semestre

## Datos Descriptivos

---

<b>Nombre de la Asignatura</b>	Programacion paralela y distribuida
<b>Titulación</b>	10AN - Master Universitario en Ingenieria Informatica
<b>Centro responsable de la titulación</b>	E.T.S. de Ingenieros Informaticos
<b>Semestre/s de impartición</b>	Tercer semestre
<b>Carácter</b>	Optativa
<b>Código UPM</b>	103000637
<b>Nombre en inglés</b>	Parallel And Distributed Programming

## Datos Generales

---

<b>Créditos</b>	4.5	<b>Curso</b>	2
<b>Curso Académico</b>	2015-16	<b>Período de impartición</b>	Septiembre-Enero
<b>Idioma de impartición</b>	Castellano	<b>Otros idiomas de impartición</b>	

## Requisitos Previos Obligatorios

---

### Asignaturas Superadas

El plan de estudios Master Universitario en Ingenieria Informatica no tiene definidas asignaturas previas superadas para esta asignatura.

### Otros Requisitos

El plan de estudios Master Universitario en Ingenieria Informatica no tiene definidos otros requisitos para esta asignatura.

## Conocimientos Previos

---

### Asignaturas Previas Recomendadas

Computacion para ciencias e ingenieria

### Otros Conocimientos Previos Recomendados

Conocimientos básicos de concurrencia.

Poseer destrezas fundamentales de la programación

## Competencias

---

CB10 - Que los estudiantes posean las habilidades de aprendizaje que les permitan continuar estudiando de un modo que habrá de ser en gran medida autodirigido o autónomo.

CB7 - Que los estudiantes sepan aplicar los conocimientos adquiridos y su capacidad de resolución de problemas en entornos nuevos o poco conocidos dentro de contextos más amplios (o multidisciplinares) relacionados con su área de estudio

CE1 - Capacidad para la integración de tecnologías, aplicaciones, servicios y sistemas propios de la Ingeniería Informática, con carácter generalista, y en contextos más amplios y multidisciplinares.

CE10 - Capacidad para comprender y poder aplicar conocimientos avanzados de computación de altas prestaciones y métodos numéricos o computacionales a problemas de ingeniería.

CE16 - Habilidad para hacer conexiones entre los deseos y necesidades del consumidor o cliente y lo que la tecnología puede ofrecer

CE4 - Capacidad para modelar, diseñar, definir la arquitectura, implantar, gestionar, operar, administrar y mantener aplicaciones, redes, sistemas, servicios y contenidos informáticos.

CG12 - Capacidad de trabajar de forma independiente en su campo profesional

CG9 - Apreciación de los límites del conocimiento actual y de la aplicación práctica de la tecnología más reciente

## Resultados de Aprendizaje

---

RA3 - Aplicar técnicas y herramientas de computación de alto rendimiento para la solución de problemas prácticos

RA163 - RA1

## Profesorado

---

### Profesorado

Nombre	Despacho	e-mail	Tutorías
Garcia Dopico, Antonio ( <b>Coordinador/a</b> )	4202	antonio.garcia.dopico@upm.es	L - 16:00 - 17:00 M - 10:00 - 12:00 X - 10:00 - 11:00 J - 16:00 - 17:00 V - 10:00 - 11:00
Perez Costoya, Fernando	4201	fernando.perez@upm.es	L - 11:30 - 13:30 X - 11:00 - 13:00 J - 11:30 - 13:30 V - 10:00 - 12:00
Perez Hernandez, María De Los Santos	4204	maria.s.perez@upm.es	M - 10:30 - 12:30 X - 16:00 - 18:00 J - 16:00 - 18:00

**Nota.-** Las horas de tutoría son orientativas y pueden sufrir modificaciones. Se deberá confirmar los horarios de tutorías con el profesorado.

## Descripción de la Asignatura

---

## Temario

---

1. Tema 1: Introducción. Necesidades de la computación científica
  - 1.1. 1.1 Necesidades de cómputo. Arquitecturas de altas prestaciones. Top 500. Green 500.
  - 1.2. 1.2 Sistemas de memoria compartida.
  - 1.3. 1.3 Sistemas de memoria distribuida, clusters y cloud.
  - 1.4. 1.4 Coprocesadores (GPUs y MIC)
  - 1.5. 1.5 Análisis de aplicaciones. Detección de errores y cuellos de botella (valgrind)
2. Tema 2: Programación paralela
  - 2.1. 2.1 Programación de sistemas de memoria compartida: OpenMP. Ejemplos
3. Tema 3: Programación distribuida
  - 3.1. 3.1 Programación de sistemas de memoria distribuida: MPI. Ejemplos
4. Tema 4: E/S de Alto Rendimiento
  - 4.1. 4.1 Computación Intensiva de datos. Motivación.
  - 4.2. 4.2 Sistemas de ficheros paralelos y escalables. Lustre, GPFS, PVFS, HDFS, Ceph, GlusterFS.
  - 4.3. 4.3 Bibliotecas de E/S paralela. MPI-IO, HDF5, NetCDF
  - 4.4. 4.4 Ejemplos de aplicación
5. Tema 5: Modelos de programación para BigData
  - 5.1. 5.1 MapReduce: Google MapReduce y Hadoop MapReduce
  - 5.2. 5.2 Modelos basados en BSP: Genéricos: Apache Hama
  - 5.3. 5.3 Modelos basados en BSP: Para el procesamiento de grafos: Google Pregel y Apache Giraph
6. Aceleradores
  - 6.1. 6.1 Many Integrated Cores o Intel Xeon Phi. Descripción. Vectorización
  - 6.2. 6.2 Graphic Processing Unit. Introducción. Descripción de la arquitectura. Programación con Cuda

## Cronograma

**Horas totales:** 48 horas

**Horas presenciales:** 45 horas (38.5%)

**Peso total de actividades de evaluación continua:**  
100%

**Peso total de actividades de evaluación sólo prueba final:**  
100%

Semana	Actividad Presencial en Aula	Actividad Presencial en Laboratorio	Otra Actividad Presencial	Actividades Evaluación
Semana 1	<b>Explicación de contenidos del Tema 1</b> Duración: 03:00 LM: Actividad del tipo Lección Magistral			
Semana 2	<b>Explicación de contenidos del Tema 2</b> Duración: 02:00 LM: Actividad del tipo Lección Magistral	<b>Ejercicio de tema 1. Valgrind</b> Duración: 01:00 PL: Actividad del tipo Prácticas de Laboratorio		
Semana 3	<b>Explicación de contenidos del Tema 3</b> Duración: 01:00 LM: Actividad del tipo Lección Magistral	<b>Ejercicio de tema 2. OpenMP</b> Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		
Semana 4	<b>Explicación de contenidos del Tema 3</b> Duración: 01:00 LM: Actividad del tipo Lección Magistral	<b>Ejercicio de tema 3. MPI</b> Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		
Semana 5	<b>Explicación de contenidos del Tema 4</b> Duración: 03:00 LM: Actividad del tipo Lección Magistral			
Semana 6	<b>Explicación de contenidos del Tema 4</b> Duración: 03:00 LM: Actividad del tipo Lección Magistral			
Semana 7	<b>Explicación de contenidos del Tema 4</b> Duración: 02:00 LM: Actividad del tipo Lección Magistral			<b>Entrega del proyecto de programación de E/S paralela</b> Duración: 01:00 TI: Técnica del tipo Trabajo Individual Evaluación continua Actividad no presencial
Semana 8	<b>Explicación de contenidos del Tema 5</b> Duración: 03:00 LM: Actividad del tipo Lección Magistral			
Semana 9	<b>Explicación de contenidos del Tema 5</b> Duración: 03:00 LM: Actividad del tipo Lección Magistral			
Semana 10	<b>Explicación de contenidos del Tema 5</b> Duración: 02:00 LM: Actividad del tipo Lección Magistral			<b>Entrega del proyecto de Big Data</b> Duración: 01:00 TI: Técnica del tipo Trabajo Individual Evaluación continua Actividad no presencial

Semana 11	<b>Explicación de contenidos del Tema 6</b> Duración: 02:00 LM: Actividad del tipo Lección Magistral	<b>Ejercicio de tema 6. MIC</b> Duración: 01:00 PL: Actividad del tipo Prácticas de Laboratorio		
Semana 12	<b>Explicación de contenidos del Tema 6</b> Duración: 02:00 LM: Actividad del tipo Lección Magistral	<b>Ejercicio de tema 6. MIC</b> Duración: 01:00 PL: Actividad del tipo Prácticas de Laboratorio		
Semana 13	<b>Explicación de contenidos del Tema 6</b> Duración: 02:00 LM: Actividad del tipo Lección Magistral	<b>Ejercicio de tema 6. GPU</b> Duración: 01:00 PL: Actividad del tipo Prácticas de Laboratorio		
Semana 14		<b>Ejercicio de tema 6. GPU</b> Duración: 03:00 PL: Actividad del tipo Prácticas de Laboratorio		
Semana 15		<b>Caso de estudio</b> Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		<b>Entrega de la práctica de programación de computación de alto rendimiento</b> Duración: 01:00 TI: Técnica del tipo Trabajo Individual Evaluación continua Actividad no presencial
Semana 16		<b>Caso de estudio</b> Duración: 03:00 PL: Actividad del tipo Prácticas de Laboratorio		
Semana 17				<b>Los que no hayan aprobado por curso tendrá un plazo extraordinario para volver a entregar los ejercicios prácticos que no ha superado satisfactoriamente</b> Duración: 01:00 TI: Técnica del tipo Trabajo Individual Evaluación sólo prueba final Actividad no presencial

**Nota.-** El cronograma sigue una planificación teórica de la asignatura que puede sufrir modificaciones durante el curso.

**Nota 2.-** Para poder calcular correctamente la dedicación de un alumno, la duración de las actividades que se repiten en el tiempo (por ejemplo, subgrupos de prácticas") únicamente se indican la primera vez que se definen.

## Actividades de Evaluación

Semana	Descripción	Duración	Tipo evaluación	Técnica evaluativa	Presencial	Peso	Nota mínima	Competencias evaluadas
7	Entrega del proyecto de programación de E/S paralela	01:00	Evaluación continua	TI: Técnica del tipo Trabajo Individual	No	25%	2 / 10	CG12, CE1, CB7, CE16, CB10
10	Entrega del proyecto de Big Data	01:00	Evaluación continua	TI: Técnica del tipo Trabajo Individual	No	25%	2 / 10	CE1, CB7, CE16, CG12, CB10
15	Entrega de la práctica de programación de computación de alto rendimiento	01:00	Evaluación continua	TI: Técnica del tipo Trabajo Individual	No	50%	2 / 10	CG12, CE1, CB7, CE16, CB10, CE4, CG9, CE10
17	Los que no hayan aprobado por curso tendrán un plazo extraordinario para volver a entregar los ejercicios prácticos que no ha superado satisfactoriamente	01:00	Evaluación sólo prueba final	TI: Técnica del tipo Trabajo Individual	No	100%	5 / 10	CG12, CE1, CB7, CE16, CB10, CE4, CG9, CE10

## Criterios de Evaluación

### Sistema de evaluación

La asignatura se evaluará mediante la realización de una serie de proyectos prácticos y ejercicios. El enunciado de los mismos se presentará en clase, en la propia aula y durante el horario regular de la asignatura, en las fechas especificadas en el calendario de la asignatura. El desarrollo de estos proyectos se llevará a cabo tanto de forma presencial en las clases prácticas en el laboratorio, como no presencial usando los recursos ofrecidos por el Centro de Cálculo para tal fin, y apoyándose en las tutorías para la resolución de cualquier aspecto vinculado con el desarrollo de los mismos. Los plazos de entrega estarán escalonados a lo largo del desarrollo del curso y se publicarán conjuntamente con el enunciado con suficiente antelación.

La nota final de la asignatura se calculará considerando un peso de 25% para el proyecto práctico de E/S paralela, otro 25% para el proyecto de BigData y otro 50% para los trabajos prácticos de computación de alto rendimiento. Para aprobar la asignatura, además de tener una nota final mayor o igual a 5, habrá que tener una nota mínima de 2 en cada una de las tres partes.

### Evaluación en la convocatoria extraordinaria de julio

En caso de que el alumno no haya superado la asignatura deberá presentarse al examen de esta convocatoria. El alumno tendrá un plazo extraordinario para volver a entregar los ejercicios prácticos que no ha superado satisfactoriamente, plazo que finaliza el día anterior al de la celebración del examen.

### Plazos de entrega

Las fechas exactas establecidas como plazo máximo para la entrega de las distintas prácticas serán publicadas en el enunciado de cada práctica. Los plazos de entrega estarán escalonados a lo largo del curso con el objetivo de facilitar el desarrollo de las prácticas en paralelo con el temario y maximizar así su beneficio. En ningún caso se admitirán prácticas entregadas fuera de plazo.



## Recursos Didácticos

Descripción	Tipo	Observaciones
Hennessy, J. L, Patterson, D. A.; "Computer Architecture: A quantitative Approach", 5th. Ed.. Morgan Kauffmann Pub., 2012	Bibliografía	Libro
Data-intensive Text Processing with MapReduce. Jimmy Lin, Chris Dyer, Morgan & Claypool Publishers, 2010	Bibliografía	Articulo
Ceph: A Scalable, High-Performance Distributed File System Sage Weil, Scott A. Brandt, Ethan L. Miller, Darrell D. E. Long, Carlos Maltzahn Proceedings of the 7th Conference on Operating Systems Design and Implementation (OSDI 06) November 2006	Bibliografía	Articulo
Overview Of The MPI-IO Parallel I/O Interface (1995) Peter Corbett , Dror Feitelson , Sam Fineberg , Yarsun Hsu , Bill Nitzberg , Jean-Pierre Prost , Marc Snir , Bernard Traversat , Parkson Wong	Bibliografía	Libro
HDFS, <a href="http://hadoop.apache.org/docs/r1.2.1/hdfs_design.html">http://hadoop.apache.org/docs/r1.2.1/hdfs_design.html</a>	Bibliografía	Web
Página web de la asignatura ( <a href="http://www.datsi.fi.upm.es/docencia/PPD">http:// www.datsi.fi.upm.es/docencia/PPD</a> )	Recursos web	Pagina web asignatura