

## Computación de Altas Prestaciones

Curso Académico: ( 2016 / 2017 )

Fecha de revisión: 11/05/2016 18:38:30

Departamento asignado a la asignatura:

Coordinador/a: GARCIA BLAS, FRANCISCO JAVIER

Tipo: Obligatoria Créditos ECTS : 3.0

Curso : 2 Cuatrimestre : 1

## REQUISITOS (ASIGNATURAS O MATERIAS CUYO CONOCIMIENTO SE PRESUPONE)

Arquitectura de Computadores, Sistemas Operativos

## OBJETIVOS

1. Capacidad para comprender y poder aplicar conocimientos avanzados de computación de altas prestaciones y métodos numéricos o computacionales a problemas de ingeniería.
2. Capacidad para analizar y diseñar aplicaciones de computación de altas prestaciones.
3. Ser capaz de aplicar técnicas de computación de altas prestaciones a problemas de ingeniería.

## DESCRIPCIÓN DE CONTENIDOS: PROGRAMA

1. Principios de la computación de altas prestaciones.
2. Diseño y análisis de aplicaciones de altas prestaciones.
3. Paradigmas de programación paralela: paso de mensaje, memoria compartida, paralelismo de datos.
4. Análisis del rendimiento, evaluación y optimización de aplicaciones.
5. Nuevas tendencias en altas prestaciones.

## ACTIVIDADES FORMATIVAS, METODOLOGÍA A UTILIZAR Y RÉGIMEN DE TUTORÍAS

1. Clases magistrales. Orientadas a la enseñanza de las competencias específicas de la materia. En ellas se presentarán los conocimientos que los alumnos deben adquirir. Para facilitar su desarrollo los alumnos recibirán las notas de clase y tendrán textos básicos de referencia que les permita completar y profundizar en aquellos temas en los cuales estén más interesados.
2. Prácticas. Dentro de esta materia se llevarán a cabo prácticas, que serán preferentemente en grupo, que estarán relacionadas con la computación de altas prestaciones
3. Trabajo personal y estudio del alumno. Orientado especialmente a la adquisición de la Capacidad para la autoorganización y planificación del trabajo individual y del proceso de aprendizaje. Puede incluir, entre otros ejercicios y lecturas complementarias, así como el estudio personal por parte del estudiante.

## SISTEMA DE EVALUACIÓN

|                                                    |    |
|----------------------------------------------------|----|
| <b>Peso porcentual del Examen/Prueba Final:</b>    | 50 |
| <b>Peso porcentual del resto de la evaluación:</b> | 50 |

La evaluación tiene como misión conocer el grado de cumplimiento de los objetivos de aprendizaje, por ello se valorará todo el trabajo del alumno, individual o colectivamente, mediante la evaluación continua de sus actividades a través de los ejercicios y exámenes, trabajos prácticos y otras actividades académicas formativas descritas anteriormente.

Se realizará una evaluación de forma continua. El 50% de la nota se obtendrá mediante evaluación continua y el 50% restante mediante un examen final. El proceso de evaluación continua consta de:

- Tres prácticas de programación, que contarán el 50 % de la nota.

|                                                    |    |
|----------------------------------------------------|----|
| <b>Peso porcentual del Examen/Prueba Final:</b>    | 50 |
| <b>Peso porcentual del resto de la evaluación:</b> | 50 |

Para poder ser evaluados según los principios de la evaluación continua un alumno debe:

- Entregar todas las prácticas de programación
- Obtener una calificación mínima de 3 puntos en cada una de las prácticas.

El examen final abarcará todos los contenidos de la asignatura. La nota mínima para aprobar el examen final (ordinario o extraordinario) es 4 en una escala de 0 a 10.

La nota mínima para aprobar la asignatura es de 5 en una escala de 0 a 10.

#### BIBLIOGRAFÍA BÁSICA

- B. Wilkinson, M. Allen Parallel Programming Techniques and Applications Using Networked Workstations and Parallel Computers, Prentice Hall, 1999.
- Bradford Nichols, Dick Buttlar, Jacqueline Proulx Farrell PThreads Programming, O'Reilly, 1997.
- D. Kirk, W. Hwo Programming Massively Parallel Processors: A Hands-on Approach, Morgan Kaufmann, 2010
- D. Sima, T. Fountain, P. Kacsuk Advanced Computer Architectures: a Design Space Approach, Addison Wesley, 1997..
- D.E. Culler, J.P. Singh, with A. Gupta Parallel Computer Architectures: a Hardware/Software Approach, Morgan Kaufmann Pub.,1999.
- G.R. Andrews Multithreaded, Parallel, and Distributed Programming, Addison Wesley, 2000..
- P.S. Pacheco Parallel Programming with MPI, Morgan Kaufmann Pub.,1997..
- R. Chandra, R. Menon,L. Dagum et al. Parallel Programming in OpenMP, Morgan Kaufmann Pub., 2000..