



Universitat de Lleida

GUÍA DOCENTE
**COMPUTACIÓN DE ALTAS
PRESTACIONES**

Coordinación: ONRUBIA PALACIOS, JORDI RICARD

Año académico 2023-24

Información general de la asignatura

Denominación	COMPUTACIÓN DE ALTAS PRESTACIONES			
Código	103084			
Semestre de impartición	2o Q(SEMESTRE) EVALUACIÓN CONTINUADA			
Carácter	Grado/Máster	Curso	Carácter	Modalidad
	Máster Universitario en Ingeniería Informática	1	OBLIGATORIA	Presencial
Número de créditos de la asignatura (ECTS)	4.5			
Tipo de actividad, créditos y grupos	Tipo de actividad	PRALAB	TEORIA	
	Número de créditos	3	1.5	
	Número de grupos	1	1	
Coordinación	ONRUBIA PALACIOS, JORDI RICARD			
Departamento/s	INGENIERÍA INFORMÁTICA Y DISEÑO DIGITAL			
Distribución carga docente entre la clase presencial y el trabajo autónomo del estudiante	30% de trabajo dirigidas con el profesor y 70% de trabajo autónomo. Las horas de trabajo dirigido con el profesor se realizarán en sesiones presenciales .			
Información importante sobre tratamiento de datos	Consulte este enlace para obtener más información.			
Idioma/es de impartición	Inglés			

Profesor/a (es/as)	Dirección electrónica\nprofesor/a (es/as)	Créditos impartidos por el profesorado	Horario de tutoría/lugar
GINE DE SOLA, FRANCESC	francesc.gine@udl.cat	0	
ONRUBIA PALACIOS, JORDI RICARD	jordi.onrubia@udl.cat	4,5	

Información complementaria de la asignatura

Para cursar la asignatura se deben tener conocimientos sólidos de programación estructurada en lenguaje C, así como de Arquitectura y Tecnología de Computadores y Sistemas Operativos.

Objetivos académicos de la asignatura

- Analizar las prestaciones de un computador HPC.
- Saber y entender los paradigmas de programación paralela.
- Conocer el funcionamiento y características de las arquitecturas HPC.
- Implementar y depurar aplicaciones paralelas utilizando los modelos de programación OpenMP, MPI y CUDA.
- Resolver algoritmos numéricos utilizando tecnologías híbridas OpenMP-MPI y OpenMP-CUDA y evaluar su rendimiento.
- Comunicar fluidamente ideas y conceptos en inglés escrito y hablado.

Competencias

Competencias estratégicas de la Universidad de Lleida

- UdL1: Corrección en la expresión oral y escrita
- UdL2: Dominio de una lengua extranjera

Competencias Específicas EPS

- EPS3: Capacidad de transmitir información, ideas, problemas y soluciones a un público tanto especializado como no especializado
- EPS4: Capacidad de concebir, diseñar e implementar proyectos y/o aportar soluciones novedosas, utilizando herramientas propias de la ingeniería

Competencias Generales

- 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 informática.
- CG8: Capacidad para la aplicación de los conocimientos adquiridos y la resolución de problemas en entornos nuevos o poco conocidos dentro de contextos más amplios y multidisciplinares, siendo capaces de la integración de estos conocimientos.

Competencias Básicas

- CB1: Poseer y comprender conocimientos que aporten una base u oportunidad de ser originales en el desarrollo y/o aplicación de ideas, a menudo en un contexto de investigación.
- CB2: Saber aplicar los conocimientos adquiridos y tener capacidad de resolución de problemas en entornos nuevos o poco conocidos dentro de contextos más amplios (o multidisciplinares) relacionados con su área de estudio.

Competencias específicas de la titulación

- CE10: Capacidad para comprender y poder aplicar conocimientos avanzados de computación de altas prestaciones y métodos numéricos o computacionales a problemas de ingeniería
- CE12: Capacidad para aplicar métodos matemáticos, estadísticos y de inteligencia artificial para modelar, diseñar y desarrollar aplicaciones, servicios, sistemas inteligentes y sistemas basados en el conocimiento.

Contenidos fundamentales de la asignatura

1. Introducción: Computación de Altas Prestaciones (1P + 2h A)
2. Procesamiento paralelo y benchmarking (5h P+13h A)
 - o Introducción
 - o Arquitecturas HPC
 - o Diseño de aplicaciones paralelas
 - o Métricas de rendimiento
 - o Herramientas de benchmarking
3. Programación paralela en entornos de memoria compartida - OPenMP (8h P+21h A)
 - o Introducción a OpenMP
 - o Regiones paralelas y series
 - o Variables de entorno
 - o Estrategias de descomposición de bucles
 - o Sincronización de tareas
 - o Modelos de memoria
4. Programación paralela en entornos de memoria distribuida - MPI (8h + 21h A)
 - o Introducción a MPI
 - o Estructura básica de un programa MPI
 - o Tipos de comunicaciones
 - o Tipos de datos derivadas
 - o Comunicadores i topologías
5. Programación paralela en entornos GPGPU - CUDA (8h P+ 21h A)
 - o Introducción a CUDA
 - o Entorno de ejecución de CUDA
 - o Administración de memoria

- o Threads, blocks e indexing
- o Comunicaciones y sincronización
- o Administración de dispositivos

P: Horas de clases presenciales.

A: Horas de trabajo autónomo.

Ejes metodológicos de la asignatura

Cada semana el estudiante recibirá:

- Dos horas de clase dirigidas por el profesor en el laboratorio, donde se explicarán tanto los contenidos más teóricos, acompañados de ejemplos ilustrativos, como los más aplicados, reforzados con prácticas ilustrativas dirigidas. Como material de apoyo de la clase se seguirán las transparencias de la asignatura. Asimismo, se dispone de material de apoyo para hacer un seguimiento no presencial que permitan reforzar temas básicos.
- El estudiante dispondrá de material de soporte para poder realizar un seguimiento no presencial de la asignatura.

La evaluación será continua y está integrada por cinco diferentes pruebas:

- 4 prácticas: Benchmarking, OpenMP, MPI-OpenMP y CUDA.
- Verificación mediante la presentación oral de la práctica híbrida (OpenMP-MPI) y de CUDA.

Plan de desarrollo de la asignatura

Esta planificación tiene un carácter orientativo y está sujeta a posibles cambios en función de la dinámica del curso.

Semana	Descripción	Actividades dirigidas	Actividades no dirigidas	Horas de trabajo autónomo
1	Introducción: Computación de Altas Prestaciones	Presentación de la asignatura Clase magistral	Repaso de los contenidos explicados	1
	Procesamiento paralelo y benchmarking	Clase magistral	Resolución de ejercicios	1
2	Procesamiento paralelo y benchmarking	Clase magistral y laboratorio Presentación entorno gestor colas SGE	Resolución de ejercicios Conocimiento herramientas del entorno del gestor de colas SGE	6
3	Procesamiento paralelo y benchmarking	Clase magistral y laboratorio	Resolución de ejercicios Práctica de benchmarking	6
4	Programación paralela en entornos de memoria compartida - OpenMP	Clase magistral y laboratorio	Resolución de ejercicios OpenMP Práctica de benchmarking	5

COMPUTACIÓN DE ALTAS PRESTACIONES 2023-24

5	Programación paralela en entornos de memoria compartida - OpenMP	Clase magistral y laboratorio	Resolución de ejercicios OpenMP Práctica HPC-OpenMP Entrega de práctica de benchmarking	5
6	Programación paralela en entornos de memoria compartida - OpenMP	Clase magistral y laboratorio	Resolución de ejercicios OpenMP Práctica HPC-OpenMP	5
7	Programación paralela en entornos de memoria compartida - OpenMP	Clase magistral y laboratorio	Resolución de ejercicios OpenMP Práctica HPC-OpenMP	6
8	Programación paralela en entornos de memoria distribuida - MPI	Clase magistral y laboratorio	Resolución de ejercicios MPI Práctica HPC-OpenMP	5
9	Programación paralela en entornos de memoria distribuida - MPI	Clase magistral y laboratorio	Resolución de ejercicios MPI Práctica HPC-OpenMP-MPI Entrega práctica HPC-OpenMP	5
10	Programación paralela en entornos de memoria distribuida - MPI	Clase magistral y laboratorio	Resolución ejercicios MPI Práctica HPC-OpenMP-MPI	5
11	Programación paralela en entornos de memoria distribuida - MPI	Clase magistral y laboratorio	Resolución ejercicios MPI Práctica HPC-OpenMP-MPI	6
12	Programación paralela en entornos GPGPU -CUDA	Clase magistral y laboratorio	Resolución ejercicios CUDA Práctica HPC-OpenMP-MPI	5
13	Programación paralela en entornos GPGPU -CUDA	Clase magistral y laboratorio	Resolución ejercicios CUDA Práctica HPC-CUDA Entrega práctica HPC-OpenMP-MPI	5
14	Programación paralela en entornos GPGPU - CUDA	Clase magistral y laboratorio	Resolución ejercicios CUDA Práctica HPC-CUDA	5
15	Programación paralela en entornos GPGPU - CUDA	Clase magistral y laboratorio	Resolución ejercicios CUDA Práctica HPC-CUDA	6
16	Semana exámenes	Tutoría	Práctica HPC-CUDA	2
17	Semana exámenes	Tutoría	Entrega práctica HPC-CUDA	2
18	Semana exámenes+verificación de las prácticas	Tutoría	Verificación de la práctica híbrida y de CUDA.	3
19	Verificación de las prácticas	Presentación Oral	Verificación de la práctica híbrida y de CUDA.	3

Sistema de evaluación

La **evaluación es continuada** y está formada por cinco bloques de evaluación con los siguientes pesos respecto de la nota final de la asignatura:

- Bloque Benchmarking: 10%
- Bloque computación memoria compartida: 25%
- Bloque computación memoria distribuida 35%
- Bloque GPGPU: 30%

Cada bloque está formado por las actividades de evaluación que se muestran en la siguiente Tabla.

Bloque Evaluación	O/V (1)	Actividades Evaluación	Pesos	I/G (2)
Bloque benchmarking	O	Práctica Benchmarking	10%	G
Bloque computación memoria compartida	O	Práctica HPC-OpenMP	25%	G
Bloque computación memoria distribuida	O	Práctica Híbrida HPC-OpenMP-MPI Verificación de la práctica	35%	G
Bloque GPGPU	O	Practica HPC-CUDA Verificación de la práctica	30%	G

(1) Obligatorio / Voluntario

(2) Individual / Grupo

La nota final se calculará de la siguiente manera:

Nota Final= 10% Benchmarking + 25% OpenMP + 35% OpenMP-MPI (práctica híbrida)*verificación +30% CUDA*verificación

Los estudiantes que participen de manera activa en la asignatura, realizando las actividades extras propuestas a lo largo del curso, podrán optar a un punto adicional sobre la Nota Final de la asignatura.

La prueba de verificación consiste en una presentación oral al profesor, donde cada grupo ha de explicar al profesor las principales decisiones de diseño tomadas durante la realización de la práctica híbrida y de CUDA, así como una demostración de su correcta ejecución. Esta presentación se realizará al final del semestre en el laboratorio de prácticas. La nota de la prueba de verificación será binaria : 1(Pasada) o 0 (No pasada).

Los estudiantes que no superen la Nota Final con una puntuación igual o superior al 5 podrán realizar actividades de recuperación de las actividades con un peso superior al 20%.

El estudiante que cuente con la aprobación para cursar la evaluación alternativa ([ver normativa de evaluación de la UdL](#)) tendrá que realizar las siguientes actividades de evaluación

- Práctica HPC-OpenMP: 25%
- Práctica HPC-MPI-OpenMP (Híbrida) y su posterior verificación realizadas en las mismas condiciones explicadas en la evaluación continuada: 40%
- Práctica HPC-CUDA y su posterior verificación realizada en las mismas condiciones explicadas en la evaluación continuada:: 35%

El estudiante que no supere la evaluación alternativa con una nota igual o superior a 5 tendrá derecho a la recuperación de las actividades con un peso superior al 20%.

Bibliografía y recursos de información

F. Giné & J.L. Lérida

Transparencias de la Asignatura

Lleida, 2018

P.S. Pacheco,

Parallel Programming with MPI,

Morgan Kaufmann Publishers , 1997

R. Chandra, L. Dagum, D. Kohr,

Parallel Programming in OpenMP,

Morgan Kaufmann Publishers , 2001

S. Duane, Y. Mete

CUDA for Engineers: An Introduction to High-Performance Parallel Computing,

Addison Wesley, 2015

J. Cheng, M. Grossman, T. McKercher

PROFESSIONAL CUDA® C Programming,

John Wiley & Sons, Inc, 2014

<https://www.cs.utexas.edu/~rossbach/cs380p/papers/cuda-programming.pdf>