

Titulación	Tipo	Curso
Investigación e Innovación en Ciencia e Ingeniería Basadas en Computadores	OP	1

Contacto

Nombre: Juan Carlos Moure Lopez

Correo electrónico: juancarlos.moure@uab.cat

Equipo docente

Nehir Sonmez Tekin

Idiomas de los grupos

Puede consultar esta información al [final](#) del documento.

Prerrequisitos

Conocimientos de programación (preferiblemente en lenguaje C)

Objetivos y contextualización

El objetivo fundamental de la asignatura es que el alumnado adquiera las capacidades de análisis, uso y evaluación de los ordenadores, para el desarrollo de aplicaciones avanzadas.

Los conceptos básicos que se expondrán en las sesiones teóricas son: procesamiento paralelo en ordenadores, memorias jerárquicas, evaluación del rendimiento y los lenguajes de programación paralela y modelos de estos sistemas. Los objetivos más específicos son los siguientes:

1. Entender las claves para obtener buen rendimiento de computadores multi-núcleo y multi-procesador, y de aceleradores de cómputo (GPUs)
2. Identificar las oportunidades de paralelización en un algoritmo o aplicación, a diferentes niveles (instrucciones, iteraciones de un bucle, llamadas a funciones...), y expresarlo formalmente
3. Diseñar las estructuras de datos de una aplicación y los algoritmos que acceden a estos datos para favorecer el buen rendimiento de la jerarquía de memoria
4. Analizar, diseñar e implantar algoritmos paralelos con patrones abstractos de cómputo (reducción, transformación...) bajo paradigmas de programación basados en variables compartidas y sincronización; y con lenguajes paralelos actuales.

5. Verificar la funcionalidad y evaluar el rendimiento de aplicaciones paralelas, identificando los cuellos de botella respecto al rendimiento
6. A partir del análisis de rendimiento, seleccionar el computador adecuado a una aplicación y/o realizar optimizaciones del código que mejoren aún más su rendimiento

Resultados de aprendizaje

1. CA04 (Competencia) Capacidad para aplicar técnicas de computación de altas prestaciones a problemas complejos de ámbitos diversos de la ciencia o la ingeniería.
2. CA05 (Competencia) Capacidad para generar preguntas e hipótesis ante nuevos retos de investigación y demostrar originalidad en la forma de abordar y resolver los problemas planteados mediante técnicas de computación de altas prestaciones.
3. KA06 (Conocimiento) Describir los patrones algorítmicos más adecuados para diferentes entornos de computación de altas prestaciones.
4. KA07 (Conocimiento) Describir los problemas de rendimiento más comunes que pueden exhibir las aplicaciones informáticas en función de la arquitectura del sistema donde sean ejecutadas.
5. KA08 (Conocimiento) Identificar en base a criterios de coste-prestaciones las pilas tecnológicas más adecuadas para desarrollar aplicaciones de altas prestaciones que aborden problemas en ámbitos de ciencia e ingeniería.
6. SA08 (Habilidad) Ser capaz de identificar los problemas de rendimiento de aplicaciones complejas seleccionando y aplicando las técnicas de perfilado y las herramientas adecuadas.
7. SA09 (Habilidad) Ser capaz de optimizar aplicaciones adaptadas a diferentes arquitecturas: sistemas multinúcleo, sistemas distribuidos, aceleradoras gráficas (GPUs) y sistemas híbridos.
8. SA10 (Habilidad) Ser capaz de diseñar aplicaciones innovadoras en ámbitos de ciencia e ingeniería aplicando modelos teóricos y usando técnicas y herramientas propias de la computación de altas prestaciones.

Contenido

1. Paralelismo de Instrucciones: Dependencias de datos entre instrucciones en un bucle y reordenación dinámica de la ejecución
2. Paralelismo de Datos: Análisis del paralelismo de datos de un algoritmo y ejecución SIMD (vectorización)
3. Paralelismo de Tareas: Ejecución multi-hilo y multi-núcleo; Jerarquía de memoria compartida y coherencia de datos; Sincronización entre *threads*
4. Algoritmos Paralelos: Modelo de paralelismo con variables compartidas y sincronización. Patrones de cómputo paralelo: reducción, transformación, *master/worker*, *divide & conquer* y *pipeline*
5. Evaluación del Rendimiento de Aplicaciones: Análisis de complejidad, paralelismo y localidad; Métricas de rendimiento (*elapsed time*, IPC, ancho de banda e intensidad aritmética); Trabajo total y camino crítico de la ejecución paralela

Actividades formativas y Metodología

Título	Horas	ECTS	Resultados de aprendizaje
Tipo: Dirigidas			

Asistencia a clase de teoría y ejercicios teóricos	26	1,04
Prácticas en Laboratorio	16	0,64
Tipo: Supervisadas		
Trabajo de Investigación	15	0,6
Tipo: Autónomas		
Estudio autónomo, redacción de informe y desarrollo de presentación	41	1,64
Estudio y Resolución de Problemas	50	2

Clases de Teoría: se expondrán los conocimientos propios de la asignatura ilustrados con ejemplos prácticos. Se resaltarán los problemas de aprendizaje más importantes y se mostrará cómo completar y profundizar en los contenidos. Se discutirán y resolverán problemas prácticos y el profesor detectará los problemas de comprensión y razonamiento más comunes y los resolverá para todos los estudiantes. Se promueve la capacidad de análisis y síntesis, el razonamiento crítico y la capacidad de resolución de problemas.

Caso práctico de estudio autónomo: durante el curso, el alumnado debe diseñar, planificar, realizar, presentar y defender de forma oral un trabajo práctico, indicando los objetivos, el desarrollo, los resultados obtenidos con esquemas, figuras y gráficas comprensibles, las incidencias más notables y, finalmente, las conclusiones más relevantes. Se espera una actitud proactiva y dinámica, la búsqueda autónoma de los conocimientos que sean necesarios, y ambición con los objetivos del trabajo.

Nota: se reservarán 15 minutos de una clase dentro del calendario establecido por el centro o por la titulación para que el alumnado rellene las encuestas de evaluación de la actuación del profesorado y de evaluación de la asignatura o módulo.

Evaluación

Actividades de evaluación continuada

Título	Peso	Horas	ECTS	Resultados de aprendizaje
Ejercicios de Laboratorio: Programación y Evaluación de Rendimiento	50	0	0	CA04, CA05, KA07, SA08, SA09, SA10
Examen Individual	20	1,5	0,06	KA06, KA07, KA08
Informe escrito y Defensa oral del trabajo autónomo	30	0,5	0,02	CA05, KA06, KA07, KA08

Las fechas de evaluación continua y entrega de trabajos se publicarán en la plataforma Moodle de la UAB (Campus Virtual) y pueden estar sujetas a cambios de programación por motivos de adaptación a posibles incidencias. Siempre se informará en la plataforma Moodle sobre estos cambios ya que se entiende que es la plataforma habitual de intercambio de información entre profesor y estudiantes.

Actividades de evaluación

Actividad A: examen escrito de teoría y problemas; realización individual y sin apuntes; 20% de la nota final; no hay nota mínima; recuperable; a final de curso

Actividad B: ejercicios de laboratorio; realización en grupo; 50% de la nota final; nota mínima de 5 sobre 10; no es recuperable; durante todo el curso

Actividad C: informe escrito y defensa oral del trabajo autónomo; realización individual; 30% de la nota final; no hay nota mínima; recuperable; a final de curso

Proceso de Recuperación

El/la estudiante se puede presentar a la recuperación siempre que haya realizado las actividades A, B y C

Irregularidades por parte de los estudiantes, copia y plagio

Sin perjuicio de otras medidas disciplinarias que se estimen oportunas, se calificarán con un cero las irregularidades cometidas por un estudiante que puedan conducir a una variación de la calificación de un acto de evaluación. Por tanto, la copia, el plagio, el engaño, dejar copiar, el uso no autorizado de la IA (p. ej., Copilot, ChatGPT o equivalentes), etc. en cualquiera de las actividades de evaluación implicará suspenderla con un cero. Las actividades de evaluación calificadas de esta forma y por este procedimiento no serán recuperables. Si es necesario superar cualquiera de estas actividades de evaluación para aprobar la asignatura, entonces esta asignatura quedará suspendida directamente, sin oportunidad de recuperarla en el mismo curso. En el caso de no superar la asignatura debido a que el estudiante ha cometido irregularidades en un acto de evaluación, la nota numérica del expediente será el valor menor entre 3.0 y la media ponderada de las notas (y por tanto no será posible el aprobado por compensación).

Bibliografía

Parallel Programming: Techniques and Applications using Networked Workstations and Parallel Computers. Barry Wilkinson. Prentice Hall, 1999.

Designing and Building Parallel Programs: Concepts and Tools for Parallel Software Engineering. Ian Foster. Addison Wesley, 1995.

Introduction to Parallel Computing. A. Grama et al. Addison Wesley, Second Edition, 2003.

Computer Architecture: A Quantitative Approach. 5th Edition (

https://bibcercador.uab.cat/permalink/34CSUC_UAB/15r2rl8/cdi_askewsholts_vlebooks_9780123838735) John Hennessy, David Patterson, Morgan Kaufmann (Elsevier) 2018 (Cap. 4 i 5) Structured Parallel Programming: Patterns for efficient computation M. McCool, J. Reinders, A. Robison, Elsevier, 2012

Parallel Programming for Multicore and Cluster Systems T. Rauber, G. Rünger, Springer (Elsevier), 2010

Programming Massively Parallel Processors: A Hands-on Approach D. Kirk, & W.M. Hwu, Morgan Kaufmann (Elsevier), 2010

Software

C language. OpenMP and OpenACC extensions.

Grupos e idiomas de la asignatura

La información proporcionada es provisional hasta el 30 de noviembre de 2025. A partir de esta fecha, podrá consultar el idioma de cada grupo a través de este [enlace](#). Para acceder a la información, será necesario introducir el CÓDIGO de la asignatura

Nombre	Grupo	Idioma	Semestre	Turno
(PLABm) Prácticas de laboratorio (máster)	1	Inglés	primer cuatrimestre	tarde
(TEm) Teoría (máster)	1	Inglés	primer cuatrimestre	tarde