

Arquitectures Avançades**2014/2015**

Código: 102778

Créditos ECTS: 6

Titulació	Tipo	Curso	Semestre
2502441 Enginyeria Informàtica	OT	4	1

Contacto

Nombre: Juan Carlos Moure Lopez

Correo electrónico: JuanCarlos.Moure@uab.cat

Uso de idiomas

Lengua vehicular mayoritaria: espanyol (spa)

Algún grupo íntegramente en inglés: No

Algún grupo íntegramente en catalán: No

Algún grupo íntegramente en español: No

Equipo docente

Miquel Àngel Senar Rosell

Alejandro Chacon de San Baldomero

Objetivos y contextualización

El objetivo fundamental de la asignatura es que los alumnos adquieran las capacidades de análisis, utilización y evaluación de arquitecturas avanzadas de computadores, para el desarrollo de aplicaciones y servicios informáticos.

Los conceptos básicos que se describirán en las sesiones teóricas son: el procesador multi-núcleo, el sistema multi-procesador de memoria compartida, la evaluación de rendimiento y prestaciones, y los modelos y lenguajes de programación de estos sistemas paralelos.

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, como OpenMP, Cilk y CUDA/Thrust.
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

Contenido

1. Paralelismo de Instrucciones

- Ejecución pipeline y múltiple
- Reordenación dinámica de la ejecución de las instrucciones
- Dependencias de datos entre instrucciones en un bucle

2. Paralelismo de Datos

- Análisis del paralelismo de datos de un algoritmo
- Paralelismo SIMD (vectorización) y MIMD (multi-thread)
- Arquitecturas Vectoriales e instrucciones SIMD
- GPUs y aceleradores de cómputo (Xeon Phi)

3. Arquitectura de computadores paralelos

- Estructura y arquitectura de los procesadores multi-hilo y multi-núcleo, y de los nodos multi-procesador
- Jerarquía de memoria (NUMA) y soporte para la coherencia de datos en cache
- Soporte H/W para la sincronización y para la comunicación entre threads

4. Algoritmos Paralelos

- Modelo de paralelismo con variables compartidas y sincronización. Patrones de cómputo paralelo
- Paralelismo de datos (SIMD): operaciones de reducción y de transformación sobre estructuras multi-dimensionales
- Paralelismo de tareas (MIMD): master/worker, divide & conquer, pipeline
- Programación no determinista y sincronización sin locks
- Lenguajes OpenMP, Cilk y CUDA (Thrust)

5. Evaluación del Rendimiento de Aplicaciones Paralelas

- Análisis de complejidad, paralelismo y localidad
- Medida del tiempo de ejecución, CPI, ancho de banda e intensidad aritmética
- Trabajo total y camino crítico de la ejecución paralela
- Speedup y eficiencia de la ejecución paralela
- Sobrecarga (overhead) de la sincronización y de las comunicaciones