

Objetivos / Competencias

La asignatura algoritmos paralelos inicia al alumno en el campo de la computación paralela proporcionándole una visión detallada de los diferentes paradigmas de procesamiento y su tratamiento tanto en arquitecturas multiprocesador como en arquitecturas multicomputador.

Esta asignatura se ve complementada en el plan de estudios con las asignaturas básicas que introduce al alumno en la programación, con la asignatura “Programación Concurrente” y con la asignatura “Programación Paralela y Distribuida”.

Los objetivos generales de la asignatura son:

- Proporcionar al alumno la capacidad de enfocar soluciones paralelas y distribuidas a los problemas
- Motivar la capacidad de exponer soluciones en público
- Motivar la capacidad de trabajo en grupo
- Motivar las dotes de dirección de equipos de trabajo en el ámbito

Los objetivos específicos de la asignatura podrían resumirse en los siguientes:

- Proporcionar al alumno una introducción básica a los diferentes modelos de computación en paralelo.
- Introducir al alumno en el área de la computación en paralelo, proporcionándole la metodología para tratar los principales paradigmas de programación en arquitecturas paralelas de memoria compartida y de memoria distribuida.
- Proporcionar al alumno las herramientas necesarias para el análisis del coste de procesamiento de un algoritmo paralelo sobre una arquitectura determinada y su comparación con el correspondiente procesamiento en secuencial.
- Motivar la capacidad del alumno para enfocar la visión en paralelo de la solución a aquellos problemas que, hasta el momento, siempre se atacaron desde el punto de vista de la computación secuencial.

Programa Teórico

Tema I: Introducción.

1. Definición y motivación de paralelismo
2. Introducción Histórica
3. Características de las arquitecturas paralelas
4. Análisis de Algoritmos Paralelos

Tema II: Paralelismo de Datos

1. Concepto de paralelismo de datos
2. Sentencia de creación de procesos FORALL
3. Ejemplo 1: Ranksort
4. Ejemplo 2: Suma y multiplicación de matrices
5. Operadores FORK y JOIN

Tema III: Comunicación entre procesos

1. Mecanismo de paso de mensajes
2. Pipeline
3. Ejemplo 1: Sistemas de ecuaciones
4. Solapamiento Comunicación/Computación
5. Arquitecturas embebidas
6. Pashed Array Communication
7. Ejemplo 2: Ordenación Bitónica
8. Agregación y Broadcasting

Tema IV: Paralelismo Síncrono

1. Concepto
2. Algoritmo de Relajación Jacobiana
3. Sincronización Local
4. Sincronización global:
 - a. Utilización de barreras lineales
 - b. Utilización de barreras logarítmicas
5. Algoritmos CAB

Tema V: Master-Worker

1. Concepto
2. El ejemplo de las n-reinas
3. Soluciones paralelas intuitivas
4. Implementación Distribuida

Contenido práctico

Realización de prácticas en ParallelJ sobre los diferentes problemas presentados en la teoría de la asignatura. ParallelJ es una extensión del lenguaje Java para la construcción de programas paralelos. Se utilizará una herramienta integrada en Eclipse que permite simular la ejecución de los programas sobre arquitecturas de memoria compartida y distribuida.

Metodologías / Actividades

La asignatura pone en práctica un esquema de aprendizaje basado en problemas. El temario de la asignatura se desarrollará empleando las tres horas semanales de clases prácticas que se tienen asignadas. Sin embargo, y dada la naturaleza práctica de la materia que se estudia, dichas clases teóricas serán también empleadas para la resolución de problemas, planteamiento de trabajos y resolución de dudas.

Semanalmente se propondrá un trabajo práctico a los alumnos. El alumno, de forma individual deberá realizar el trabajo y entregar el resultado dentro del plazo habilitado para ello. Quincenalmente se desarrollará una sesión de revisión con el profesor (durante las horas de tutoría) para analizar la progresión del alumno, problemas que encuentra, soluciones a tales problemas, etc.

Recomendaciones

Se recomienda a los alumnos que deseen cursar la asignatura optativa de 5º curso "Programación Distribuida" haber cursado esta previamente.

Criterios evaluación

evaluación continua a través de las prácticas y trabajos entregados semanalmente y la entrevista mantenida con el profesor al final del curso. Los alumnos que no superen la asignatura mediante este mecanismo tienen opción a realizar un examen consistente en una serie de problemas sobre el contenido analizado en la asignatura.

Bibliografía

- [Les93] Bruce P. Lester. "The Art of Parallel Programming".
Prentice Hall. 1993. Ba-2317
- [Akl92] Selim G. Akl. "Diseño y análisis de algoritmos paralelos".
Editorial Ra-ma. Serie paradigma. 1992. B-1275
- [Mor94] H. Stephen Morse. "Practical Parallel Computing".
AP Professional, 1994.