



MAESTRIA EN CIENCIAS DE LA COMPUTACION

Área: Sistemas Distribuidos

Programa de Asignatura: Programación Concurrente y Paralela

Código: MCOM 20700

Tipo: Obligatoria

Créditos: 9

Fecha: Noviembre 2012



1. DATOS GENERALES

Nombre del Programa Educativo:	Maestría en Ciencias de la Computación
Modalidad Académica:	Escolarizada
Nombre de la Asignatura:	Programación Concurrente y Paralela
Ubicación:	Primer semestre (Obligatoria)

2. REVISIONES Y ACTUALIZACIONES

Autores:	Dr. Luis Carlos Altamirano Robles
Fecha de diseño:	Noviembre 2012
Fecha de la última actualización:	Abril de 2019
Revisores:	Dr. Mario Rossainz López
Sinopsis de la revisión y/o actualización:	Actualización de todas las unidades del temario y actualización de referencias bibliográficas. Se actualizó la redacción del objetivo general y específicos.



3. OBJETIVOS

General:

El alumno aprenderá a desarrollar soluciones paralelas de problemas que se pueden paralelizar y aprenderá a mejorar el rendimiento o performance en sus aplicaciones.

4.- Específicos:

- El alumno aprenderá a resolver los problemas que surgen en la programación concurrente al comunicar y sincronizar procesos cuando éstos comparten recursos: Mutex, condiciones de sincronización del tipo productor-consumidor, interbloqueos o deadlocks, regiones críticas, semáforos y monitores.
- El alumno conseguirá ejecutar un programa secuencial en menos tiempo utilizando para ello más de un procesador y analizará el rendimiento de la aplicación paralela utilizando diversas métricas de rendimiento: speedup, Amdahl, CPIs, etc.
- El alumno entenderá la relación que existe entre la programación concurrente y paralela y las taxonomías de multiprocesadores de memoria compartida, memoria distribuida y memoria compartida distribuida.
- El alumno aprenderá a programar concurrentemente utilizando el esquema de programación de memoria compartida y paso de mensajes en diversos lenguajes de programación además de utilizar librerías diversas.
- El alumno aprenderá a analizar y diseñar algoritmos paralelos utilizando patrones de comunicación entre procesos como farms o granjas, trees o árboles, pipelines o cauces, cubos, hipercubos, mallas de procesos, etc., en la solución de un problema
- El alumno conocerá las tendencias actuales de la programación concurrente y paralela tales como Cómputo de Alto Rendimiento, Cómputo Paralelo Estructurado, entre otros.



4. CONTENIDO

Unidad	Contenido Temático/Actividades de aprendizaje
1.- Introducción a la programación concurrente y paralela	1.1. Conceptos Básicos 1.1.1. Concurrencia 1.1.2. Paralelismo 1.1.3. Sistemas Distribuidos 1.2. Taxonomía de Computadoras Paralelas 1.2.1. Flynn 1.2.2. Arquitecturas Paralelas actuales 1.3. Modelos de Programación Concurrente y Paralela 1.4. Taxonomía de Lenguajes Paralelos
2.- Programación Paralela con Memoria Compartida	2.1. Procesos e Hilos 2.2. Problemas Inherentes 2.2.1. Exclusión mutua 2.2.2. Algoritmos de exclusión mutua 2.2.3. Regiones críticas 2.3. Primitivas de sincronización 2.3.1. Candados 2.3.2. Semáforos 2.4. Primitivas de comunicación 2.4.1. Variables de condición 2.4.2. Espera y notificación 2.4.3. Monitores 2.5. Caso de estudio
3.- Programación Paralela con Memoria Distribuida (PMI)	3.1. Introducción 3.2. Elementos de comunicación 3.2.1. Procesos emisores (operación de envío) 3.2.2. Procesos receptores (operación de recepción) 3.2.3. Canales de comunicación (capacidad, flujo de datos, tipados, no-tipados) 3.2.4. Tipos de comunicación síncrona y asíncrona (directa-simétrica, directa-asimétrica, indirecta). 3.2.5. Mensajes (tipo, tamaño, paso por copia vs. paso por referencia) 3.3. Caso de estudio
4. Análisis de	4.1. Propiedades de corrección



Unidad	Contenido Temático/Actividades de aprendizaje
Corrección y Rendimiento	4.1.1. Seguridad 4.1.2. Vivacidad 4.1.3. No-Inanición 4.1.4. Equidad 4.2. Definiciones 4.2.1. Rendimiento 4.2.2. Escalabilidad 4.2.3. Granularidad 4.2.4. Aceleración 4.3. Métricas 4.3.1. Tiempo paralelo de ejecución 4.3.2. Speedup 4.3.3. Ley de Amdahl 4.3.4. CPI 4.3.5. Eficiencia
5.- Estrategias de Paralelización	5.1. Particionamiento 5.2. Divide y vencerás 5.3. Pipeline 5.4. Farms
6.- Algoritmos Paralelos	6.1. Algoritmos de ordenación 6.2. Algoritmos de búsqueda y optimización 6.3. Algoritmos numéricos
7. Tendencias de la Programación Paralela	7.1. Computación de altas prestaciones 7.2. Computación heterogénea 7.3. Programación paralela estructurada 7.4. Objetos paralelos



Bibliografía	
Básica	Complementaria
1. Raynal Michel. <i>“Concurrent Programming: Algorithms, Principles and Foundations”</i> . Springer Verlag. 2012.	1. Breitman, K. K., Horspool R. N. <i>“Patterns, Programming and Everything”</i> . Springer Verlag. 2012.
2. Sasikumar M., et al. <i>“Introduction to Parallel Processing”</i> . Second Edition. PHI Learning. USA 2014. ISBN: 8120350316	2. Capel, I. M., Rodríguez S. <i>“Sistemas Concurrentes y Distribuidos. Tomo II. Práctica”</i> . Universidad de Granada. España 2012.
3. Almeida F., Giménez D., Mantas J.M., Vidal A. <i>“Introducción a la Programación Paralela”</i> . Paraninfo CENGAGE Learning. España 2010.	3. Capel, I. M., Rodríguez S. <i>“Sistemas Concurrentes y Distribuidos. Tomo I. Teoría”</i> . Universidad de Granada. España 2012.
4. Bauer B. <i>“Practical Parallel Programming”</i> . Elsevier Science Publishing”. USA 2016. ISBN: 9780120828104	4. Faber R. <i>“Parallel Programming with OpenAcc”</i> . Morgan Kaufmann Pub. USA, 2016.
5. Pacheco P. <i>“An Introduction to Parallel Programming”</i> . Elsevier Inc. USA 2011. ISBN: 978-0-12-374260-5.	5. Blanquer I., Román J.E., Alonso J.M. <i>“Ejercicios de Programación Paralela con OpenMP y MPI”</i> . Universidad Politécnica de Valencia. España, 2018. ISBN: 9788490487150

5. CRITERIOS DE EVALUACIÓN

Criterios	Porcentaje
• Exámenes	25%
• Participación en clase	
• Tareas	10%
• Exposiciones	
• Simulaciones	
• Trabajo de investigación	20%
• Prácticas de laboratorio	20%
• Visitas guiadas	
• Reporte de actividades académicas y culturales	
• Mapas conceptuales	
• Portafolio	
• Proyecto final	25%
Total	100%