



BENEMÉRITA UNIVERSIDAD AUTÓNOMA DE PUEBLA

FACULTAD CIENCIAS DE LA COMPUTACION

PROGRAMA DE LA MATERIA CORRESPONDIENTE A LA LICENCIATURA EN CIENCIAS DE LA COMPUTACIÓN.

Coordinación: Área de Programación

NOMBRE DE LA MATERIA: Programación Concurrente y Paralela

Clave: LCC 318
Créditos: 10
Modalidad: Escolarizada

Nivel de Ubicación: Formativo
Tipo de Materia: Obligatoria

PRE-REQUISITOS: LCC 200 Algoritmos y Estructura de Datos

MATERIA CONSECUENTE: LCC 226 Sist. Operativos Central y Dist.

TIEMPO TOTAL ASIGNADO: 96 hrs.

PRIMAVERA – OTOÑO

HRS. TEÓRICAS/SEM: 4 **HRS. PRÁCTICAS/SEM:** 2

VERANO

HRS. TEÓRICAS/SEM: 8 **HRS. PRÁCTICAS/SEM:** 4

AUTOR(ES) DEL PROGRAMA:	
M.C. Beatriz Beltrán Martínez	M.C. Mariano Larios Gómez
M.C. Hilda Castillo Zacatelco	Dra. Darnes Vilariño Ayala
Dr. Mario Rossainz López	M.C. Rafael de la Rosa Flores
M.C. José Andrés Vázquez Flores	

REVISADO POR:	Área de Programación Coordinador: Beatriz Beltrán Martínez
APROBADO POR:	Academia
AUTORIZADO POR:	Docencia

FECHA DE ELABORACIÓN/REVISIÓN:	Septiembre 2006 / Agosto 2007
VIGENCIA:	A partir del período Primavera 2007

JUSTIFICACIÓN:

La programación concurrente originada en los años 60 en un inicio fue de importancia para el desarrollo de sistemas operativos, posteriormente con el diseño de máquinas multiprocesadoras ofreció no sólo un reto para los diseñadores de sistemas operativos, sino una oportunidad que los programadores podían aprovechar. El primer reto para dichos programadores fue el problema de la sección crítica y para mediados de los 70's se hizo evidente la necesidad de contar con métodos formales para controlar la complejidad de los programas concurrentes. En la actualidad la proliferación del procesamiento paralelo, del cómputo cliente servidor, la utilización de Internet y las estaciones y PC multiprocesos han generalizado el hardware concurrente y hecho más relevante que nunca la programación concurrente. Bajo este marco es vital que los estudiantes de Ciencias de la Computación conozcan y apliquen los modelos y principios de la comunicación síncrona y asíncrona.

OBJETIVO GENERAL DE LA MATERIA:

Que el estudiante conozca las diferentes técnicas de la programación concurrente y paralela, los lenguajes de programación concurrente más representativos y las técnicas de sincronización de procesos en un ambiente concurrente.

CONTRIBUCIÓN DE LA SIGNATURA AL PERFIL DE EGRESO:

En el perfil del egresado se plantea que éste tendrá una visión general de las ciencias de la computación y poseerá conocimientos sólidos para la construcción de soluciones basadas en sistemas concurrentes y paralelos.

CONTENIDO TEMÁTICO

MATERIA:

UNIDAD:1		TÍTULO: Introducción a la Programación Concurrente y Paralela				
<p>OBJETIVO ESPECÍFICO: Que el alumno conozca los conceptos básicos de la programación concurrente y paralela, así como su historia y la clasificación de los lenguajes concurrentes.</p> <p>Bibliografía[4,5,6,7]</p>						
CONTENIDO DE LA UNIDAD		Tiempo de impartición (hrs.)		Actividades de Aprendizaje	Técnicas	Recursos Necesarios
		HT	HP			
1.1	Historia de la programación concurrente	1/2		Explicar la evolución de la programación concurrente.	Explicación del profesor, lluvia de ideas, discusión grupal.	Cañón, computadora, pizarrón, plumones.
1.2	Lenguajes de Programación Concurrente	1/2		Explicar las características de los diferentes lenguajes de programación concurrente.	Explicación del profesor, lluvia de ideas, discusión grupal.	
1.3	Conceptos básicos.	1		Explicar y analizar los conceptos básicos de la programación concurrente y paralela.	Explicación del profesor, lluvia de ideas, discusión grupal.	
HORAS TOTALES:		2				

UNIDAD: 2		TÍTULO: Arquitecturas paralelas y distribuidas				
OBJETIVO ESPECÍFICO:						
Que el alumno conozca las características principales de las computadoras paralelas y distribuidas, así como el uso de la memoria compartida en el nivel de aplicación y el uso de arquitecturas para hacer computo paralelo y distribuido						
Bibliografía [1]						
CONTENIDO DE LA UNIDAD		Tiempo de impartición (hrs.)		Actividades de Aprendizaje	Técnicas	Recursos Necesarios
		HT	HP			
2.1	Arquitecturas simples 2.1.1 SISD, SIMD, MISD y MIMD 2.1.1.1 Procesador de vectores 2.1.1.2 Procesador de Arreglos 2.1.2 UMA Y NUMA 2.1.2.1 Bus 2.1.2.2 Conmutador 2.1.2.3 CC-NUMA 2.1.2.4 CN-NUMA 2.1.3 COMA	2		Definir las arquitecturas simples de computadoras y sus posibles usos en una LAN	Exposición del profesor y lluvia de ideas	Cañón, computadora, pizarrón, plumones.
2.2	Arquitecturas paralelas 2.2.1 SMP 2.21 COW 2.22 MPP 2.22.1 Cuadrícula 2.22.2 Hypercubo	2		Aplicación computo paralelo en una arquitectura simple o paralelas	Exposición del profesor y lluvia de ideas	Cañón, computadora, pizarrón, plumones.
2.3	Arquitecturas distribuidas 2.3.1 GRID 2.3.2 Coordinación de actividades	2		Discutir las diferentes arquitecturas. Introducción a las aplicaciones distribuidas en	Exposición del profesor y lluvia de ideas	Cañón, computadora, pizarrón, plumones.

Middleware en GRID			arquitecturas como GRID en
2.3.3 GLOBUS			WAN
2.3.4 CROSS-BROKER			
HORAS TOTALES:	6		

UNIDAD: 3

TÍTULO: Diseño de Algoritmos Paralelos y Distribuidos

OBJETIVO ESPECÍFICO:

El alumno identificará y aplicará técnicas de diseño tanto de aplicaciones paralelas como distribuidas.

Bibliografía: [2]

CONTENIDO DE LA UNIDAD	Tiempo de impartición (hrs.).		Actividades de Aprendizaje	Técnicas	Recursos Necesarios
	HT	HP			
3.1 Algoritmos paralelos elementales	7	3	Explicar las diferentes técnicas para el diseño de algoritmos paralelos y experimentar con cada una de ellas para resolver algún problema.	Exposición por parte del profesor, elaboración por parte de los estudiantes de algoritmos utilizando tales técnicas.	Pizarrón, borrador, proyector de acetatos, cañón y equipo de computo
3.1.1 Clasificación de algoritmos MIMD					
3.1.2 Algoritmos de Reducción					
3.1.2.1 Modelo de Hipercubos					
3.1.2.2 Modelo SIMD de intercambio					
3.1.2.3 Modelo SIMD en mallas 2-D					
3.1.2.4 Modelo de multiprocesadores UMA					
3.1.3 Algoritmos de emisión (Broadcast)					
3.1.4 Sumas de prefijos					
3.2 Paradigmas algorítmicos	7	3	Explicar los diversos paradigmas que influyen directamente en el desarrollo de los algoritmos.	Exposición por parte del profesor, elaboración por parte de los estudiantes de algoritmos utilizando tales paradigmas	Pizarrón, borrador, proyector de acetatos, cañón y laboratorio de computo
3.2.1 Paralelismo de Datos					
3.2.1.1 Granjas de Procesos					
3.2.2 Pipeline o Encauzamiento					
3.2.3 Maestro-Esclavo					
3.2.3.1 Divide y Vencerás					
3.2.3.2 Ramificación y Poda					

3.2.4 Autómatas Celulares				
3.2.5 Pares totales				
HORAS TOTALES:		14	6	

UNIDAD: 4				TÍTULO: Introducción a la programación paralela y distribuida		
<p>OBJETIVO ESPECÍFICO: Que el alumno identifique los problemas que existen en la comunicación entre procesos y la forma de resolverlos. Además que aplique algunas técnicas de sincronización de procesos en un ambiente paralelo y en un ambiente distribuido.</p>						
CONTENIDO DE LA UNIDAD		Tiempo de impartición (hrs.)		Actividades de Aprendizaje	Técnicas	Recursos Necesarios
		HT	HP			
4.1	Exclusión mutua, región crítica y sincronización con hilos 4.1.1 Alternancia 4.1.2 Falta de exclusión 4.1.3 Interbloqueo 4.1.4 Espera infinita 4.1.5 Algoritmo de Dijkstra 4.1.6 Algoritmo de Dekker 4.1.7 Algoritmo de Peterson	8	2	Identificar el concepto de condición de competencia, exclusión mutua, región crítica, analizando los diferentes algoritmos para garantizar la exclusión mutua, espera ocupada, comunicación y sincronización de hilos.	Explicación del profesor, lluvia de ideas, discusión grupal. Práctica de laboratorio	Cañón, computadora, pizarrón, plumones
4.2	Monitores, Paso de mensajes	6	4	Analizar el concepto de monitor como un medio de obtener exclusión mutua, y la necesidad de utilizar paso de mensajes	Explicación del profesor, lluvia de ideas, discusión grupal. Prácticas de laboratorio.	Canon, computadora, pizarrón, plumones

4.3	Procesamiento paralelo 4.3.1 Rendimiento o Speed-Up 4.3.2 Ley de Amdahl 4.3.3 Los ciclos por instrucción	4	2	entre procesos en la misma computadora y en procesos en computadoras distintas. Implementar programas utilizando estos conceptos en algún lenguaje de programación, por ejemplo JAVA. Identificar los conceptos básicos del procesamiento paralelo, analizando ventajas y desventajas, en cuanto a costos, rendimiento y complejidad. Discutir y analizar el rendimiento proporcionado por un programa en paralelo contra uno secuencial, así como la ley de Amdahl y el método de ciclos por instrucción.	Explicación del profesor, lluvia de ideas, discusión grupal. Prácticas de laboratorio.	Canon, computadora, pizarrón, plumones
HORAS TOTALES:		18	8			

UNIDAD: 5		TÍTULO: Aplicaciones				
OBJETIVO ESPECÍFICO: El estudiante aplicara los deferentes tipos de algoritmos utilizados, considerando diferentes esquemas de comunicación en dependencia del problema a resolver						
Bibliografía: [3]						
	CONTENIDO DE LA UNIDAD	Tiempo de impartición (hrs.).		Actividades de Aprendizaje	Técnicas	Recursos Necesarios
		HT	HP			
5.1	Algoritmos de Ordenación	4	2	Desarrollar paralelamente diversos algoritmos de ordenación	Exposición por parte del profesor de los algoritmos a paralelizar. Diseñar	Salón, pizarrón, plumones, proyector de acetatos o cañón y

5.2	Algoritmos Numéricos	12	6	Desarrollar paralelamente estas técnicas	conjuntamente con el estudiante el algoritmo paralelo y elaboración de programas paralelos por parte de los estudiantes	laptop
	5.2.1 Multiplicación de matrices 5.2.2 Solución de Sistemas de Ecuaciones Lineales 5.2.3 Iterativos 5.2.4 Búsqueda y Optimización 5.2.4.1 Algoritmos Genéticos 5.2.4.2 Refinamiento Sucesivo 5.2.4.2 Algoritmos Hill Climbing				Exposición por parte del profesor de los algoritmos a paralelizar. Diseñar conjuntamente con el estudiante el algoritmo paralelo y elaboración de programas paralelos por parte de los estudiantes	Salón, pizarrón, plumones, proyector de acetatos o cañón y laptop
	HORAS TOTALES:	16	8			

PRÁCTICAS			
UNIDAD	NOMBRE DE LA PRÁCTICA	OBJETIVO	HORAS
3.1	Diseño de algoritmos con diferentes arquitecturas	Que el estudiante aprenda a desarrollar diferentes esquemas algorítmicos, considerando la arquitectura	3
3.2	Diseño de algoritmos utilizando diferentes esquemas de paralelización.	Que el desarrolle algoritmos siguiendo diferentes esquemas de comunicación, lo que le permitirá, darse cuenta la importancia de los mismos en el diseño de algoritmos paralelos	3
4.1	Algoritmos para lograr la exclusión mutua	Que el alumno identifique los principales problemas que surgen al tratar de implementar la exclusión mutua.	2
4.2	Implementación de un problema típico de sincronización de procesos utilizando monitores	Que el alumno comprenda como utilizar monitores en la sincronización de problemas	2

4.2	Implementación de un problema típico de sincronización de procesos utilizando paso de mensajes	Que el alumno comprenda como utilizar paso de mensajes en la sincronización de problemas	2
4.3	Implementación de un programa paralelo	Que el alumno utilice las técnicas de paralelismo para escribir un programa, además el alumno deberá utilizar alguna técnica para medir el rendimiento del algoritmo correspondiente.	2
5.1	Paralelización de diferentes algoritmos de ordenación	Realizara la paralelización de diferentes algoritmos de ordenación. Discusión de la eficiencia en los algoritmos diseñados	2
5.2	Desarrollo de algoritmos paralelos para multiplicar matrices	Desarrollo de un algoritmo paralelo eficiente que permita multiplicar matrices.	1
5.2	Paralelizacion de diferentes algoritmos exactos para resolver sistemas de ecuaciones lineales: Gauss con selección parcial del pivote, Cholevsky, descomposición LU, descomposición QR	Que el estudiante aprenda a desarrollar algoritmos paralelos que permitan resolver de manera eficiente sistemas de ecuaciones lineales	3
5.2	Paralelizacion de diferentes tipos de algoritmos de búsqueda: Algoritmos genéticos , refinamiento sucesivo entre otros	Diseñar algoritmos paralelos para resolver diversos tipos de problemas de optimización	3

CRITERIOS DE EVALUACIÓN

EXÁMENES PARCIALES

Parcial	Contenido a evaluar	Periodos
I	1 y 2	6ª semana
II	3 y 4	10ª semana
III	5	15ª semana

	%
Asistencias y participación:	A criterio del Profesor
Exámenes parciales:	30%
Tareas:	10%
Trabajos de Investigación:	10%
Prácticas de Laboratorio:	30%
Proyecto Final	20%
TOTAL:	100%

REQUISITOS DE ACREDITACIÓN:

Obtener una calificación mayor o igual a seis en el promedio de exámenes y en el proyecto final.

FOMENTO DE VALORES:

El estudiante será capaz de trabajar en equipo, de analizar problemas y de colaborar con sus demás compañeros.

BIBLIOGRAFÍA:

- [1] Andrew S. Tanenbaum. Organización de computadoras. Pearson educación
- [2] Michael J. Quinn : Parallel Computing. Theory and Parctice. McGRAW- Hill, INC. 1994
- [3] Parallel Programming. Techique and Applicatrions using Networked Workstation san Parallel Computers. Barry Wilkinson, Michael Allen. Prentice Hall, New Jersey. 1999
- [4] Doug Lea, "Programación Concurrente en Java Principios y patrones de diseño", Addison Wesley segunda edición 2001.
- [5] Burns, A.; Davis, G., "Concurrente Programming", Addison Wesley 1994.
- [6] Ben-Ari M., "Principles of Concurrent and Distributed Programming", Prentice Hall, 1990.
- [7] HARTLEY, S. J. "Concurrent Programming. The Java Programming Language", Oxford University Press, 1998.