



CONCURRENCIA Y PARALELISMO

Carrera: *Ingeniería en Computación*

Año: 4°

Duración: *Semestral*

Correlativas:

Taller de Lenguajes II

Redes de Datos 1

Profesores: Dr. Marcelo Naiouf
Dra. Laura De Giusti
Lic. Franco Chichizola

Hs. semanales: 6 hs.

FUNDAMENTACIÓN:

La temática de la Concurrencia es central en el desarrollo actual de la Ciencia Informática, en particular por el creciente desarrollo de arquitecturas multiprocesador que permiten implementar físicamente los conceptos teóricos de concurrencia “real”. El impacto de la concurrencia se refleja en diferentes ámbitos de la disciplina tales como las arquitecturas, los sistemas operativos, los lenguajes y el diseño y desarrollo de aplicaciones. En este sentido, se impone que los futuros profesionales sean capaces de desarrollar soluciones que utilicen adecuadamente la tecnología disponible con fundamentos teóricos firmes.

OBJETIVOS GENERALES:

Dar los conceptos fundamentales de Concurrencia en software. Analizar la semántica y sintaxis para especificar concurrencia. Estudiar la sincronización de procesos concurrentes por memoria compartida y mensajes. Desarrollar estudios de casos con diferentes lenguajes/ herramientas para concurrencia.

CONTENIDOS MINIMOS:

- Especificación de la ejecución concurrente.
- Comunicación y sincronización.
- Concurrencia con variables compartidas.
- Concurrencia con pasajes de mensajes.
- Lenguajes de programación concurrente.
- Introducción a los conceptos de procesamiento paralelo.



PROGRAMA ANALÍTICO:

1. *Conceptos básicos*

Objetivos de los sistemas concurrentes.

Procesamiento secuencial, concurrente y paralelo. Características.

Concurrencia y paralelismo. Relación con la arquitectura.

Sincronización (por exclusión mutua y por condición) y comunicación (por memoria compartida y por mensajes).

2. *Concurrencia y sincronización*

Especificación y semántica de la ejecución concurrente. La sentencia *co* y *process*

Acciones atómicas y sincronización.

El problema de interferencia. Historias válidas e inválidas.

Atomicidad de grano fino y de grano grueso.

La propiedad de “A lo sumo una vez”.

La sentencia *Await*. Semántica. Especificación de la sincronización.

Técnicas para evitar interferencia.

Propiedades de seguridad y vida.

Políticas de scheduling y Fairness.

3. *Concurrencia con variables compartidas*

Sincronización por variables compartidas

Sincronización de grano fino.

Secciones críticas (SC). Definición del problema. Propiedades necesarias de las soluciones. Planteo de soluciones clásicas.

Sincronización *Barrier*. Definición del problema. Planteo de soluciones.

Sincronización por semáforos

Defectos de la sincronización por variables compartidas.

Semáforos. Sintaxis y semántica.

Usos básicos y técnicas de programación con semáforos.

Sincronización por monitores

Evolución histórica a partir de semáforos.

Monitores. Sintaxis y semántica.

Sincronización en monitores. Disciplinas de señalización: “*Signal and wait*” y “*Signal and continue*”.

Usos básicos y técnicas de programación con monitores.

Lenguajes para programación con variables compartidas

Librería para manejo de threads – *Pthreads*.



4. Programación distribuida. Concurrencia con pasaje de mensajes

Programas distribuidos. Relación entre mecanismos de comunicación.

Pasaje de mensajes asincrónicos (PMA)

Sintaxis y semántica. Canales. Operaciones.

Usos básicos y técnicas de programación con Pasaje de Mensajes Asincrónicos.

Pasaje de mensajes sincrónicos (PMS)

Sintaxis y semántica.

Conceptos de CSP. Comunicación guardada. Sintaxis y semántica.

Usos básicos y técnicas de programación con Pasaje de Mensajes Sincrónicos.

Remote Procedure Calls (RPC) y Rendezvous.

Sintaxis y semántica.

Similitudes y diferencias.

Usos básicos y técnicas de programación con RPC y Rendezvous.

Lenguaje para Programación distribuida

Librería para manejo de mensajes – MPI.

Lenguajes para Rendezvous - Ada.

5. Introducción a la Programación Paralela

Objetivos del procesamiento paralelo.

Necesidad del paralelismo.

Concepto de Sistema Paralelo.

Diseño de algoritmos paralelos.

Métricas de rendimiento de los sistemas paralelos (speedup y eficiencia).

Concepto de asignación de tareas y balance de carga.

Balance de carga estático y dinámico.

Arquitecturas orientadas a Procesamiento Paralelo.

6. Paradigmas de cómputo paralelo

Objetivo de los paradigmas de resolución de programas paralelos.

Paradigma Master/Slave.

Paradigma Divide/Conquer.

Paradigma Pipelining.

Otros paradigmas.

En la práctica se realiza trabajo experimental sobre arquitecturas multiprocesador distribuidas (clusters), multiprocesadores con memoria compartida e híbridos.



BIBLIOGRAFÍA OBLIGATORIA:

- Andrews G.. “Foundations of Multithreaded, Parallel and Distributed Programming”, Addison Wesley, 2000
- Ben-Ari, M. “Principles of Concurrent and Distributed Programming, 2/E”. Addison-Wesley. 2006. ISBN 0-321-31283-X
- Grama A., Gupta A., Karypis G., Kumar V., "An Introduction to Parallel Computing. Design and Analysis of Algorithms", Pearson Addison Wesley, 2nd Edition, 2003
- Ghosh, Sukumar. “Distributed Systems: An Algorithmic Approach”. Chapman & Hall/CRC, 2007. ISBN1584885645, 9781584885641

BIBLIOGRAFÍA COMPLEMENTARIA:

- Akl S., “Parallel Computation. Models and Methods”, Prentice-Hall, Inc., 1997.
- Barnes J., “Programming in Ada 2005 with CD”, Addison Wesley, 2006.
- Brinch Hansen, P., “Studies in Computational Science. Parallel Programming Paradigms”, Prentice Hall, 1995.
- Chandy, Misra, “Parallel Program Design. A Foundation”, Addison Wesley, 1988.
- Chiola G., G. Ciaccio, “Lightweighth Messaging Systems”, in R. Buyya Ed., High Performance Cluster Computing: Architectures and Systems, Vol. 1, Prentice-Hall, Upper Saddle River, NJ, USA, pp. 246-269, 1999.
- Downey, Allen. “The Little Book of Semaphores, Second Edition”. Free book disponible en <http://www.freetechbooks.com/the-little-book-of-semaphores-second-edition-t519.html>, 2007.
- Goetz B., Peierls T., Bloch J., Bowbeer J., “Java Concurrency in Practice”, Addison Wesley, 2006.
- Habermann A., Perry D., “Ada for Experienced Programmers”, Addison-Wesley Publishing Company, Inc. 1993.
- Herlihy M., Shavit N., “The Art of Multiprocessor Programming”. Morgan Kaufmann, 2008.
- Hoare C., “Communicating Sequential Processes”, Englewood Cliffs, Prentice Hall, 1985
- Jordan H.F., Alaghband G., Jordan H.E., "Fundamentals of Parallel Computing", Prentice Hall, 2002.
- Leopold C., "Parallel and Distributed Computing. A survey of Models, Paradigms, and Approaches", Wiley Series on Parallel and Distributed Computing. Albert Zomaya Series Editor, 2001.
- Snir, M., Otto, S., Huss-Lederman, S., Walker, D., Dongarra, J. *MPI: The Complete Reference*. Cambridge, MA: MIT Press, 1996. Available in web site: <http://www.netlib.org/utk/papers/mpi-book/mpi-book.html>.
- Taubenfeld, Gadi. “Synchronization Algorithms and Concurrent Programming”. Prentice Hall. 2006.