



Universidad Tecnológica de la Mixteca

Clave DGP: 200089

Maestría en Sistemas Distribuidos PROGRAMA DE ESTUDIOS

NOMBRE DE LA ASIGNATURA

Sistemas de Cómputo Paralelos y Distribuidos

SEMESTRE	CLAVE DE LA ASIGNATURA	TOTAL DE HORAS
Primer Semestre	100104V	80

OBJETIVO(S) GENERAL(ES) DE LA ASIGNATURA

Que el alumno conozca los conceptos de diseño de los sistemas de cómputo paralelos y distribuidos en la base de los enfoques modernos.

TEMAS Y SUBTEMAS

1. Introducción en sistemas paralelos y distribuidos
 - 1.1. Tendencias modernas de desarrollo de sistemas de cómputo.
 - 1.2. Organizaciones de sistemas de cómputo (sistema de clasificación de Flynn).
 - 1.3. Modelos de la memoria (sistemas con la memoria compartida y distribuida).
 - 1.4. Arquitecturas paralelas alternativas (dataflow architecture, systolic architecture).
 - 1.5. Nuevas tendencias de tecnología.
 - 1.6. Generaciones de sistemas de cómputo paralelos y distribuidos.
 - 1.7. Tareas.

2. Las direcciones principales en el diseño de procesadores
 - 2.1. Los procesadores RISC y CISC.
 - 2.2. Segmentación de path de datos (una maquina RISC – MIPS64).
 - 2.3. Riesgos de segmentación.
 - 2.4. Extensión de la segmentación de MIPS para manipular operaciones multiciclo.
 - 2.5. Planificación dinámica de la segmentación.
 - 2.6. Predicción de saltos.
 - 2.7. Ejecución especulativa.
 - 2.8. Emisión múltiple.
 - 2.9. Procesadores VLIW.
 - 2.10. Tareas.

3. Paralelismo como una base de aumento de rendimiento de computadoras
 - 3.1. Paralelismo a nivel de instrucción.
 - 3.2. Dependencias.
 - 3.3. Paralelismo a nivel de programa.
 - 3.4. Planificación estática de segmentación.
 - 3.5. Desenrollamiento de los ciclos.
 - 3.6. Paralelismo a nivel de tarea.
 - 3.7. Tareas.

4. Programas paralelos
 - 4.1. Casos de aplicaciones paralelas.
 - 4.2. Proceso de paralelización.
 - 4.2.1. Descomposición (de computación en las tareas).
 - 4.2.2. Asignación.

- 4.2.3. Orquestación.
- 4.2.4. Mapeo.
- 4.3. Paralelización de un programa de ejemplo.
- 4.4. Tarea.

- 5. Rendimiento de sistemas de cómputo paralelos
 - 5.1. Rendimiento de algoritmos paralelos.
 - 5.2. Acceso a los datos y comunicaciones en sistemas de multimediamoria.
 - 5.3. Jerarquía extendida de la memoria.
 - 5.4. Explotación de localidad temporal y espacial.
 - 5.5. Tareas.

- 6. Modelos de arquitecturas de memoria de sistemas de cómputo paralelos y distribuidos
 - 6.1. Modelos de arquitecturas de la memoria de acceso uniforme (UMA).
 - 6.2. Modelos de arquitecturas de la memoria de acceso no uniforme (NUMA).
 - 6.3. Modelos de arquitecturas con la memoria distribuida.
 - 6.4. La coherencia de la caché en los sistemas paralelos.
 - 6.5. Soluciones de problemas de la coherencia de caché con software.
 - 6.6. Soluciones de problemas de la coherencia de caché con hardware.
 - 6.7. Protocolos de vigilancia (snoop protocols).
 - 6.8. Protocolos en la base de referencia.
 - 6.9. Tareas.

- 7. Topologías de sistemas de cómputo
 - 7.1. Funciones de enrutamiento de datos.
 - 7.2. Topologías estáticas.
 - 7.3. Topologías dinámicas.
 - 7.4. Diseño de conmutadores.
 - 7.5. Red de CRAY T3D.
 - 7.6. Redes IBM SP-1, SP-2.
 - 7.7. Interfaz coherente escalable.
 - 7.8. Interconexión SGI Origin.
 - 7.9. Tareas.

- 8. Sistemas de cómputo SIMD
 - 8.1. Procesamiento vectorial.
 - 8.2. Colocación de los vectores de datos en la memoria.
 - 8.3. Procesador vectorial.
 - 8.4. Sistemas de cómputo con estructura sistólica.
 - 8.5. Procesadores VLIW.
 - 8.6. Tareas.

- 9. Sistemas de cómputo MIMD
 - 9.1. Multiprocesadores escalables.
 - 9.2. Escalabilidad.
 - 9.2.1. Integración de nivel de chip.
 - 9.2.2. Integración de nivel de tarjeta.
 - 9.2.3. Integración de nivel de sistema.
 - 9.3. Modelos de programación.
 - 9.4. Procesamiento de mensajes.
 - 9.5. Espacio compartido de direcciones físicas.
 - 9.6. Caso de estudio: Cray T3D.
 - 9.7. Clústeres y redes de estaciones de trabajo.
 - 9.8. Tareas.

- 10. Redes de interconexión
 - 10.1. Definiciones básicas de redes de interconexión.
 - 10.2. Una red simple.
 - 10.3. Medios de las redes de interconexión.
 - 10.4. Conexión entre más de dos computadoras.

10.5. Tareas.

11. Modelos de Sistemas distribuidos

- 11.1. Características de los sistemas de cómputo distribuido y de cómputo paralelo.
- 11.2. Arquitectura de los sistemas distribuidos.
- 11.3. Ejemplos documentados de sistemas distribuidos.
 - 11.3.1. SETI@home.
 - 11.3.2. Akamai.
 - 11.3.3. BOINC.
 - 11.3.4. Proyectos P2P.
- 11.4. Problemas y desafíos en el diseño de sistemas distribuidos.
- 11.5. Tareas.

12. Sincronización en los sistemas distribuidos

- 12.1. Importancia de la sincronización en los sistemas distribuidos.
- 12.2. Relojes, eventos y estados.
- 12.3. Sincronización de relojes físicos.
- 12.4. Tiempo lógico y relojes lógicos.
- 12.5. Estados globales.
- 12.6. Algoritmos de sincronización.
- 12.7. Tareas.

13. Consistencia y replicación

- 13.1. Particionamiento.
- 13.2. Replicación de datos.
- 13.3. Recuperación de fallas.
- 13.4. Tareas.

14. Recuperación y tolerancia a fallo

- 14.1. Clases de fallas.
- 14.2. Detección de fallas.
- 14.3. Recuperación de transacciones.
- 14.4. Recuperación de datos.
- 14.5. Enmascarado jerárquico y de grupos.
- 14.6. Tareas.

15. Balanceo de la carga

- 15.1. Clasificación del balanceo de carga.
- 15.2. Modelos de asignación estática de carga.
- 15.3. Modelos de planificación dinámica de carga.
- 15.4. Casos de estudio.
- 15.5. Tareas.

16. Casos de estudio

- 16.1. Google.
- 16.2. Grids.
- 16.3. Sistemas de archivos.
- 16.4. Sistemas P2P.
- 16.5. Bases de datos.
- 16.6. Tareas.

ACTIVIDADES DE APRENDIZAJE

Lecturas semanales de las actividades indicadas por el profesor. Lecturas de artículos y documentos especializados de los temas vistos. Implementación de los modelos arquitectónicos más importantes planteados en el plan. Búsqueda y documentación por parte de los alumnos de los casos de estudio más importantes indicados por el profesor. Implementación de los algoritmos distribuidos más importantes para retroalimentar los conocimientos adquiridos durante las lecturas semanales.

CRITERIOS Y PROCEDIMIENTOS DE EVALUACIÓN Y ACREDITACIÓN

Tareas semanales, exámenes parciales y/o proyectos y examen final. Esto tendrá una equivalencia del 100 % en la calificación final del semestre.

BIBLIOGRAFÍA (TIPO, TÍTULO, AUTOR, EDITORIAL Y AÑO)**Básica:**

1. Parallel Computer Architecture: A Hardware/Software Approach. David Culler, J.P. Singh, Anoop Gupta. Morgan Kaufmann; 1 edition. 1998. ISBN-10: 1558603433. ISBN-13: 978-1558603431.
2. Distributed Systems: Concepts and Design. George Coulouris, Jean Dollimore. Addison-Wesley; 5 edition. 2011. ISBN-10: 0132143011. ISBN-13: 978-0132143011.
3. Distributed Systems: An Algorithmic Approach. Sukumar Ghosh. Chapman and Hall/CRC; 2 edition. 2014. ISBN-10: 1466552972. ISBN-13: 978-1466552975.
4. Computer Organization and Design: The Hardware/Software Interface. David A. Patterson, John L. Hennessy. Morgan Kaufmann; 5 edition. 2013. ISBN-10: 0124077269. ISBN-13: 978-0124077263.

Consulta:

1. Distributed Computing: Principles, Algorithms, and Systems. Ajay D. Kshemkalyani, Mukesh Singhal. Cambridge University Press; Reissue edition. 2011. ISBN-10: 0521189845. ISBN-13: 978-0521189842.
2. Distributed Systems: Principles and Paradigms. Andrew S. Tanenbaum, Maarten Van Steen. Prentice Hall; 2 edition. 2006. ISBN-10: 0132392275. ISBN-13: 978-0132392273.
3. From P2P and Grids to Services on the Web: Evolving Distributed Communities (Computer Communications and Networks). Ian J. Taylor, Andrew Harrison. Springer; 2 edition. 2008. ISBN-10: 1848001223. ISBN-13: 978-1848001220.
4. Computer Arithmetic: Algorithms and Hardware Designs. Behrooz Parhami. Oxford University Press; 2 edition. 2009. ISBN-10: 0195328485. ISBN-13: 978-0195328486.

PERFIL PROFESIONAL DEL DOCENTE

Maestro en Ciencias o Doctor en Electrónica, Computación o afín, es necesario conocimientos en electrónica digital, compiladores y sistemas distribuidos. Con experiencia en esta modalidad educativa.