Ministerio de Cultura y Educación **(Programa del año 2006)**
Universidad Nacional de San Luis **(Programa en trámite de aprobación)**
Facultad de Ciencias Físico-Matemáticas y Naturales **(Presentado el 29/11/2006 10:44:12)**
Departamento: Informatica
Area: Area II: Sistemas de Computacion

I - Oferta Académica

Materia	Carrera	Plan	Año	Período
ARQUITECTURA DEL PROCESADOR II	LIC. CS. COMP.	006/05	3	2c

II - Equipo Docente

Docente	Función	Cargo	Dedicación
AGUIRRE, GUILLERMO CARLOS	Prof. Responsable	P.ADJ EXC	40 Hs
ARROYUELO, MONICA DEL VALLE	Responsable de Práctico	A.1RA SEM	20 Hs
PERNA, JUAN IGNACIO	Responsable de Práctico	JTP EXC	40 Hs
ARROYUELO BILLIARDI, JORGE A.	Auxiliar de Práctico	A.1RA EXC	40 Hs

III - Características del Curso

Credito Horario Semanal				
Teórico/Práctico	**Teóricas**	**Prácticas de Aula**	**Práct. de lab/ camp/ Resid/ PIP, etc.**	**Total**
Hs	2 Hs	4 Hs	Hs	6 Hs

Tipificación	Periodo
C - Teoria con prácticas de aula	2 Cuatrimestre

Duración			
Desde	**Hasta**	**Cantidad de Semanas**	**Cantidad de Horas**
07/08/2006	10/11/2006	14	84

IV - Fundamentación

La creciente demenda de mayor poder computacional de los procesadores ha dado origen a arquitecturas complejas con unidades funcionales especializadas, memorias de alta velocidad, multiprocesadores, división de tareas en etapas concurrentes, ejecución de intrucciones a medida que están disponibles sus operandos, etc. Todas estas particularidades hacen necesaria la aplicación de técnicas y estrategias más complejas que la involucradas en los procesadores simples. Las arquitecturas avanzadas están concebidas para procesar problemas específicos que deben ser identificados y resueltos mediante prácticas de programación especiales.

V - Objetivos

Al finalizar el curso los alumnos deberan poder:

*Identificar los principales parámetros que miden el desempeño de los procesadores.

*Reconocer las similitudes y diferencias entre los distintos esquemas de memorias cache.

*Desarrollar el análisis cuantitativo del desempeño de procesadores
provistos con memoria cache.

*Conocer las características de los proccesadores pipeline y las

particularidades consideraras al medir el desempeño de los mismos.

*Comprender el funcionamiento general de los procesadores de arreglo y
las particularidades de los problemas que se adecuan a estas arquitecturas.

*Dominar los detalles involucrados en el funcionamiento de los procesadores pipeline con etapas multiciclo y los estrategias empleadas para despachar instrucciones fuera de orden.

*Describir el modo en que se realizan las computaciones en los
sistemas dataflow.

*Percibir la complejidad requerida para computar soluciones en sistemas de multiprocesadores y las redes de interconexion necesarias en estos sistemas.

*Determinar como se clasifican los diferentes procesadores según el nivel de paralelismo que presentan.

VI - Contenidos

Unidad 1: Performance en procesadores RISC.

Procesadores RISC. Arquitectura del set de instrucciones RISC. Frecuencia y ciclo de reloj. Unidades pequeñas de tiempo. Ecuación de tiempo de CPU. Ciclos por instrucción. Relación entre el ciclo y el tiempo de CPU. Desempeño y tiempo de ejecución. Arquitectura multiciclo. Señales de control y camino de datos. Set de instrucciones y desempeño. Aceleración (speed-up). Ley de Amdahl.

Unidad 2: Pipelining.

¿Qué es pipelining?. Comportamiento básico del pipeline. Performance ideal. Problemas en los pipe: los hazards. Distintos tipos de hazards: estructurales, de datos y de control. Dependencia de datos. Hazards de datos: RAW, WAW, WAR. Los atascos (stalls). Adelantamiento. Interlock. Impacto de los hazards en el análisis de la performance. Penalidades por branch. Técnicas de predicción de branch. Salto demorado. La implementación de pipeline en MIPS de 5 y 8 etapas.

Unidad 3: Paralelismo a nivel de instrucción y Pipelines avanzados.

Ejecución multi-ciclo. Finalización fuera de orden. Latencia e intervalo de iniciación. Superando dependencias de datos con scheduling dinámico de instrucciones. Scoreboarding. Algoritmo de Tomasulo, aplicación en una unidad de punto flotante. Procesadores vector. Instrucciones vector. Procesamiento escalar vs. procesamiento vectorial.Particulridades de los procesadores vector.

Unidad 4: Memorias.

Jerarquía de memorias. Memorias cache. Hit y Miss. Consideraciones generales del empleo de cache. Organizaciones de cache: directas, asociativas y conjunto asociativo. Performance de memorias cache. Tipos de cache:write-back y write-through. Memoria virtual y cache. Comportamiento de la jerarquía de
memorias. Penalidades por miss. Desempeño considerando el usa de jerarquía de memoria. Ejemplo Alpha AXP 21264.

Unidad 5: Multiprocesadores y redes de interconexión.

Nivel de paralelismo en los programas. Clasificación de Flynn. Arquitecturas de memoria centralizada (UMA) y distibuida (NUMA). Redes de interconexión: consideraciones
generales. Interconexiones dinámicas. Esquemas de única etapa y de múltiples etapas. Sistemas de buses. Redes crossbar. Redes Omega. Redes bloqueantes y no bloqueantes. Cache en sistemas de
multiprocesadores. Coherencia de cache. Protocolos snoopy y directorio.

VII - Plan de Trabajos Prácticos

Prácticos Nro. 1 y 2: Programación assembly y Medidas de Performance.
Programación en MIPS. Uso de la Ley de Amdhal para el cálculo de la perfomance ganada por una mejora (Speedup). Uso de la ecuación de la CPU. Camino de datos

Prácticos Nro. 3 : Pipelines y riesgos
Implementación del pipelining del MIPS. Impacto de problemas en el pipe: hazard de estructurales y hazard de datos. Los hazard de control: técnicas utilizadas para las instrucciones de salto. Planificación estática de instrucciones para reducir los hazards.

Prácticos Nro. 4 : Planificación Dinámica
Algoritmo de scoreboarding. Manejo de los distintos tipos de hazards. Algoritmo de Tomasulo. Scoreboarding vs. Tomasulo

Práctico Nro. 5: Arquitecturas vector
Set de instrucciones Vector. Performance de los procesadores vectos. Rutinas para manipular operandos vector.

Práctico Nro. 6: Distintas organizaciones de memoria cache. Técnicas de ubicaciones de bloques: mapeo directo, conjuntos asociativos, memoria completamente asociativa. Pasos en la lectura y escritura de la memoria cache. Reduciendo los miss de cache: Comparación entre diferentes arquitecturas con memoria cache de diferentes tamaños.

VIII - Regimen de Aprobación

* Regularización
Para regularizar la materia el alumno deberá cumplir con los siguientes requisitos:
Aprobar un examen parcial, o su correspondiente recuperación sobre temas que abarcan las cuatro primeras unidades.
Aprobar un trabajo práctico que cubre la última unidad de la materia. Este trabajo puede ser defendido y presentado en grupos de hasta 3 alumnos.
Aquellos alumnos que estén en condiciones tendrán derecho a una recuperación adicional por trabajo.
Mostrar compromiso con la materia a través de asistencia regular a clase y realización de los prácticos de alula.

* Examen Final
Los alumnos regulares deberán rendir un examen final (que podrá ser oral o escrito) que consistirá en preguntas sobre los temas desarrollados durante el dictado de la materia.
* Alumnos libres
Los alumnos que desean rendir libre la materia se deberán poner en contacto con la cátedra con 5 días de anticipación a los efectos de realizar un práctico, el cual contendrá ejercicios similares a los desarrollados en los prácticos durante el dictado de la materia. Aprobando éste trabajo práctico el alumno tendrá derecho a rendir un examen oral con iguales características que el de los alumnos regulares.

IX - Bibliografía Básica

[1] Computer architecture a quantitative approach. John Hennessy y David Patterson. 3nd Edition. Editorial Morgan Kaufmann. (2003)
[2] Computer Organization & design. David Patterson and Jhon L. Hennessy. 2nd Edition. Editorial Morgan Kaufmann. (1998)
[3] Computer Architecture design and performance. Barry Wilkinson.Editorial Prentice-Hall (1996)

X - Bibliografia Complementaria

[1] Roland N. Ibbett \\\"HASE DLX Simulation Model,\\\" IEEE Micro, Vol 20, no 3, p 57-65, 2000.
[2] P.M. Sailer & D.R. Kaeli \\\"The DLX Instruction Set Architecture
[3] Handbook\\\" Morgan Kaufmann, 1996.
[4] P.S. Coe, F.W. Howell, R.N. Ibbett and L.M. Williams \\\"A Hierarchical
[5] Computer Architecture Design and Simulation Environment\\\" ACM Transactions on Modeling and Computer Simulation vol. 8, no. 4 (October 1998)
[6] James E. Smith \"Characterizing computer performance with a single number\"Communications of the ACM. Oct. 1988

Vol 32 Nro 10.

XI - Resumen de Objetivos

Al finalizar el curso los alumnos deberan poder:

*Identificar los principales parámetros que miden el desempeño de los procesadores.

*Desarrollar diferentes analisis cuantitativos del desempeño de procesadores provistos con memoria cache.

*Conocer las características de los proccesadores pipeline y las particularidades consideraras al medir el desempeño de los mismos.

*Comprender el funcionamiento general de los procesadores de arreglo y las particularidades de los problemas que se adecuan a estas arquitecturas.

*Describir el modo en que se realizan las computaciones en los sistemas dataflow.

*Percibir la complejidad requerida para computar soluciones en sistemas de multiprocesadores y las redes interconexión necesarias en estos sistemas.

XII - Resumen del Programa

Medidas de performance. Jerarquía de memorias. Memorias cache. Procesadores pipeline. Tipos de hazards. Performance en sistemas pipelining. Impacto de los branches en el pipe. Pipeline multiciclo. Latencia e intervalo de iniciación. Arquitecturas data-flow. Planificación dinámica de instrucciones. Multiprocesadores. Arquitecturas UMA y NUMA. Coherencia de cache. Nivel de paralelismo de los programas. Sistemas de interconexión.

XIII - Imprevistos

Comunicarse con la cátedra.
Arquitectura del Procesador II.
Departamento de Informática.
Facultad de Cs. Físico, Matemáticas y Naturales.
Universidad Nacional de San Luis.
Ejército de los Andes 950. CP 5700.

ELEVACIÓN y APROBACIÓN DE ESTE PROGRAMA	
	Profesor Responsable
Firma:	
Aclaración:	
Fecha:	