

Universidad
Nacional
de Córdoba



FAMAF
Facultad de Matemática,
Astronomía y Física

EXP-UNC: 49517/2015

Resolución CD N° 361/2015

PROGRAMA DE ASIGNATURA

ASIGNATURA: Arquitectura de Computadoras	AÑO: 2015
CARÁCTER: Obligatoria	
CARRERA: Licenciatura en Ciencias de la Computación	
RÉGIMEN: Cuatrimestral	CARGA HORARIA: 120 hs.
UBICACIÓN en la CARRERA: 3er Año – 2do Cuatrimestre	

FUNDAMENTACIÓN Y OBJETIVOS

Que el alumno sea capaz de interpretar el funcionamiento de los bloques "internos" asociados a Arquitectura de Computadoras No Convencionales (No "Von Neumann", Procesadores de Alta Prestación y Computadoras Reconfigurables).

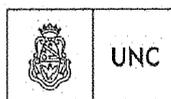
CONTENIDO

-Unidad 1: Computación SISD ("Single Instruction, Single Data")

- 1.1.-Arquitecturas tipo SISD, subtipo RISC (Reduced Instruction Set Computer).
- 1.2.-Arquitecturas tipo SISD, subtipo CISC (Complex Instruction Set Computer).
- 1.3.-Ejemplos de Arquitecturas SISD tipo No-Von Neuman.
- 1.4.- Concepto de Segmentación Encausada, (Pipe-Line).
- 1.5.- Ejemplos de Arquitecturas SISD, RISC.
- 1.6.- Ejemplos de Arquitecturas SISD, CISC.
- 1.7.- Arquitectura y Set de Instrucciones de Procesadores con Pipe-line.
- 1.8.- Ejemplos de Procesadores con Pipe-Line.
- 1.9.- Organización, Jerarquía y Administración de Memorias en Sistemas con Pipe-Line.
- 1.10.- Arquitectura y Set de Instrucciones de Procesadores Vectoriales.
- 1.11.- Características de los lenguajes para procesamiento Vectorial.
- 1.12.- Características de los Compiladores para Procesadores Vectoriales.
- 1.13.- Ejemplos de Procesadores con Segmentación Encausada y Vectoriales.
- 1.14.- Caso de estudio práctico para integración de conceptos.

-Unidad 2: Computación SIMD ("Single Instruction, Multiple Data")

- 2.1.-Procesadores Matriciales o SIMD. Concepto.
- 2.2.-Arquitecturas de los Procesadores SISD.
- 2.3.-Redes de Interconexión.
- 2.4.- Algoritmos para procesadores SIMD. Ejemplos.
- 2.5.- Procesadores SIMD Asociativos.
- 2.6.- Memorias Asociativas.
- 2.7.- Algoritmos para procesadores SIMD, Asociativos.
- 2.8.- GPGPU (General Purpose Graphic Processor Unit) y Computación heterogénea: Origen y Conceptos.
- 2.9.- Implementación: OpenCL (Open Computer Language)
- 2.10.- Arquitectura y Modelo de Plataforma.
- 2.11.- Modelo de Ejecución y de Memoria.
- 2.12.- Lenguaje e Interfaces.
- 2.13.- Operaciones con Matrices en OpenCL
- 2.14.- Caso de estudio Práctico para integración de conceptos.



FAMAF
Facultad de Matemática,
Astronomía y Física

EXP-UNC: 49517/2015

Resolución CD N° 361/2015

-Unidad 3: Computación MIMD (Multiple Instruction, Multiple Data) y de Flujo de Datos

- 3.1.-Computadores MIMD ligeramente acoplados.
- 3.2.-Computadores MIMD estrechamente acoplados.
- 3.3.-Distintos tipos de Buses y Redes de Interconexión.
- 3.4.-Estructuración y Organización de la Memoria.
- 3.5.-La Problemática de los Sistemas Multicaché.
- 3.6.-Características de los S.O. para Sistemas MIMD.
- 3.7.-Algoritmos en Procesadores SIMD.
- 3.8.- Ejemplos de sistemas MIMD.
- 3.9.- Nociones de computadores de Flujo de Datos.
- 3.10.- Arquitecturas de computadoras de Flujo de Datos.
- 3.11.- Casos de estudio de ejemplo.

-Unidad 4: Nociones de Computación Reconfigurable (C. R.) y de Alta Performance (HPC)

- 4.1.-Conceptos generales, historia y estado del arte de la C. R.
- 4.2.-El uso de HDL en computación reconfigurable.
- 4.3.-Nociones de Codiseño Hardware-Software y su aplicación en C.R.
- 4.4.- Conceptos generales de HPC.
- 4.5.- Historia y estado del arte HPC.
- 4.6.- Nociones de HPC y distribuida. Nociones básicas de Clusters.
- 4.7.- Nociones básicas de Arquitecturas Grid.

BIBLIOGRAFÍA

- Arquitectura de Computadoras y Procesamiento Paralelo. Kai Hwang y Faye A. Briggs. Mc Graw-Hill (1988).
- David A. Patterson and John L Hennessy: "Computer Organization and Design – The Hardware/Software Interface". Fourth Edition. Elsevier – Morgan Kaufmann (ISBN 978-0-12-374493-7).
- John L Hennessy and David A. Patterson: "Computer Architecture – A quantitative Approach". Fourth Edition. Elsevier – Morgan Kaufmann.
- Volnei. Pedroni. Circuit Design Using VHDL. MIT Press, Cambridge, Massachusetts, 2004.
- Douglas Perry. VHDL: Programming by Example. Mc. Graw Hill, NY, 2002.
- Enoch Hwang. Microprocessor Design: Principles and Practices with VHDL. Brooks/Coole. 2004.

METODOLOGÍA DE TRABAJO

CLASES TEÓRICAS: Cuatro horas semanales.

CLASES PRÁCTICAS: Cuatro horas semanales. Corresponden a prácticas de laboratorio en donde el alumno resolverá problemas utilizando el centro de cómputo.

EVALUACIÓN

FORMAS DE EVALUACIÓN

-Examen final teórico/práctico escrito. Para aprobar deberán obtener como mínimo un 40%



Universidad
Nacional
de Córdoba



FAMA F
Facultad de Matemática,
Astronomía y Física

EXP-UNC: 49517/2015
los regulares y un 60% los libres.

Resolución CD N° 361/2015

CONDICIONES PARA OBTENER LA REGULARIDAD

-Se toman tres (3) parciales teóricos/prácticos que deberán aprobar con un mínimo del 40%, pudiéndose recuperar los tres.

CONDICIONES PARA OBTENER LA PROMOCIÓN

-Deberán aprobar los tres parciales ya mencionados con un mínimo del 60%, pudiéndose recuperar los tres. Además deberán tener un promedio mínimo del 70%, no teniéndose en cuenta en el mencionado promedio la calificación del parcial recuperado ya que, en su lugar se usará la nota obtenida en el recuperatorio. El promedio que se trata será la calificación del examen final.

TA