



EX-2022-00597456- -UNC-ME#FAMAF

PROGRAMA DE ASIGNATURA	
<b>ASIGNATURA:</b> Arquitectura de Computadoras	<b>AÑO:</b> 2022
<b>CARACTER:</b> Obligatoria	<b>UBICACIÓN EN LA CARRERA:</b> 3° año 2° cuatrimestre
<b>CARRERA:</b> Licenciatura en Ciencias de la Computación	
<b>REGIMEN:</b> Cuatrimestral	<b>CARGA HORARIA:</b> 120 horas

#### 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.

##### -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.



EX-2022-00597456- -UNC-ME#FAMAF

- 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**

##### **BIBLIOGRAFÍA BÁSICA**

- 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). (2009)
- John L Hennessy and David A. Patterson: "Computer Architecture – A quantitative Approach". Fourth Edition. Elsevier – Morgan Kaufmann. (2007)
- Volnei. Pedroni. Circuit Design Using VHDL. MIT Press, Cambridge, Massachusetts, 2004.

##### **BIBLIOGRAFÍA COMPLEMENTARIA**

- Arquitectura de Computadoras y Procesamiento Paralelo. Kai Hwang y Faye A. Briggs. Mc Graw-Hill (1988).
- Douglas Perry. VHDL: Programming by Example. Mc. Graw Hill, NY, 2002.
- Enoch Hwang. Microprocessor Design: Principles and Practices with VHDL. Brooks/Coole. 2004.

#### **EVALUACIÓN**

##### **FORMAS DE EVALUACIÓN**

- 1. Se tomarán dos exámenes parciales escritos sobre temas teórico-prácticos de la materia. Para cada parcial habrá una instancia de recuperación. Se arpeban con 4 (cuatro).
- 2. Se realizarán dos trabajos prácticos o de laboratorio obligatorios.
- 3. El examen final para los alumnos regulares y libres será escrito, sobre temas teórico-prácticos de la materia y podrá incluir una instancia oral, sobre temas de los laboratorios de la materia.

##### **REGULARIDAD**

- 1. Aprobar al menos dos evaluaciones parciales o sus correspondientes recuperatorios.
- 2. Aprobar al menos el 60% de los trabajos prácticos o de laboratorio.

##### **PROMOCIÓN**

- 1. Aprobar todas las evaluaciones parciales con una nota no menor a 6 (seis), y obteniendo un promedio no menor a 7 (siete).
- 2. Aprobar todos los trabajos prácticos o de laboratorio.