



UNIVERSIDAD NACIONAL DE RÍO CUARTO

FACULTAD DE CIENCIAS EXACTAS, FÍSICO-QUÍMICAS Y NATURALES

DEPARTAMENTO DE COMPUTACIÓN

CARRERA/S: LICENCIATURA EN CIENCIAS DE LA COMPUTACIÓN.

PLAN DE ESTUDIOS: 1999

ASIGNATURA: Organización del Procesador

CÓDIGO: 1949

DOCENTE RESPONSABLE: Dr. Germán Regis

EQUIPO DOCENTE: Ing. Guillermo Frascchetti, Lic. Tardivo Laura

AÑO ACADÉMICO: 2019

REGIMEN DE LA ASIGNATURA: Cuatrimestral (2do cuatrimestre)

RÉGIMEN DE CORRELATIVIDADES: (para cursado)

<i>Aprobada</i>	<i>Regular</i>
	Introducción a la algoritmica y Programacion - 3300

CARGA HORARIA TOTAL: 112

TEÓRICO: 56 hs **PRÁCTICAS-LABORATORIO:** 56 hs

CARÁCTER DE LA ASIGNATURA: Obligatoria

A. CONTEXTUALIZACIÓN DE LA ASIGNATURA

La asignatura se dicta en el 2 año de las carreras de Ciencias de la Computación.

B. OBJETIVOS PROPUESTOS

- Adquirir conocimientos básicos sobre los componentes electrónicos que conforman la base de los procesadores/arquitecturas

- Adquirir conocimientos sobre arquitecturas y diseño de sistemas de computadoras. Especialmente la arquitectura Von Neumann.
- Utilización de lenguajes de bajo nivel (ensamblador) de máquinas basadas en un procesador genérico de 8 bits, práctica en un procesador Intel de 32 bits y Arduino (Atmega328p).

C. CONTENIDOS BÁSICOS DEL PROGRAMA A DESARROLLAR

Unidades funcionales básicas de una computadora y diferentes maneras de estructurarlas (organizarlas). Circuitos operacionales básicos. Métodos de direccionamiento, secuenciamiento de programas y set de instrucciones. Control microprogramado. Organización de entrada/salida. Aritmética entera y flotante. Memoria entrelazada, virtual y cache. Paralelismo y pipelining.

Los contenidos se especifican en el programa analítico.

D. FUNDAMENTACIÓN DE LOS CONTENIDOS

La informática es una disciplina que incluye diversas técnicas y actividades relacionadas con el tratamiento automático y lógico de la información. Ha adquirido un enorme desarrollo gracias a las computadoras. Son precisamente estas con sus diseños las que determinan y muchas veces modifican el comportamiento de dichas actividades. Es necesario entender cómo funcionan internamente para poder obtener todas sus capacidades e interpretar correctamente todas sus limitaciones.

La arquitectura Von Neumann con procesadores Intel es la más difundida. Las arquitecturas paralelas por otra parte, o bien sus técnicas de procesamiento, se están convirtiendo en un standard actual para nuestra profesión.

E. ACTIVIDADES A DESARROLLAR

(Consiguar las formas metodológicas)

Se dictan las clases teóricas a todos los alumnos que cursan la materia, con un total de 4hs semanales.

Se divide el total de alumnos en dos comisiones de prácticos, asistiendo a clase dos veces por semana de 2hs cada una. En un plazo de aproximadamente un mes realizan un proyecto en assembler de IA-32.

CLASES TEÓRICAS: Presencial en aula, 56 Hs totales

CLASES PRÁCTICAS: Presencial en sala de maquinas, 56 Hs totales

F. NÓMINA DE TRABAJOS PRÁCTICOS

- 1) Sistemas Numéricos, Circuitos, Números Enteros, Números Reales
- 2) Introducción a Assembler, Modos de direccionamiento y Subrutinas
- 3) Microprogramas
- 4) Memoria Principal y Cache

G. HORARIOS DE CLASES:

HORARIO DE CLASES DE CONSULTAS:

Responsable: Martes 16.00 hs.

Auxiliares: Miércoles 10 hs, Martes 16 hs.

H. MODALIDAD DE EVALUACIÓN:

- **EVALUACIONES PARCIALES:** 2 exámenes parciales escritos sobre el práctico de la materia, con recuperación.
- **EVALUACIONES LABORATORIO:** 1 ejercicio de evaluación en laboratorio.
- **EVALUACIÓN FINAL:** Los exámenes finales son escritos u orales sobre la teoría y práctica de la materia.
- **CONDICIONES DE REGULARIDAD:** Aprobar los dos exámenes prácticos (parciales); aprobar un examen de laboratorio y el proyecto final.
- **CONDICIONES DE PROMOCIÓN:** No tiene.

I. PROGRAMA ANALÍTICO

CONTENIDOS

Unidad 1.

Introducción a la asignatura y el contexto histórico del desarrollo de dispositivos/sistemas computacionales. Introducción básica a la electrónica: Resistencia, Voltaje, Amperaje; componentes eléctricos, diodos, transistores, capacitores; compuertas lógicas electrónicas, Flip-Flop. Estructura básica de un sistema de cómputos. Arquitectura Von Neumann, unidades funcionales; estructuras de base. Ciclo completo de instrucción y secuenciamiento de programa.

Unidad 2.

Unidad Aritmética. Representación de números; adición y sustracción de números enteros positivos y negativos. Diseño lógico. Multiplicación y división de números enteros. Números y operaciones en punto flotante.

Unidad 3.

Unidad central de proceso. Conceptos y definiciones fundamentales. Registros. Microoperaciones, Microprogramación. Especificación de instrucciones. Descripción de la ejecución de una instrucción completa. Secuenciamiento de las señales de control. Unidad de control por hardware. Conjunto de instrucciones y modos de direccionamiento. Formato de instrucciones: 1 operando, 2 operandos, 3 operandos. Distintos tipos de instrucciones. Modos de direccionamiento.

Unidad 4.

Memoria Principal. Conceptos y definiciones básicas. Clasificación. Celda de memoria; distintas organizaciones. Estructura de un banco de memoria. Memorias RAM y ROM. Memorias entrelazadas y de múltiples módulos; necesidad del manejo de memoria: memoria virtual, cache y asociativa.

Unidad 5.

Organización de entrada-salida. Direccionamiento de dispositivos de E/S. Transferencia de datos. Sincronización. Manejo de interrupciones. Interface de E/S: canales. Dispositivos de E/S. Dispositivos de almacenamiento.

Unidad 6.

Comunicación de datos: transmisión de datos digitales; sincrónica y síncrona. Distintos tipos de enlaces: simplex, semi-duplex (half duplex), duplex (full duplex). Manejo de errores; procedimientos de recuperación; protocolos.

Unidad 7.

Multiprogramación y procesamiento: introducción. Conceptos básicos de paralelismo. Paralelismo temporal: organizaciones Pipeline; Paralelismo espacial. Clasificación de Flynn: SISD, SIMD, MISD, MIMD. Conceptos de Arquitecturas Reconfigurables. FPGAs.

J. CRONOGRAMA DE CLASES Y PARCIALES

Semana	Fecha	Teóricos/ práctico	Fecha	Prácticos	Fecha	L a b	Parciales / Recuper.
1	13/08/19	Unidad 1					
1	15/08/19	Unidad 1		Guía Práctica 1			
2	20/08/19	Unidad 2	20/08/19	Guía Práctica 1			
2	22/08/19	Unidad 2	22/08/19	Guía Práctica 1			
3	27/08/19	Unidad 2	27/08/19	Guía Práctica 1			
3	29/08/19	Unidad 3	29/08/19	Guía Práctica 1			
4	03/09/19	Unidad 3	03/09/19	Guía Práctica 2			
4	05/09/19	Unidad 3	05/09/19	Guía Práctica 2			
5	10/09/19	Unidad 3	10/09/19	Guía Práctica 2			
5	12/09/19	Unidad 3	12/09/19	Guía Práctica 2			
6	17/09/19	Consulta	17/09/19	Guía Práctica 2			
6					19/09/19		1er parcial
7	24/09/19	Integración					
7	26/09/19	Repaso			27/09/19		1er Recup.
8	01/10/19	Unidad 4	01/10/19	Guía Práctica 2			
8	03/10/19	Unidad 4	03/10/19	Guía Práctica 3			
9	08/10/19	Unidad 4	08/10/19	Guía Práctica 3			
9	10/10/19	Unidad 5	10/10/19	Guía Práctica 3			
10	15/10/19	Unidad 5	15/10/19	Guía Práctica 3			
10	17/10/19	Unidad 6	17/10/19	Guía Práctica 4			
11	22/10/19	Unidad 6	22/10/19	Guía Práctica 4			
11	24/10/19	Unidad 6	24/10/19	Guía Práctica 4			
12	29/10/19	Unidad 7	29/10/19	Guía Práctica 4			
12	31/10/19	Unidad 7	31/10/19	Guía Práctica 4			
13	05/11/19	Repaso	05/11/19	Repaso			
13					7/11/19		2do parcial
14	12/11/19	Proyecto	12/11/19	Repaso			
14					13/11/19		Recuperatorio

(Recordar las fechas de parciales deberán ser consensuadas con los responsables de las demás asignaturas del cuatrimestre correspondiente, en acuerdo con la Res. C.S. 356/10)

K. BIBLIOGRFÍA

- Charles Petzold. *Code: The Hidden Language of Computer Hardware and Software*, 978-0735611313, 2000.
- Linda Null, Julia Lobur. *The Essentials of Computer Organization and Architecture 4th Edition*, 978-1284045611, 2014.
- Carl Hamacher and Zvonko Vranesic. *Computer Organization and Embedded Systems*, 978-0073380650, 2011.
- Thomas L. Floyd. *Digital Fundamentals (10th Edition)*, 978-0132359238, 2008.
- Paul Scherz, Simon Monk. *Practical Electronics for Inventors*, Fourth Edition. 978-1259587542, 2016.
- Tanenbaum. *Structured Computer Organization (6 edition)*, 978-8120347205. 2013
- Paul A. Carter. *PC Assembly Language*. www.scs.stanford.edu