



Ministerio de Cultura y Educación  
 Universidad Nacional de San Luis  
 Facultad de Ciencias Físico Matemáticas y Naturales  
 Departamento: Informatica  
 Area: Area II: Sistemas de Computacion

(Programa del año 2026)  
 (Programa en trámite de aprobación)  
 (Presentado el 07/05/2026 10:24:52)

### I - Oferta Académica

Materia	Carrera	Plan	Año	Período
ORGANIZACIÓN Y ARQUITECTURA DE COMPUTADORAS	ING. INFORM.	OCD-3-2/2025	2026	1° cuatrimestre

### II - Equipo Docente

Docente	Función	Cargo	Dedicación
ARROYUELO BILLIARDI, JORGE A.	Prof. Responsable	P.Adj Exc	40 Hs
ARROYUELO, MONICA DEL VALLE	Responsable de Práctico	JTP Exc	40 Hs
BAGLIONI, VALENTINA	Auxiliar de Práctico	A.1ra Simp	10 Hs

### III - Características del Curso

Credito Horario Semanal				
Teórico/Práctico	Teóricas	Prácticas de Aula	Práct. de lab/ camp/ Resid/ PIP, etc.	Total
Hs	1 Hs	2 Hs	3 Hs	6 Hs

Tipificación	Periodo
B - Teoria con prácticas de aula y laboratorio	1° Cuatrimestre

Duración			
Desde	Hasta	Cantidad de Semanas	Cantidad de Horas
11/03/2026	23/06/2026	15	90

### IV - Fundamentación

En esta asignatura se aborda el estudio de la organización y arquitecturas de computadoras, tomando como punto de partida los procesadores secuenciales de tipo Von Neumann.

El alumno será involucrado en:

- \*Problemas de representación de los datos que serán manipulados por el procesador y la organización de los componentes principales del procesador que permiten procesar, almacenar y transferir información.
- \*Comprender cuál es el conjunto de instrucciones provistas habitualmente por los procesadores y sus modos de direccionamiento.
- \*Como se controla la realización de estas instrucciones a medida que transcurre el tiempo y como medimos su desempeño.
- \*Como un sistema de computación se comunica mediante dispositivos de entrada/salida.

### V - Objetivos / Resultados de Aprendizaje

- \*Aprender como representar datos y manipularlos en un sistema de computación.
- \*Comprender como están diseñados los procesadores secuenciales y como es su ciclo de instrucción.
- \*Desarrollar una actitud crítica frente al diseño de distintos procesadores.
- \*Obtener experiencia en programación de bajo nivel. Comprender como interactúan los procesadores con su medio externo.

- Identificar, formular y resolver problemas de informática.
- Utilizar técnicas y herramientas de aplicación en la informática.
- Auspiciar la comunicación efectiva.
- Fundamentar la acción ética y responsable.
- Aprender nociones para el aprendizaje continuo.

## VI - Contenidos

### PROGRAMA ANALÍTICO Y DE EXAMEN

#### Contenidos mínimos:

Sistemas Numéricos. Representación y aritmética de números enteros y fraccionarios. Organización Básica de una Computadora. Camino de datos (Datapath), Unidad central de procesamiento (CPU) y Memoria. Ciclo de instrucción. Conjunto de instrucciones de un procesador, formatos y modos de direccionamiento. Lenguaje ensamblador, ensamblador de 2 pasadas y programación. subrutinas y registros de activación. Entrada /Salida. Interfaz con la CPU. Protocolos de entrada/Salida. Interrupciones. Excepciones. DMA (Acceso directo a Memoria). Medidas de performance. Concepto de Pipeline. Concepto de Jerarquía de memoria.

#### Unidad 1: Sistemas numéricos y Aritmética.

Sistemas numéricos posicionales. Sistema binario, hexadecimal y octal. Conversión de decimal a binario y viceversa. Operaciones aritméticas en binario. Representación signo/magnitud: representación y operaciones. Complemento a dos: representación y suma. Rango de representación en los distintos sistemas. Punto punto flotante Estándar IEEE 754: representación y operaciones.

#### Unidad 2: Organización y Funcionamiento de una computadora.

Maquinas de Von Neumann. Memorias. Elementos de procesamiento, control, estado y caminos de datos. Conjunto de instrucciones. Ciclo de instrucción. Acceso a datos en memoria. Modos de direccionamiento: conceptos generales. Modo registro, absoluto, inmediato, registro indirecto. Modo de direccionamiento de múltiples componentes: indexado, relativos: a una base y a la próxima instrucción.

#### Unidad 3:Arquitectura MIPS.

Lenguaje Assembly. Conjunto de Instrucciones. Datos estructurado: arreglos, registros y listas vinculadas. Estructuras de control: selección, iteración, selección por caso. Llamadas a subrutina y retornos de subrutina. Pasaje de parámetros a las subrutinas: por valor, por dirección. Formas de pasaje de parámetros: en registros, en la pila. Ejemplos en el procesador MIPS.

#### Unidad 4:Entrada/Salida.

Características generales de la entrada/salida (I/O) y los sistema de interconexión (buses). Módulos de entrada/salida: interfaz con la CPU e interfaz con el dispositivo. Protocolos de entrada/salida. Entrada/Salida programada. Organización de la entrada/salida: dedicada y embebida en el espacio de memoria.

#### Unidad 5: Pipeline y Desempeño

Concepto de pipeline. conflictos en la ejecución de instrucciones: datos, control y estructurales. Medidas de performance: cpu\_time.

#### unidad 6: Jerarquía de Memoria

Concepto general de jerarquía de memoria. Memorias cache. Organización de memorias Cache.

#### Unidad N° 7: Interrupciones y DMA (Direct Memory Access)

Concepto de multiprogramación. Problemas presentados por la multiprogramación. Cambios de control necesarios para soportar multiprogramación: Interrupciones, despacho y llamadas al sistema. Excepciones. Condiciones asíncronas y síncronas. Atención de una interrupción asíncrona. Finalización de una interrupción síncrona. Manejo típico de una interrupción. Permiso para interrumpir: prioridades y deshabilitación de interrupciones. DMA.

## VII - Plan de Trabajos Prácticos

Los trabajos prácticos consisten en problemas cuya resolución requiere la aplicación de los conceptos desarrollados en clases teóricas. En ellos se incluyen ejercicios de aula, laboratorio e investigación.

El desarrollo de los trabajos prácticos de aula y laboratorio se llevan a cabo mayormente en el aula/sala de maquinas, en los horarios previstos para las clases prácticas, en los cuales los estudiantes son guiados por los docentes mediante la explicación en pizarrón y discusión grupal de soluciones y el intercambio de conclusiones a las que arriban.

Por otra parte, se ofrece a los estudiantes la posibilidad de reforzar lo visto en clases mediante el acceso digital a guías de estudio, autoevaluaciones en plataforma de aula virtual, archivos complementarios sobre los principales conceptos teóricos y ejemplos.

Los ejercicios de laboratorios constan de problemas desarrollarse en computadoras, con el objetivo de plasmar los conceptos vistos tanto en las teorías como en los prácticos de aula. Para las unidades de programación en lenguaje Assembly se utilizan simuladores gratuitos que pueden disponerse de forma online o descargarse en computadoras y/o dispositivos móviles. Estos simuladores son fáciles de aprender y muy útiles para visualizar la ejecución de instrucciones o programas en el procesador utilizado como caso de estudio.

Finalmente, en los ejercicios de investigación se propone a los estudiantes investigar sobre temas breves, algunos de los cuales serán evaluados en exámenes parciales, fomentando en ellos una actitud de responsabilidad para realizar el esfuerzo necesario por aprenderlos lo mejor posible cumpliendo con los tiempos pautados. Con este tipo de actividades se busca también generar en ellos autoconfianza y autonomía en el aprendizaje continuo.

### Prácticos de la Unidad N° 1

Representación de números. sistema binario, octal y hexadecimal. Representación en binario con bit de signo. Representación en sistemas de complemento. Representación en exceso. Operaciones aritméticas en los distintos sistemas de representación. Detección de desborde en los distintos sistemas de representación. Código ASCII. Representación y aritmética de números reales en punto flotante, rango de representación.

Laboratorios usando lenguaje C. Con esta actividad se evalúa la capacidad del alumno de identificar, formular y resolver problemas de informática, el uso de conocimientos adquiridos en la disciplina matemática aplicados a la informática y de adquirir nociones para el aprendizaje continuo.

### Prácticos de la Unidad N° 2

Arquitectura y programación de máquinas. Características generales de la arquitectura del MIPS. Organización de la memoria. Familiarización con el set de instrucciones del MIPS y pseudo instrucciones. Programas en lenguaje assembly simples. Instrucciones aritméticas y lógicas. Instrucciones de control condicional.

Laboratorios en MIPSim y webMIPS.

Esta actividad permite evaluar la capacidad del alumno de, utilizar técnicas y herramientas de aplicación en la informática; identificar, formular y resolver problemas de informática. Además se evalúa la capacidad de adquirir nociones para el aprendizaje continuo y la comunicación efectiva.

### Prácticos de la Unidad N° 3

Realizar programas en los que se usen los distintos modos de direccionamiento: inmediato, absoluto, registro, base, indexado y relativos. Implementación de las estructuras de control if-then, if-then-else, case, while y repeat. Implementación de arreglos multidimensionales. Instrucciones de llamada a subrutina y de retorno de subrutina. Manejo de la pila.

Implementación en lenguaje assembly de pasaje de parámetros.

Laboratorios en MIPSim y webMIPS.

Esta actividad permite evaluar la capacidad del alumno de, utilizar técnicas y herramientas de aplicación en la informática; identificar, formular y resolver problemas de informática. Además se evalúa la capacidad de adquirir nociones para el

aprendizaje continuo y la comunicación efectiva.

#### Prácticos de la Unidad N° 4

Protocolo entre los dispositivos para hacer una entrada/salida programada. Programas que realicen entrada/salida en forma programada. Cuestionario sobre conceptos generales de entrada/salida

Esta actividad permite evaluar la capacidad del alumno de utilizar técnicas y herramientas de aplicación en la informática e identificar, formular y resolver problemas de informática. Además se evalúa la capacidad de adquirir nociones para el aprendizaje continuo y la comunicación efectiva.

#### Prácticos de la Unidad N° 5

Ejecución de códigos en arquitectura de Mips con pipeline. Detección de conflictos y soluciones.

Esta actividad permite evaluar la capacidad del alumno de, utilizar técnicas y herramientas de aplicación en la informática e identificar, formular y resolver problemas de informática. Además se evalúa la capacidad de adquirir nociones para el aprendizaje continuo y la comunicación efectiva.

#### Prácticos de la Unidad N° 6

Detección de situaciones de Hit y Miss en secuencia de direcciones a memoria.

Esta actividad permite evaluar la capacidad del alumno de, utilizar técnicas y herramientas de aplicación en la informática e identificar, formular y resolver problemas de informática. Además se evalúa la capacidad de aprender nociones para el aprendizaje continuo y la comunicación efectiva.

#### Prácticos de la Unidad N° 7

Programación con interrupciones y DMA.

Esta actividad permite evaluar la capacidad del alumno de, utilizar técnicas y herramientas de aplicación en la informática e identificar, formular y resolver problemas de informática. Además se evalúa la capacidad de adquirir nociones para el aprendizaje continuo y la comunicación efectiva.

Por cada unidad la catedra solicita la presentación de ejercicios prácticos con fecha limite, donde se evalúa además de los conceptos y temas de cada unidad la acción ética y responsable, cumpliendo plazos y entregando trabajos originales.

### **VIII - Regimen de Aprobación**

Condiciones para regularizar la materia:

- Presentar la totalidad de los ejercicios prácticos solicitados por la catedra en tiempo y forma.
- Aprobar El examen parcial o sus recuperaciones con nota mayor o igual a 6 (seis).
- Contar con el 70% de asistencia a clase.

Se otorga, tal como lo expresa la reglamentación vigente, dos recuperaciones para el examen parcial a todos los estudiantes.

Condiciones para promocionar la materia:

- Aprobar la totalidad de los ejercicios prácticos solicitados por la catedra, en tiempo y forma, con una nota mayor o igual a 7 (siete).
- Aprobar el examen parcial en primera instancia con nota mayor o igual a 7 (siete).
- Contar con el 80% de la asistencia a clase.

En caso de regularizar y no promocionar la asignatura, el alumno deberá rendir un examen final escrito u oral.

Alumnos libres:

No se permite rendir en carácter de alumno libre, dado que, durante el transcurso de la misma, el alumno deberá desarrollar prácticas de laboratorio solicitadas por la cátedra.

## **IX - Bibliografía Básica**

- [1] John L. Hennessy, David A. Patterson, Christos Kozyrakis. Computer Architecture: A Quantitative Approach. The Morgan Kaufmann Series in Computer Architecture and Design, 7th Edición. [2025].
- [2] David A. Patterson and John L. Hennessy. Computer Organization and Design MIPS Edition: The Hardware/Software Interface. The Morgan Kaufmann Series in Computer Architecture and Design, 6th Edición. [2020].
- [3] Sara Harris & David Money Harris. Digital Design and Computer Architecture. Ed. Morgan Kaufmann 2016.
- [4] WILLIAM STALLINGS. Computer Organization and Architecture: Designing for Performance. ED. PEARSON PRENTICE HALL [2010].
- [5] JHON F. WAKERLEY. Microcomputer Architecture and Programming. Ed. JOHN WILEY AND SONS [1981].
- [6] NIKLAUS WIRTH. Digital Circuit Design. An Introductory Textbook. Springer [1995].
- [7] HAYES J. Computer Architecture and Organization. ED. MC. GRAW HILL [1988].
- [8] WILLIAM STALLINGS. Computer Organization and Architecture fourth edition. ED. PRENTICE HALL [1996].
- [9] JOHN L. HENNESSY & DAVID PATTERSON. Computer Architecture: A Quantitative Approach. 2nd Edition. ED. MORGAN AND KAUFMANN [1990].
- [10] JEAN-PIERRE MAINADIER. Estructura y Funcionamiento de los Computadores Digitales. Editorial AC, Madrid [1980].
- [11] JHON CRAWFORD. Architecture of the Intel 80386. Readings in Computer Architecture, pag. 157-162. Edited by M. D. Hill, n. P. Jouppi and G. S. Sohi. ISBN 1-55860-539-8. [2000]
- [12] Dandamudi Sivarama P.. Introduction to Assembly Language Programming, for Pentium and RISC Processors. Springer. ISBN 0-387-20636-1. [2004].
- [13] Dandamudi Sivarama P.. Guide to Assembly Language Programming in Linux. Springer. ISBN 0-387-25897-3. [2005].

## **X - Bibliografía Complementaria**

- [1] GERRIT A. BLAAUW-FREDERICK P. BROOKS, Jr. Computer Architecture. Concepts and Evolution. ED. ADDISON-WESLEY. [1997].
- [2] PETER J. ASHENDEN. The Designer's Guide to VHDL (Second Edition). ED. Morgan Kaufmann Publishers [2002].
- [3] BEHROOZ PARHAMI Computer Arithmetic. Oxford University Press; 2da edición [2010].

## **XI - Resumen de Objetivos**

- \*Aprender como representar datos en una computadora
- \*Comprender como están diseñados los procesadores secuenciales y como es su ciclo de instrucción.
- \*Desarrollar una actitud crítica frente al diseño de distintos procesadores.
- \*Obtener experiencia en programación de bajo nivel. Comprender como interactúan los procesadores con su medio externo.

## **XII - Resumen del Programa**

Sistemas Numéricos. Representación y aritmética de números enteros y fraccionarios. Organización Básica de una Computadora. Camino de datos (Datapath), Unidad central de procesamiento (CPU) y Memoria. Ciclo de instrucción. Conjunto de instrucciones de un procesador, formatos y modos de direccionamiento. Lenguaje ensamblador, ensamblador de 2 pasadas y programación. subrutinas y registros de activación. Entrada /Salida. Interfaz con la CPU. Protocolos de entrada/Salida. Interrupciones. Excepciones. DMA (Acceso directo a Memoria). Medidas de performance. Concepto de Pipeline. Concepto de Jerarquía de memoria.

## **XIII - Imprevistos**

Comunicarse con la cátedra.  
Arquitectura del Procesador.

Departamento de Informática.  
Of. 22. Bloque II. 1er. Piso.  
Facultad de Cs. Físico, Matemáticas y Naturales.  
Universidad Nacional de San Luis.  
Ejército de los Andes 950. CP 5700.

#### **XIV - Otros**

Las vías de comunicación con los estudiantes son las siguientes:

Correos electrónicos de los docentes:

Jorge Arroyuelo <jarroyu@gmail.com>

Monica Arroyuelo <arroyuelom@gmail.com>

<b>ELEVACIÓN y APROBACIÓN DE ESTE PROGRAMA</b>	
	<b>Profesor Responsable</b>
Firma:	
Aclaración:	
Fecha:	