



Campus de la UNA
SAN LORENZO-PARAGUAY

UNIVERSIDAD NACIONAL DE ASUNCIÓN
FACULTAD POLITÉCNICA
CONSEJO DIRECTIVO

RESOLUCIÓN 25/18/38-00
ACTA 1226/25/08/2025

"POR LA CUAL SE APRUEBA EL PROGRAMA DE ESTUDIO DE LA ASIGNATURA FUNDAMENTOS Y ARQUITECTURA DE COMPUTADORAS, DE LA CARRERA LICENCIATURA EN CIENCIAS INFORMÁTICAS – PLAN 2026, SEDE SAN LORENZO, FILIAL VILLARRICA Y FILIAL CORONEL OVIEDO"

VISTO: El Memorando DA/1707/2025 del Director Académico de la FP-UNA, Prof. MSc. Felipe Santiago Uzabal Escurra, con el cual remite el Memorando CCPTCC/030/2025 de la Comisión Coordinadora del Proyecto de Transformación Curricular de Carreras de Grado de la FP-UNA, en el que presenta la propuesta de Programas de Estudio de las Asignaturas de la Carrera Licenciatura en Ciencias Informáticas.

CONSIDERANDO: La Ley 4995/2013 de Educación Superior, el Estatuto de la Universidad Nacional de Asunción y las deliberaciones sobre el tema.

Que la Comisión Coordinadora del Proyecto de Transformación Curricular de Carreras de Grado, solicita la aprobación del Programa de Estudio de la asignatura **"Fundamentos y Arquitectura de Computadoras"**, de la carrera Licenciatura en Ciencias Informáticas – Plan 2026.

Que los programas fueron elaborados conforme a las disposiciones establecidas por el Consejo Nacional de Educación Superior (CONES) en materia de **créditos académicos**, según lo dispuesto en la Resolución CONES N.º 221/2024, que regula el *Sistema de Créditos Académicos – Paraguay* y los criterios para su publicación en las carreras de grado.

**EL CONSEJO DIRECTIVO DE LA FACULTAD POLITÉCNICA
RESUELVE:**

25/18/38-01 APROBAR el Programa de Estudio de la Asignatura **"Fundamentos y Arquitectura de Computadoras"**, de la carrera Licenciatura en Ciencias Informáticas – Plan 2026, Sede San Lorenzo, Filial Villarrica y Filial Coronel Oviedo, detallado en el ANEXO 03 de la presente Acta.

25/18/38-02 COMUNICAR, copiar y archivar.

Prof. Abg. Joel Arsenio Benítez Santacruz

Secretario

Prof. Ing. Silvia Teresa Leiva León, MSc.

Presidenta





Campus de la UNA
SAN LORENZO-PARAGUAY

UNIVERSIDAD NACIONAL DE ASUNCIÓN
FACULTAD POLITÉCNICA
CONSEJO DIRECTIVO

Resolución 25/18/38-00 Acta 1226/25/08/2025
ANEXO 03

DEPARTAMENTO DE ENSEÑANZA DE INFORMÁTICA
PROGRAMA DE ESTUDIO

I. IDENTIFICACIÓN

Nivel				Grado											
Asignatura				Fundamentos y Arquitectura de Computadoras											
Carrera				Plan		Sede/Filial		Carácter		Semestre		Prerrequisitos			
Licenciatura en Ciencias Informáticas				2026		Sede San Lorenzo/Filial Villarrica/Filial Coronel Oviedo		Obligatoria		Primero		Ninguno.			
Semanal					Periodo										
HT		HP		HTD	HTI	HS	PA	THTD		THTI		THA		CA-PY	
2		2		4	4	8	18	72		72		144		5	

- *HT: Horas Teóricas semanales.
- *HP: Horas Prácticas semanales.
- *HTD: Horas semanales de Trabajo académico con acompañamiento Docente.
- *HTI: Horas semanales de Trabajo académico Independiente del estudiante.
- *HS: Horas Semanales (HTD+HTI).
- *PA: Periodo Académico en semanas.
- *THTD: Total de Horas de Trabajo académico con acompañamiento Docente (HTD*PA).
- *THTI: Total de Horas de Trabajo académico Independiente del estudiante (HTI*PA).
- *THA: Total de Horas de trabajo Académico (THTD+THTI).
- *CA-PY: Créditos académicos de la asignatura

II. FUNDAMENTACIÓN

Fundamentos y Arquitectura de Computadoras es una asignatura esencial en de la formación de los estudiantes de la carrera de Ciencias Informáticas, al proporcionar las bases teóricas y prácticas para comprender cómo funcionan los sistemas computacionales, desde su estructura física hasta la interacción con el software. Este curso es clave para entender el papel fundamental que tiene el hardware en el rendimiento y funcionamiento de las aplicaciones informáticas modernas.

El curso presenta la terminología fundamental en cuanto a hardware, arquitectura y software, permitiendo a los estudiantes desarrollar un lenguaje técnico preciso. A continuación, se introduce la organización y arquitectura de computadoras, brindando una comprensión clara de los diferentes componentes que conforman un sistema de cómputo, sus interacciones y las implicaciones de diseño.

Uno de los ejes principales es el estudio del diseño y funcionamiento de la CPU, donde se analizan los ciclos de instrucción, las unidades de control y los componentes clave del procesador. También se estudian los sistemas de memoria, abordando las jerarquías de memoria y las estrategias de gestión de almacenamiento, como la memoria caché y la virtualización.

El manejo del almacenamiento y entrada/salida de datos es fundamental para comprender cómo los dispositivos de almacenamiento y los periféricos se comunican con el sistema y afectan su rendimiento. Además, se explora la arquitectura del conjunto de instrucciones (ISA), proporcionando una comprensión detallada de cómo las instrucciones son ejecutadas por el hardware.

La microarquitectura también es un tema crucial, analizando los niveles internos del procesador y cómo estos afectan el procesamiento de datos. A su vez, el curso profundiza en la arquitectura de sistemas a

nivel de hardware y software, explorando cómo ambos niveles se integran para optimizar el rendimiento de los sistemas.

Se analiza el rendimiento de la computadora, enseñando a los estudiantes a evaluar y optimizar los recursos computacionales para lograr un equilibrio entre costo, velocidad y eficiencia. Finalmente se da una introducción a arquitecturas avanzadas como los procesadores multinúcleo y los sistemas de procesamiento paralelo, estudiando sus aplicaciones y ventajas.

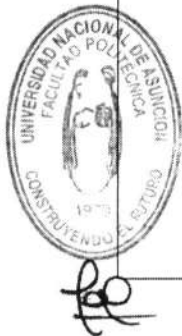
En conjunto, este curso no solo proporciona una base sólida para asignaturas más avanzadas, sino que también dota a los estudiantes de las competencias necesarias para enfrentar los desafíos en el diseño, desarrollo y optimización de sistemas computacionales modernos.

III. COMPETENCIAS DEL PERFIL DE EGRESO ASOCIADAS

- 1. Aplicar en la práctica profesional los valores humanos, la ética y los mecanismos de seguridad laboral.
- 2. Liderar y trabajar en equipo con eficacia y responsabilidad tomando decisiones basadas en evidencias.
- 3. Seleccionar, utilizar y construir instrumentos innovadores asociados al ejercicio de las ciencias informáticas.
- 4. Aplicar, producir y difundir conocimientos técnicos y científicos en el área de las ciencias informáticas.
- 5. Actualizarse permanentemente mediante la obtención y gestión autónoma de información de calidad, utilizando tecnología de la información y comunicación.

IV. ORGANIZACIÓN DE LA ASIGNATURA

Unidades	Contenidos	Resultados de aprendizaje
1. Introducción a los Sistemas Computacionales y los fundamentos de Organización y Arquitectura de Computadoras.	1.1. Definición y evolución de los sistemas computacionales. 1.2. Componentes fundamentales: hardware, software, y firmware. 1.3. Conceptos de organización y arquitectura de computadoras. Diferencias. 1.4. Modelo de Von Neumann y sus implicaciones en el diseño moderno. 1.5. Relación entre el hardware y el software: procesos de abstracción. 1.6. Clasificación de los sistemas computacionales: desde sistemas embebidos hasta supercomputadoras. 1.7. Representación de la información: sistemas	1. Explica terminologías y conceptos fundamentales relacionados con los sistemas computacionales, organización y arquitectura de computadoras, incluyendo los componentes de hardware y software. 2. Aplica sistemas de numeración para la representación de información. 3. Utiliza álgebra booleana en el diseño de sistemas digitales. 4. Diseña circuitos digitales básicos con compuertas lógicas.



Unidades	Contenidos	Resultados de aprendizaje
	binarios, hexadecimal, octal y de complemento a dos. 1.8. Álgebra booleana y su aplicación. 1.9. Lógica Combinacional: Compuertas lógicas y diseño de circuitos digitales reducidos. 1.10. Multiplexores, comparadores, decodificadores.	
2. Diseño y funcionamiento de la CPU.	2.1. Estructura interna de la CPU: unidad de control, unidad aritmética-lógica (ALU) y registros. 2.2. Ciclo de instrucción: fetch, decode, execute. 2.3. Pipelining: conceptos básicos y mejora del rendimiento. 2.4. Procesamiento de instrucciones en paralelo: multithreading, hyperthreading. 2.5. Tipos de CPU: CISC vs. RISC.	1. Identifica y describe los componentes de la CPU. 2. Explica el ciclo de instrucción y ejecución. 3. Analiza el concepto de pipelining y paralelismo.
3. Sistemas de memoria.	3.1. Jerarquía de memoria: registros, caché, RAM, almacenamiento secundario. 3.2. Características de la memoria: capacidad, velocidad, volatilidad y costo. 3.3. Memoria caché: principios de funcionamiento, niveles (L1, L2, L3). 3.4. Técnicas de gestión de memoria: paginación y segmentación. 3.5. Memoria virtual: concepto, implementación y ventajas.	1. Clasifica y explica los diferentes niveles de la jerarquía de memoria. 2. Diferencia entre los tipos de memoria. 3. Aplica principios de administración de memoria.
Almacenamiento y Entrada-Salida de Datos.	4.1. Dispositivos de almacenamiento: discos duros, SSD, memorias flash. 4.2. Arquitectura y	1. Evalúa y selecciona dispositivos de almacenamiento adecuados. 2. Describe y analiza métodos de entrada/salida (I/O) y almacenamiento de datos,



Unidades	Contenidos	Resultados de aprendizaje
	funcionamiento de los buses de datos. 4.3. Técnicas de transferencia de datos. 4.4. Interfaces de entrada y salida. 4.5. Controladores de dispositivos y sus interacciones con el procesador. 4.6. Sistemas de almacenamiento en red.	identificando los dispositivos más comunes y las técnicas de transferencia de datos. 3. Analiza sistemas de almacenamiento en red.
5. Arquitectura del Conjunto de Instrucciones (ISA).	5.1. Estructura y diseño del conjunto de instrucciones. 5.2. Tipos de instrucciones: aritméticas, lógicas, de control y de acceso a memoria. 5.3. Modos de direccionamiento. 5.4. Lenguaje ensamblador y su interacción con el hardware. 5.5. Ejemplos de ISAs comunes (x86, ARM).	1. Identifica y analiza las arquitecturas de conjunto de instrucciones (ISA), comprendiendo cómo las instrucciones son procesadas por el hardware y los diferentes modos de direccionamiento. 2. Relaciona la arquitectura de conjunto de instrucciones con el lenguaje de programación.
6. Microarquitectura.	6.1. Diseño a nivel de circuitos en la CPU. 6.2. Concepto de microinstrucciones. 6.3. Implementación de instrucciones a nivel de hardware.	1. Explica los principios de la microarquitectura y su impacto en la optimización del rendimiento de los sistemas computacionales.
7. Arquitectura de Sistemas a Nivel de Hardware y Software.	7.1. Interacción entre el hardware y el software en la ejecución de procesos. 7.2. Diseño de sistemas embebidos: características y optimización. 7.3. Arquitecturas de computadoras contemporáneas. 7.4. Sistemas operativos y su papel en la arquitectura. 7.5. Interfaces hardware-software: drivers y sistemas operativos. 7.6. Gestión de recursos a nivel de hardware y software:	1. Aplica conocimientos sobre la interacción entre hardware y software en la ejecución de tareas, describiendo cómo los sistemas operativos gestionan los recursos del hardware.



Unidades	Contenidos	Resultados de aprendizaje
	procesadores, memoria y dispositivos de I/O.	
8. Rendimiento de la computadora.	8.1. Definición y medición del rendimiento: throughput, latencia, tiempo de respuesta. 8.2. Factores que afectan el rendimiento: arquitectura, software, cargas de trabajo. 8.3. Técnicas de optimización del rendimiento: paralelismo, caching, predicción de saltos. 8.4. Evaluación del rendimiento mediante benchmarks. 8.5. Impacto del rendimiento en la eficiencia energética y sostenibilidad.	1. Evalúa y mide el rendimiento de una computadora, considerando factores como el throughput, la latencia y el uso eficiente de los recursos.
9. Introducción a Arquitecturas Avanzadas.	9.1. Procesadores multinúcleo y su funcionamiento. 9.2. Procesamiento paralelo y distribuido. 9.3. Arquitecturas GPU vs. CPU. 9.4. Conceptos básicos de computación cuántica y tendencias futuras en arquitectura.	2. Demuestra conocimiento de arquitecturas avanzadas, como los procesadores multinúcleo y los sistemas de procesamiento paralelo, comprendiendo sus aplicaciones y ventajas. 3. Compara arquitecturas de CPU y GPU, y describir conceptos emergentes en computación como la computación cuántica.

V. ESTRATEGIAS DIDÁCTICAS

El desarrollo del programa se basará en un enfoque teórico-práctico, combinando la exposición de conceptos con actividades que permitan la aplicación de los mismos. Se describen algunas estrategias:

- **Clases expositivas:** El docente presentará los conceptos teóricos fundamentales utilizando presentaciones, diagramas, y ejemplos prácticos que ilustran el funcionamiento de los sistemas computacionales. Estas sesiones se complementarán con discusiones para fomentar la participación activa de los estudiantes.
- **Resolución de problemas y ejercicios:** Se propondrán ejercicios prácticos donde los estudiantes aplicarán los conceptos vistos en clase.
- **Laboratorios prácticos:** Se realizarán sesiones de laboratorio donde los estudiantes tendrán la oportunidad de interactuar con simuladores de arquitectura de computadoras, realizar mediciones de rendimiento de sistemas. Estas actividades fomentarán el aprendizaje experiencial, permitiendo a los estudiantes experimentar directamente con los conceptos estudiados.



- **Estudios de casos:** Se analizarán casos de estudio sobre arquitecturas de computadoras reales, como procesadores x86 y ARM. Estos permitirán a los estudiantes relacionar la teoría con sistemas usados en la industria, y fomentar la capacidad crítica para evaluar decisiones de diseño.
- **Aprendizaje colaborativo:** Se fomentará el trabajo en equipo a través de actividades colaborativas, como la discusión de problemas complejos en pequeños grupos y el desarrollo de proyectos cortos como programación, configuración, simulación de los temas estudiados. Estas actividades promueven el intercambio de ideas y el aprendizaje mutuo.

La elección particular de la estrategia didáctica aplicada será explícita en el Planeamiento de la Asignatura, de acuerdo con el perfil de los estudiantes, los recursos disponibles y el contexto educativo.

VI. ESTRATEGIAS EVALUATIVAS

La evaluación del curso se estructurará en función de los resultados de aprendizaje esperados y las actividades de enseñanza propuestas. Se emplearán diversas estrategias de evaluación para medir tanto el conocimiento teórico como las habilidades prácticas de los estudiantes: evaluaciones escritas con preguntas de tipo ensayo, análisis de arquitecturas, y problemas prácticos sobre diseño de CPU, memoria y sistemas de I/O, tareas de laboratorio que consistirán en actividades prácticas, los estudiantes presentarán un proyecto integrador, donde deberán diseñar o analizar la arquitectura de un sistema computacional real o simulado, identificando oportunidades de mejora en rendimiento. Además, se evaluarán los trabajos colaborativos en los que los estudiantes deban resolver problemas en equipo o discutir casos de estudio.

Con fines de calificación y promoción se aplicará el Reglamento Académico vigente en la institución que prevé valoraciones de proceso y final.

VII. MEDIOS AUXILIARES

Aula virtual, pizarra acrílica, proyector, marcadores, computadora portátil, simuladores y herramientas de software, recursos de comunicación a través de internet.

VIII. BIBLIOGRAFÍA

- Patterson, D. A., & Hennessy, J. L. (2021). Computer Organization and Design: The Hardware/Software Interface (6th ed.). Morgan Kaufmann.
- Stallings, W. (2020). Computer Organization and Architecture: Designing for Performance (11th ed.). Pearson.
- Tanenbaum, A. S., & Austin, T. (2013). Structured Computer Organization (6th ed.). Pearson.
- Harris, D. M., & Harris, S. L. (2012). Digital Design and Computer Architecture (2nd ed.). Morgan Kaufmann.
- Floyd, M. (2006). Sistemas Digitales. Prentice Hall.
- Null, L., & Lobur, J. (2018). The Essentials of Computer Organization and Architecture (5th ed.). Jones & Bartlett Learning.
- Tocci, R. J., Widmer, N. S., & Moss, G. L. (2016). Sistemas Digitales: Principios y Aplicaciones (11.ª ed.). Pearson.



[Handwritten signature]

[Handwritten signature]