

PROGRAMA DE ASIGNATURA: ORGANIZACIÓN DEL COMPUTADOR

Facultad:	Ingeniería	Departamento:	Gestión de Proyectos y Sistemas
Código:	BPTSP03	Asignatura:	Organización Del Computador
Créditos:	3	Tipo:	<input checked="" type="checkbox"/> Obligatoria <input type="checkbox"/> Electiva
Carreras:	Ingeniería de Sistemas (IS)	Trimestres:	VI (IS)
Prerrequisito	Arquitectura del Computador (BPTEN12)	Modalidad:	Presencial
Número de horas semanales			
En aula	Prácticas supervisadas	Laboratorio	Aprendizaje Autónomo
4			4
Coordinador:	Christian Guillén Drija	Fecha de actualización	Sep. 2025

1. **Justificación:**

Comprender el funcionamiento interno de las computadoras es esencial para la innovación y eficiencia tecnológica actual. Esta asignatura ofrece una base sólida sobre el procesamiento, almacenamiento y gestión de información, capacitando al ingeniero para diseñar y optimizar sistemas. Al dominar la arquitectura y la programación de bajo nivel, los estudiantes desarrollan una actitud proactiva para potenciar el avance tecnológico y aprovechar al máximo el hardware.

2. **Propósito:**

Capacitar al estudiante en los principios fundamentales de la arquitectura y organización de sistemas computacionales contemporáneos. El curso busca que el alumno comprenda la interacción entre hardware y software, domine el lenguaje ensamblador y analice la jerarquía de memoria y el funcionamiento del procesador. Esto le permitirá evaluar el rendimiento de los sistemas y proponer mejoras técnicas alineadas con las demandas de la ingeniería moderna.

3. **Objetivos**

- Identificar y definir los componentes principales de un sistema de computación (hardware y software), así como sus funciones y relaciones. (Objetivos 1 y 4 de la carrera)
- Aplicar los sistemas de representación de datos (binario, octal, decimal, hexadecimal) para realizar conversiones y operaciones aritméticas/lógicas, resolviendo problemas específicos de manipulación de información (numérica y alfanumérica). (Objetivos 1 y 2 de la carrera)
- Desarrollar programas sencillos en lenguaje ensamblador, utilizando diferentes modos de direccionamiento y tipos de instrucciones para resolver problemas específicos. (Objetivos 1 y 2 de la carrera)

4. **Resultados de aprendizaje**

- RA10 - Diseño de soluciones de ingeniería de sistemas.** Capacidad para aplicar metodologías de diseño de soluciones informáticas, integrando conocimientos técnicos y habilidades de gestión para identificar, analizar y resolver problemas o nuevos requerimientos, considerando las necesidades del entorno. (N2)
- RA11 - Práctica de la ingeniería.** Capacidad para desarrollar, implementar, gestionar aplicaciones, productos informáticos y sistemas de información, usando metodologías de desarrollo, técnicas y herramientas computacionales apropiadas, asegurando calidad, seguridad y fiabilidad del producto final. (N2)

5. **Contenido**

Tema	Contenido	Herramientas técnicas y actividades (proyectos, trabajos, laboratorios)	Horas dedicadas
1	Introducción a la programación de sistemas: Elementos de Software en un Sistema Computador. Elementos de Hardware de un Sistema Computador. Origen y evolución del hardware y software. Generaciones de Computadoras.	Clase teórica con apoyo de presentaciones (Prezi/PPT) y videos explicativos. Análisis histórico del hardware.	2
2	Representación de datos: Sistemas Numéricos: Binario, octal, decimal, hexadecimal y Base N. Números Enteros; Representación de negativos mediante Complemento a 1 (restringido) y 2 (auténtico).	Ejercicios prácticos de conversión y aritmética binaria. Uso de calculadoras especializadas y hojas de cálculo para validación de precisión.	4

3	Arquitectura del computador: Conjuntos de Instrucciones. Clasificación. Clasificación de Máquinas de Registros. Direccionamiento de Memoria. Operaciones. Tipo y Tamaño de los operandos. Arquitecturas comerciales. IBM 360/370, INTEL 80x86. Caso de Estudio: Arquitectura MIPS.	Visualización de diagramas de arquitectura. Estudio de casos comerciales y uso de simuladores de hardware.	6
4	Representación de instrucciones: Formato de las Instrucciones. Sintaxis. Constantes, Literales, etc. Modos de Direccionamiento. Tipos de Instrucciones. Subrutinas y macros. Lenguaje de Maquina MIPS. Modelo de Programación MIPS	Clase teórica sobre sintaxis MIPS. Desarrollo de programas sencillos en ensamblador utilizando simuladores de arquitectura (MARS/SPIM).	8
5	Traductores, Interpretadores, Compiladores Ensambladores. Funciones básicas. Características del ensamblador. Ensamblador de la máquina MIPS.	Clase teórica: Explicación de conceptos para contextualización del tema. Talleres de programación en ensamblador.	2
6	Interrupciones: Definición. Tipos de Interrupciones. Clasificación. Mecanismos para el tratamiento de las interrupciones. Enmascaramiento y prioridades. Interrupciones en la máquina MIPS.	Estudio de casos reales en sistemas operativos. Debate grupal sobre la gestión de eventos y prioridades en el procesador.	4
7	Programación de entrada y salida: Periféricos. Unidad de control. Canales. Conexión de periférico-computador. Entrada/Salida en la máquina MIPS.	Clase teórica: Presentación de los conceptos de entrada y salida, los periféricos y los canales.	4
8	Dispositivos de Almacenamiento Externo. Concepto y Evolución de los dispositivos magnéticos. Organización de archivos: Secuenciales, Indexados, Aleatorios, Algoritmos de Hashing. Índices de Búsqueda	Clase teórica. Proyecto final y defensa: En equipos, los estudiantes deberán desarrollar un programa en MIPS capaz de realizar alguna tarea.	10
9	Métodos de compresión de archivos: Métodos de compresión estadística y compresión diccionario.	Clase teórica.	2

6. **Métodos de aprendizaje:**

Se emplea una metodología activa que combina la exposición teórica del docente con el aprendizaje basado en problemas y el diseño de proyectos prácticos. El curso fomenta el aprendizaje colaborativo mediante debates y resolución conjunta de casos, junto al aprendizaje autónomo apoyado en investigación y recursos digitales. Las sesiones incluyen prácticas supervisadas en laboratorio para el dominio del lenguaje ensamblador y la arquitectura de hardware, complementadas con tutorías personalizadas que aseguran el seguimiento continuo del progreso del estudiante y la consolidación de competencias técnicas.

7. **Métodos de evaluación:**

Aprendizaje en contacto con el docente (60%)	Aprendizaje práctico experimental (0%)	Aprendizaje autónomo (40%)
Exposiciones, Participación en clases, Debates, Exámenes escritos u orales, Talleres, Defensa de proyectos, entre otros.	Resolución de problemas prácticos, Prácticas de laboratorio, Salidas de campo o visitas técnicas, Manejo de software especializado, Prototipado técnico, Estudios de caso técnicos, entre otros.	Elaboración de informes, Resolución de problemas y ejercicios, Ensayos de investigación, Creación de mapas conceptuales, Participación en foros, entre otros.

8. **Referencias**

Obligatoria:

- Stallings, W. (2006). *Organización y arquitectura de computadores* (7.ª ed.). Pearson.
- Tanenbaum, A. S. (2000). *Organización de computadoras: Un enfoque estructurado*. Prentice Hall.
- Patterson, D. A., & Hennessy, J. L. (2021). *Computer organization and design: The hardware/software interface* (6th ed.). Morgan Kaufmann.
- Rocabado Moreno, S. H., & Arias Figueroa, D. (2020). *Arquitectura y organización de la computadora: Microprocesadores y programación assembler*. Universidad Nacional de La Plata.
- Patterson, D. A., & Hennessy, J. L. (1995). *Organización y diseño de computadores: La interfaz hardware/software*. McGraw-Hill.