

## FACULTAD/ESCUELA DE INGENIERÍA, DISEÑO Y CIENCIAS APLICADAS DEPARTAMENTO DE CIENCIAS FÍSICAS Y EXACTAS

### *Información de la asignatura*

<b>Nombre de la asignatura</b>	Arquitectura de computadores
<b>Código de la asignatura</b>	11343 - CFT
<b>Periodo Académico</b>	202320
<b>Nrc</b>	11654
<b>Grupo</b>	003
<b>Programas/Semestres</b>	SIS 05, TEL 05
<b>Intensidad horaria</b>	2
<b>Intensidad Semanal</b>	4
<b>Créditos</b>	3
<b>Docente(s)</b>	Duvan Fernando Garcia Cedeño

### *Introducción o presentación general del curso*

En el curso de Arquitectura de Computadores se estudian los principios fundamentales que sustentan los sistemas informáticos, mostrando de manera concreta cómo estas ideas afectan la funcionalidad y utilidad de los programas de aplicación. Además, se enseñarán técnicas de conducción de experimentos como herramientas para evaluar y analizar el rendimiento de los sistemas informáticos desde el punto de vista de el tiempo de ejecución y la eficiencia energética.

Al comprender el funcionamiento de los componentes de un sistema de cómputo, podrá reconocer sus capacidades y limitaciones en la ejecución de operaciones específicas y en función de una configuración determinada. También se estudiará la relación entre el hardware y el software y se explorarán los elementos esenciales de un sistema computacional, como la unidad central de procesamiento, la memoria, los dispositivos de entrada/salida y los buses de datos, entre otros.

### *Formación en competencias*

**SO-4.** Responsabilidad profesional: Reconocer sus responsabilidades éticas y profesionales como ingenieros, y emitir juicios informados al considerar el impacto de las soluciones de ingeniería en contextos globales, económicos, ambientales y sociales.

**SO-6.** Experimentación: Habilidad para desarrollar y conducir experimentos, así como analizar e interpretar datos para generar conclusiones a partir de juicios basados en principios de ingeniería.

**SO-7.** Aprendizaje permanente: Aprender de forma autónoma y aplicar nuevos conocimientos según sea necesario, utilizando estrategias de aprendizaje adecuadas.

### ***Objetivo general de aprendizaje***

- Brindar a los estudiantes las habilidades y conocimientos necesarios para evaluar la interacción entre los componentes de hardware y de software de un sistema de cómputacional con el fin emitir un juicio respecto a su comportamiento y desempeño.

#### **Objetivos terminales–resultados de aprendizajes de la asignatura**

1. **(OT1)** Comprender los conceptos básicos de la arquitectura de computadores y su relevancia en el contexto de los sistemas informáticos modernos.
2. **(OT2)** Analizar las capacidades y limitaciones que imponen los formatos de representación de la información en su procesamiento y almacenamiento en un sistema de cómputo.
3. **(OT3)** Establecer las relaciones entre el software y arquitectura hardware subyacente del sistema de cómputo en relación al impacto de los lenguajes de programación y el principio de localidad de los programas.
4. **(OT4)** Emitir un juicio sobre aspectos del comportamiento o desempeño de un sistema de cómputo utilizando técnicas de conducción de experimentos y teoría y conceptos de la arquitectura de computadores.

### ***Objetivos terminales - Resultados de aprendizajes***

<b>Objetivo terminal-Resultado de aprendizaje del curso o asignatura</b>	<b>Competencia en formación</b>	<b>Resultado de aprendizaje de la competencia de egreso al que se contribuye</b>
OT3	SO-7, SO-4	SO7-P1 (T) Aprendizaje permanente: Reconocer la importancia del aprendizaje permanente asumiendo la experiencia de aprendizaje como un proceso de autoaprendizaje. SO4-PI2 (T) Impacto de soluciones de ingeniería: Explicar el contexto global, ambiental, social y económico de las soluciones de ingeniería.

Objetivo terminal-Resultado de aprendizaje del curso o asignatura	Competencia en formación	Resultado de aprendizaje de la competencia de egreso al que se contribuye
OT4	SO-6, SO-4, SO-7	<p>SO6-PI1(A): Desarrollar protocolos de experimentación: seleccionando el tipo de protocolo de experimentación más adecuado, según una hipótesis dada, los recursos disponibles y los factores que deben medirse y controlarse. SO6-PI2 (A) Ejecutar un experimento: mediante el uso de herramientas adecuadas, siguiendo los procedimientos definidos y reportando adecuadamente el proceso y los resultados. SO6-PI3 (A) Analizar e interpretar datos: sacar conclusiones utilizando juicios de ingeniería. SO4-PI2 (T) Impacto de soluciones de ingeniería: Explicar el contexto global, ambiental, social y económico de las soluciones de ingeniería. SO7-PI2 (T) Analizar información relevante: Buscar/identificar, organizar y analizar información relevante en oportunidades de aprendizaje independiente.</p>
OT1	SO-7	<p>SO7-PI1 (T) Aprendizaje permanente: Reconocer la importancia del aprendizaje permanente asumiendo la experiencia de aprendizaje como un proceso de autoaprendizaje. SO 7-PI3 (T) Adquirir nuevos conocimientos: Demostrar la capacidad de adquirir nuevos conocimientos y aplicarlos para conseguir los resultados esperados.</p>
OT2	SO-4	<p>SO4-PI2 (T) Impacto de soluciones de ingeniería: Explicar el contexto global, ambiental, social y económico de las soluciones de ingeniería.</p>

## ***Unidades de aprendizaje***

### **UNIDAD 1: INTRODUCCIÓN A LA ARQUITECTURA DE UN SISTEMA DE CÓMPUTO**

Objetivos específicos:

- Identificar los niveles de abstracción de un computador convencional y su estructura básica.
- Dar razón de la evolución de la computación

Contenido:

- Definición y conceptos básicos
- Niveles de abstracción de un computador
- Evolución de los sistemas informáticos.
- Importancia de la arquitectura en el diseño de sistemas.

### **UNIDAD 2: ARITMÉTICA DEL SISTEMA DE CÓMPUTO**

Objetivos específicos:

- Manejar formatos de numeración en punto flotante y complemento a 2 mediante ejemplos concretos.
- Describir cómo opera una unidad aritmética y una Unidad de punto flotante.
- Identificar las limitaciones de los sistemas de numeración y reconocer los inconvenientes que puede ocasionar los cambios de tipo formato en los datos y análisis de errores por redondeo, truncamiento y desbordamiento.
- Reflexionar sobre los casos de catástrofes de software y tipo de errores de computación.

Contenido:

- Aritmética del computador: operaciones aritméticas con números enteros y reales en complemento a 2.
- El empleo de notación hexadecimal.
- Generalidades sobre cantidades y operaciones en punto flotantes: números, formatos y descripción de algunas aplicaciones y restricciones de las unidades de punto flotante en el manejo de la información.
- Las banderas como bits de información adicional a las operaciones realizadas por la ALU.
- Limitaciones y riesgos de la aritmética del computador

### **UNIDAD 3: ORGANIZACIÓN DE UNA UNIDAD CENTRAL DE PROCESAMIENTO**

Objetivos específicos:

- Detallar la organización interna básica de una CPU típica.
- Explicar el proceso en un esquema basado en buses (buses de datos, direcciones y de control) durante la búsqueda, decodificación y ejecución de instrucciones de los procesadores.
- Identificar la relación entre lenguaje máquina y lenguaje ensamblador

Contenido:

- Estructura interna aplicable a toda CPU: Unidad Aritmética Lógica (ALU), registros internos.
- Ejemplos con simulador de un procesador básico de 8 bits.
- Relación entre Instrucciones en lenguaje de máquina y ensamblador y lenguajes de alto nivel.
- Proceso detallado de las diferentes fases en el tratamiento de las instrucciones por un

procesador: 1) Búsqueda en memoria de las instrucciones (Fetch); 2) decodificación y 3) ejecución.

- Arquitecturas multiprocesador y multinúcleo

#### **UNIDAD 4: ARQUITECTURA DEL CONJUNTO DE INSTRUCCIONES (ISA)- LA INTERFAZ HARDWARE / SOFTWARE**

Objetivos específicos:

- Explicar los pasos requeridos para cargar y ejecutar un programa que es escrito en lenguaje de alto nivel en un sistema de cómputo.
- Estar en capacidad de traducir un programa corto de alto nivel (sentencias secuenciales básicas) a lenguaje ensamblador y dar razón de lo que sucede en cada una de las capas de abstracción del sistema de cómputo.
- Dar razón de las diferentes técnicas que se usan para mejorar la eficiencia de los procesadores.

Contenido:

- Compilación y ensamblaje de programas.
- Lenguaje de máquina y lenguaje ensamblador.
- Qué son los procesadores RISC y cómo operan. Ejemplos; semejanzas y diferencias con los CISC.
- Modos de direccionamiento, manejo de la pila; subrutinas; ejemplos.
- Conceptos básicos sobre el proceso de instrucciones sometidas a un “Pipeline” y Microprocesadores Superescalares
- Paralelismo a nivel de instrucción (ILP)

#### **UNIDAD 5: ORGANIZACIÓN DE MEMORIA Y EL PRINCIPIO DE LOCALIDAD**

Objetivos específicos:

- Dar razón de las jerarquías de memoria y poder diferenciar capacidades, rapidez y costos.
- Comprender el funcionamiento de la memoria caché y describir cómo afecta el rendimiento y la utilidad de los programas de aplicación.
- Explicar el proceso completo que se lleva a cabo en el manejo de las interrupciones por parte del procesador y demás elementos hardware.
- Comprender el uso y la importancia de las interrupciones y cambio de contexto en el procesamiento concurrente en los sistemas operativos.

Contenido:

- Principio de localidad.
- Organización jerárquica de la memoria, memoria principal y memoria secundaria. Memoria caché, concepto y descripción, métodos de implementación típicos, múltiples niveles.
- Ejemplos. Conceptos de memoria virtual y Paginación.
- Mecanismos generales de la comunicación con los periféricos y de su control por la CPU: Interrupciones de varias clases y procesos de atención a las mismas

#### **UNIDAD 6: DISEÑO Y CONDUCCIÓN DE EXPERIMENTOS SOBRE SISTEMAS DE CÓMPUTO**

Objetivos específicos:

- Identificar las etapas en el diseño de experimentos: objetivo, número de factores y efectos.
- Aplicar la metodología de diseño y conducción de experimentos partiendo de una hipótesis relacionada con el desempeño y comportamiento de un sistema de cómputo.

- Interpretar y analizar los datos obtenidos en un experimento así como de las pruebas de benchmarking para diferentes configuraciones de computadores y relacionar dichos resultados con los componentes hardware y software específicos.

Contenido:

- Planteamiento y declaración de hipótesis, ANOVA de una vía
- Importancia del diseño de los experimentos
- Definición de variables, factores, restricciones de un experimento.
- Interacciones y ANOVA de 2 vías
- Tipos de experimentos, factorial completo
- Tipos y etapas de experimentos adecuadas en sistemas de cómputo.
- Análisis e interpretación de datos arrojados de un experimento.

## **UNIDAD 7: TENDENCIAS ACTUALES EN LA ARQUITECTURA DE COMPUTADORAS\***

Objetivos específicos:

- Estudiar las innovaciones y avances más recientes en la arquitectura de computadoras, como la computación cuántica, la computación en la nube, la inteligencia artificial, virtualización entre otros.

Contenido:

- Arquitecturas de computación heterogénea (GPUs, FPGAs, etc.)
- Arquitecturas de bajo consumo energético
- Avances en computación cuántica

*\*La unidad 7 es opcional y su estudio dependerá del desarrollo del cronograma*

## ***Metodologías de aprendizajes***

El curso se desarrolla usando la metodología aprendizaje basado en problemas. El curso tiene varios momentos en donde en cada uno de ellos se le presenta al estudiante un problema real que debe resolver analíticamente usando los conceptos de la arquitectura de computadores.

Los estudiantes deberán preparar, antes de la clase, los temas que se asigne. Bajo el esquema de trabajo de este curso, preparar un tema significa estudiarlo siguiendo las recomendaciones dadas por el/la profesor(a) a través de las guías de estudio. Durante la clase, el estudiante deberá participar en las discusiones del tema, en la formulación de preguntas orientadas a resolver las dudas que hayan surgido al realizar las actividades de preparación del tema, y en la solución de ejercicios prácticos, de simulación, y estudio de casos. Después de cada clase, además de realizar los trabajos asignados por el/la profesor(a) para poner en práctica y afianzar conceptos, el estudiante deberá reflexionar sobre la relación del tema con temas anteriores de este y otros cursos y con su futuro desempeño profesional.

El laboratorio además de complementar y afianzar los conceptos mediante, éste será un insumo importante para desarrollar el trabajo final experimental.

Se definirán horarios de atención por fuera de clase para la solución de dudas, y canales de comunicación además del sistema Intu, para el acompañamiento oportuno durante el proceso de aprendizaje de los estudiantes.

El laboratorio además de complementar y afianzar los conceptos mediante, éste será un insumo importante para desarrollar el trabajo final experimental.

Se definirán horarios de atención por fuera de clase para la solución de dudas, y canales de comunicación además del sistema Intu, para el acompañamiento oportuno durante el proceso de aprendizaje de los estudiantes.

### *Evaluación de aprendizajes*

<b>Mecanismo o actividad evaluativa</b>	<b>Porcentaje de la nota final</b>	<b>Relación con objetivos terminales - resultado de aprendizaje del curso</b>	<b>Relación con el resultado de aprendizaje de la competencia de egreso</b>
VerTar	20	OT1, OT2, OT3 Verificaciones y Tareas	N/A
PracLabs	15	OT1,OT3,OT2, OT4 Prácticas de laboratorio	N/A
Parcial1	25	OT1, OT2 Examen parcial 1 (Arquitectura del computador, aritmética del computador)	N/A
Parcial2	25	OT3 Examen parcial 2 (Interface Hardware Software)	N/A
TrabajoFin	15	OT2, OT3, OT4 Trabajo Final (conducción de experimentos)	N/A

### *Recursos de apoyo*

#### LIBROS

Randal E. Bryant, David R. O'Hallaron (2015) . Computer Systems: A Programmer's Perspective, Edición 3er edición, Pearson, ISBN: 978-0134092669

Humberto Gutiérrez Pulido (2012) . Análisis y diseño de experimentos, Edición 3er edición, McGraw Hill, ISBN: 978-6071507259

#### TEXTOS COMPLEMENTARIOS

#### LIBROS

Jon M. Stokes (2006) . Inside the Machine: An Illustrated Introduction to Microprocessors and Computer Architecture. , Edición 1st Edición, No Starch Press, ISBN: 978-1593276683