



FACULTAD DE CIENCIAS EXACTAS, FÍSICAS y NATURALES



Universidad  
Nacional  
de Córdoba

Asignatura: **Sistemas de Computación**

Código: 10-09705

RTF

10

Semestre: Sexto

Carga Horaria

96

Bloque: Tecnologías Aplicadas

Horas de Práctica

16

Departamento: Computación

Correlativas:

- Electrónica Digital 3

Contenido Sintético:

- Arquitectura funcional de un sistema de computación.
- Componentes de hardware.
- Componentes de software.
- Interfaces internas y externas.
- Performance.
- Almacenamiento de datos.
- Sistemas de computación en red.

Competencias Genéricas:

- CG4: Utilizar de manera efectiva las técnicas y herramientas de aplicación en ingeniería.
- CG5: Contribuir a la generación de desarrollos tecnológicos y/o innovaciones tecnológicas.
- CG10: Actuar con espíritu emprendedor.

Aprobado por HCD: 993-HCD-2023

RES: Fecha: 12/11/2023

### Competencias Específicas:

CE1.1: Diseñar, proyectar y calcular sistemas, equipos y dispositivos de generación, transmisión y/o procesamiento de campos y señales analógicos y digitales; circuitos integrados; hardware de cómputo de propósito general y/o específico y el software a él asociado; hardware y software de sistemas embebidos y dispositivos lógicos programables; sistemas de automatización y control; sistemas de procesamiento y de comunicación de datos y sistemas irradiantes, para brindar soluciones óptimas de acuerdo a las condiciones técnicas, legales, económicas, humanas y ambientales.

CE1.4.3: Analizar, diseñar, programar, implementar, probar, depurar y evaluar hardware y software para sistemas de computación de propósitos específicos.

CE1.4.4: Analizar, diseñar, programar, implementar, probar, depurar y evaluar hardware y software para sistemas de computación de propósitos generales.

CE1.5.6: Analizar, diseñar, programar, implementar, probar, depurar y evaluar sistemas de comunicación de datos (hardware/software).

CE3.1: Validar y certificar el funcionamiento, condición de uso o estado de los sistemas mencionados anteriormente.

## Presentación

Sistemas de Computación es una asignatura que tiene como objeto de estudio las arquitecturas de computadoras de propósito general vinculadas a una red tipo internet. Pertenece al sexto semestre (tercer año) de la carrera Ingeniería Electrónica. Al momento de transitar este espacio curricular el estudiante ha adquirido competencias en electrónica digital y programación avanzada.

La programación del hardware de computadoras y sus arquitecturas han experimentado una gran expansión, debido a la confluencia de diversos factores entre los que se destacan:

- Lenguajes de programación que implementan diferentes paradigmas que suministran la base y el modelo para resolver problemas utilizando software.
- Arquitecturas de computadoras orientadas a resolver problemas tecnológicos de forma generalizada.
- Desarrollo de herramientas que facilitan la implementación de soluciones de software a problemas tecnológicos particulares sobre arquitecturas de computadoras de propósito general.

Considerando estos factores, en esta asignatura se desarrollan dos líneas conceptuales en las que los estudiantes desarrollen competencias:

### DISEÑO DE SOFTWARE

A partir de una especificación de requerimientos, ser capaz de diseñar e implementar un sistema de software con la problemática del hardware específico.

- Representar un diseño de software que contemple el hardware a utilizar.
- Reconocer las actividades más relevantes en el proceso de diseño e implementación de software.
- Utilizar distintos modelos de software y su representación.
- Identificar, evaluar y resolver los problemas que abarquen el desarrollo de software en el contexto de un hardware específico.

### SOFTWARE DE ALTO Y BAJO NIVEL

Ser capaz de diseñar y realizar sistemas de software de alto y bajo nivel en forma conjunta.

- Programar dispositivos de hardware con lenguajes de bajo y alto nivel.
- Utilizar la mecánica de comunicación entre lenguajes de programación de bajo y alto nivel.
- Realizar programas que interactúen directamente con diferentes sistemas operativos.
- Utilizar herramientas para lograr los objetivos arriba descritos.

El desarrollo de la asignatura se orienta a que el estudiante construya competencias en identificar, analizar, diseñar, construir y evaluar sistemas de software que exploten las ventajas del conocimiento del hardware subyacente; para lograr sistemas compuestos por hardware y software con eficiencia y eficacia en un contexto donde los requerimientos son volátiles y flexibles. Todo expresado como un diseño, su implementación y las pruebas de sistema.

# Contenidos

## **Unidad 1. Arquitectura de computadoras de propósito general**

Fundamentos de la arquitectura de computadoras. Arquitecturas clásicas y modernas. Evolución de la tecnología sobre la arquitectura de las computadoras. Taxonomía según su arquitectura. Mediciones de performance y rendimiento.

## **Unidad 2. Modelo de computadoras para programadores de aplicaciones y sistemas**

Programador de sistemas o programador de aplicaciones. Registros internos para el programador de aplicaciones. Segmentación en modo real. Segmentación en modo protegido. Juego de registros de una unidad en coma flotante. Registros del sistema. Registros de segmento. Registros de tablas. Registro de tareas. Registro de señalizadores. Registro de control. Registros de depuración. Registros de prueba de la TLB.

## **Unidad 3. Convenciones de llamadas**

Operandos de instrucciones de lenguajes de bajo nivel. Instrucciones básicas. Directivas de datos. Entradas y salidas. Creación de un primer programa. Dependencias del compilador. Ensamblado. Compilación con lenguajes de alto nivel. Depuración.

## **Unidad 4. Jerarquías de almacenamiento y buses**

Memoria caché. Acceso a una memoria caché. Gestión de caché misses. Gestión de escritura. Máquinas virtuales. Memoria virtual. Gestión de memoria virtual. Concepto de Translation-Lookaside Buffer (TLB). Tipos de almacenamiento de datos, formato de datos. Buses de datos más populares. Latencia y velocidad de acceso.

## **Unidad 5. Organización de memoria**

Organización de la memoria. La memoria en modo real. La memoria en modo protegido. El espacio lineal. Manejo de descriptores. Tablas de descriptores. El modelo plano. Mecanismos de paginación y segmentación. Formato de entradas al directorio y a las tablas de paginación. Tabla de traducción de direcciones lineales. Estructura y funcionamiento de la TLB.

## **Unidad 6. Mecanismos de protección**

Aspectos generales. Necesidad de protección. Tipos de protección. Niveles de protección. Protección entre tareas. Protección de segmentos. Protección de páginas. Protección de instrucciones.

## **Unidad 7. Compuertas de llamadas**

Transferencias de control. Definición y comportamiento de las Puertas de Llamada. Comportamiento de la pila en las transferencias internivel. El escenario del caballo de troya. Descriptores alias. Particularidades de los segmentos de pila.

## **Unidad 8. Aspectos de Sistemas operativos**

Procesos, hilos y tareas. Implantación de la multitarea. Conmutación de tareas. Puertas de tarea. Mapa de bits de permiso de E/S. Interrupciones y excepciones.

Entradas de la IDT. Código de error. Reglas de atención de una interrupción o una excepción. Sistemas operativos de tiempo real duro y blando.

### **Unidad 9. Redes de computadoras**

Paradigma cliente-servidor. Modelo TCP/IP. Capa de enlace. Capa de red. IPv4. Capa de transporte. Protocolos TCP y UDP. Capa de aplicación. Protocolo HTML. Hardware de red. Configuración de router y switch. Comunicaciones de datos.

### **Unidad 10. Modelado de sistemas de hardware y software**

Diseño de sistemas complejos con componentes de hardware y software. Definición de interfaces. Interfaces internas y externas. Lenguaje de modelado de sistemas. Performance. Pruebas de rendimiento. Verificación y validación de sistemas de hardware y software. Normativa.

## **Metodología**

El desarrollo general de la materia se sustenta en clases teórico-prácticas, mediante exposición dialogada, estudio de casos y resolución de problemas. Cada unidad se desarrolla a partir de un material bibliográfico obligatorio. A su vez se ofrecerán trabajos prácticos que orientan el proceso de lectura y análisis del contenido como forma de evaluación y acreditación de cada unidad. Los trabajos prácticos están guiados por los casos de estudio presentados en cada unidad y se orientan a resolver problemas concretos, aportando los contenidos teóricos necesarios para su resolución.

En ambientes de laboratorio el estudiante realiza casos de estudio, verifica su implementación con el cumplimiento de los requerimientos funcionales a nivel software y componente de hardware.

## **Evaluación**

La evaluación está inspirada en los criterios de evaluación continua. Se valora el desarrollo de las competencias durante todo el proceso de aprendizaje de la asignatura mediante una serie de pruebas de carácter sumativa y formativas distribuidas a lo largo del curso, que permiten al estudiante abordar la asignatura de forma progresiva. Se garantiza la retroalimentación temprana en el proceso de aprendizaje del alumno y permite a los profesores hacer un seguimiento global, con la posibilidad de actuar en caso de que lo aconsejen indicadores o situaciones determinadas. La evaluación de la parte relacionada con las prácticas se realiza al finalizar la actividad de laboratorio correspondiente.

Para la evaluación de la asignatura se utilizan las rúbricas basadas en los siguientes criterios, relacionados con los resultados del aprendizaje:

- El estudiante entiende los componentes básicos presentes en un sistema de computación (RA1).
- El estudiante puede modelar un sistema de computación compuesto por hardware y software (RA2).
- El estudiante, al modelar, reconoce actividades críticas del proceso de diseño e implementación de un sistema de computación (RA3).
- El estudiante identifica los principales problemas abordados en el desarrollo de un sistema de computación (RA4).

- El estudiante puede evaluar los problemas que abarquen el desarrollo de un sistema de computación (RA5).
- El estudiante puede resolver los problemas que abarquen el desarrollo de un sistema de computación (RA6).
- El estudiante puede programar dispositivos de hardware de propósito general con lenguajes de bajo y alto nivel (RA7).
- El estudiante comprende la mecánica de comunicación entre lenguajes de programación de bajo y alto nivel (RA8).
- El estudiante puede construir software que interactúe de forma directa con sistemas operativos sobre hardware de propósitos generales. (RA9)
- El estudiante puede confirmar la exactitud en el cumplimiento de los requerimientos funcionales implementados en un sistema de computación (RA10).
- El estudiante puede exponer oralmente los principios técnicos y funcionales intervinientes en los sistemas de computación construidos (RA11).
- El estudiante puede producir e interpretar textos técnicos (memorias, informes, etc.) y presentaciones públicas (RA12).
- El estudiante se desempeña en distintos roles, según lo requiera la tarea, la etapa del proceso y la conformación del equipo (RA13).
- El estudiante puede validar y verificar un sistema contrastándolo con las especificaciones y normativas aplicables (RA14).
- El estudiante muestra una actitud emprendedora durante el desarrollo de la asignatura (propone ejemplos, busca bibliografía, etc.) (RA15).

### **Instrumentos de evaluación**

El rendimiento de los alumnos es valorado mediante la toma de tres instancias de evaluación y cinco actividades prácticas de laboratorio, todo con carácter sumativo y formativo.

En el caso de las instancias de evaluación, se busca evidenciar las competencias desarrolladas a través de presentación de casos prácticos (análisis, diseño, resolución de problemas, preguntas conceptuales, etc.). Constituyen en sí mismos una instancia de evaluación formativa ya que luego los estudiantes reciben una realimentación de los errores cometidos, además de resolver los temas del parcial en la clase siguiente.

En el caso de las Actividad Práctica de Laboratorio grupal (APL, ver más adelante), demanda la presentación de prototipos funcionales e informes, donde los estudiantes explican individualmente las diferentes partes que componen el prototipo (criterios de diseño, fuentes consultadas, etc.), y particularmente sus detalles constructivos. Es el momento donde el estudiante pone en juego su participación en el equipo, el rol y peso de sus decisiones, su capacidad de comunicar detalles de diseño, y su correcto manejo y comprensión de las decisiones tecnológicas presentes en el prototipo. Para la entrega que demande cada prototipo el equipo de cátedra genera un repositorio virtual con control de versiones.

### **Condiciones de aprobación**

Condiciones de regularización

- Asistir al 80% de las clases.

- Aprobar todas las actividades prácticas de laboratorio con al menos el 60% o más de los criterios de evaluación expresados en la sección anterior.
- Alcanzar una puntuación mínima de 60% en los resultados de aprendizaje.
- Aprobar al menos dos instancias de evaluación, incluida instancia de recuperación, con el 60% como mínimo.

Condiciones de aprobación por promoción (no requiere examen final)

- Cumplir con todas las condiciones de regularización.
- Aprobar cada una de las instancias de evaluación con un mínimo de 60%.

Condiciones de aprobación por examen final

- Todas las condiciones de regularización expuestas anteriormente.
- Aprobación de un examen final con un mínimo de 60%.
- 

Para la nota final se promedian las notas obtenidas en cada una de las actividades prácticas de laboratorio, y el resultado se promedia con las notas de las instancias de evaluación.

### **Actividades prácticas de laboratorio**

Se realizan 5 (cinco) actividades donde se llevan a la práctica los conocimientos estudiados en la asignatura y se desarrollan las competencias esperadas. Estos trabajos se realizan en grupos, lo que permite que se desarrollen competencias de trabajo en equipo y coordinación de tareas.

**APL1** – Convención de llamadas de software, benchmarking: En este trabajo práctico de laboratorio los estudiantes diseñan e implementan las primeras aplicaciones donde ponen aplican conocimientos sobre ensamblador y depuración.

**APL2** – Sistemas operativos de tiempo real y performance: En este trabajo los estudiantes aplican los conceptos básicos de conmutación de tareas y organización de memoria. Continúan mejorando sus herramientas de depuración y profiling (benchmark) para medir tiempos de latencia de una aplicación simple en un microcontrolador y lo comparan con un microprocesador con un sistema operativo de propósito general.

**APL3** – Programación de drivers en Linux: Todo componente electrónico que se conecte a una computadora necesita de un driver para interactuar directamente con el sistema operativo. En este trabajo práctico se desarrolla un driver sencillo para un sistema operativo de propósito general, además se observará cómo interactúa un proceso con el kernel del sistema operativo y alguno de los buses de datos.

**APL4** – API Rest y su integración con lenguajes de bajo nivel en hardware virtualizado: En este trabajo los estudiantes desarrollan actividades que les permitirán aplicar los conocimientos de redes e integrar el resto de los conceptos aprendidos en los trabajos anteriores. Desarrollan un servidor simple en un microcontrolador virtualizado y otro en un microprocesador de propósitos generales virtualizado. Para ello deberán utilizar un enrutador y diseñar una red acorde al problema planteado

**APL5** – Implementación de un sistema de computación complejo: En este trabajo práctico de laboratorio los estudiantes diseñan e implementan un sistema de computación complejo que incluye la adquisición, almacenamiento, transmisión y visualización de datos distribuidos. Para ejercitar e integrar los conocimientos adquiridos, se realizan ejercicios con diferentes herramientas para cada una de las funcionalidades, se acompaña y guía a los estudiantes en el proceso de diseño e implementación. Finalmente deben presentar una prueba de concepto funcional junto a la documentación del proyecto que incluye diseño y validación.

## Resultados de aprendizaje

Los resultados de aprendizaje a promover en el desarrollo de la asignatura son catorce, en relación con el descriptor “Fundamentos de Programación de Sistemas Informáticos” y “Sistemas de Comunicación”, ambos dentro del bloque de “Tecnologías Aplicadas”.

En la siguiente enumeración de resultados de aprendizaje, llamamos “sistemas de computación” a los artefactos compuestos por software y hardware de computadoras de propósitos generales, destinados a resolver problemas tecnológicos de propósito general y específico.

RA1. Entender los componentes básicos presentes en un sistema de computación.

RA2. Modelar un sistema de computación.

RA3. Reconocer actividades críticas del proceso de diseño e implementación de un sistema de computación.

RA4. Identificar los problemas que abarquen el desarrollo de un sistema de computación.

RA5. Evaluar los problemas que abarquen el desarrollo de un sistema de computación.

RA6. Resolver los problemas que abarquen el desarrollo de un sistema de computación.

RA7. Programar dispositivos de hardware de propósito general con lenguajes de bajo y alto nivel.

RA8. Comprender la mecánica de comunicación entre lenguajes de programación de bajo y alto nivel.

RA9. Construir software que interactúe de forma directa con sistemas operativos sobre hardware de propósitos generales.

RA10. Confirmar la exactitud en el cumplimiento de los requerimientos funcionales implementados en un sistema de computación.

RA11: Exponer oralmente los principios técnicos y funcionales intervinientes en los sistemas de computación construidos.

RA12: Producir e interpretar textos técnicos (memorias, informes, etc.) y presentaciones públicas.

RA13: Desempeñar distintos roles, según lo requiera la tarea, la etapa del proceso y la conformación del equipo.

RA14: Validar y verificar un sistema contrastandolo con las especificaciones y normativas aplicables.

RA15: Mantener actitud emprendedora frente a los distintos desafíos que se le presentan.

En la Tabla 1 se muestran las competencias específicas y los resultados de aprendizaje relacionados.

| <b>Desagregación de Competencias Específicas</b>   | <b>Resultados de Aprendizaje</b>    |
|--|-------------------------------------|
| CE1.1: Diseñar, proyectar y calcular sistemas, equipos y dispositivos de generación, transmisión y/o procesamiento de campos y señales analógicos y digitales; circuitos integrados; hardware de cómputo de propósito general y/o específico y el software a él asociado; hardware y software de sistemas embebidos y dispositivos lógicos programables; sistemas de automatización y control; sistemas de procesamiento y de comunicación de datos y sistemas irradiantes, para brindar soluciones óptimas de acuerdo a las condiciones técnicas, legales, económicas, humanas y ambientales. | RA1, RA2, RA3, RA12.                |
| CE1.4.3_A: Analizar y diseñar hardware y software para sistemas de computación de propósitos específicos.  | RA1, RA2, RA3, RA14.                |
| CE1.4.3_B: Programar, implementar, probar y depurar hardware y software para sistemas de computación de propósitos específicos.  | RA4, RA5, RA6, RA7, RA8, RA9, RA13. |
| CE1.4.3_C: Evaluar hardware y software para sistemas de computación de propósitos específicos.   | RA10, RA11.                         |
| CE1.4.4_A: Analizar y diseñar hardware y software para sistemas de computación de propósitos generales.  | RA1, RA2, RA3, RA12.                |
| CE1.4.4_B: Programar, implementar, probar y depurar hardware y software para sistemas de computación de propósitos generales.  | RA4, RA5, RA6, RA7, RA8, RA9, RA13. |
| CE1.4.4_C: Evaluar hardware y software para sistemas de computación de propósitos generales.   | RA10, RA11.                         |
| CE1.5.6: Analizar, diseñar, programar, implementar, probar, depurar y evaluar sistemas de comunicación de datos (hardware/software).   | RA9, RA12                           |

|   |                     |
|---|---------------------|
| CE3.1: Validar y certificar el funcionamiento, condición de uso o estado de los sistemas mencionados anteriormente. | RA14                |
| CG4: Utilizar de manera efectiva las técnicas y herramientas de aplicación en ingeniería.                           | RA2, RA7, RA9, RA14 |
| CG5: Contribuir a la generación de desarrollos tecnológicos y/o innovaciones tecnológicas.                          | RA4, RA5, RA6       |
| CG10: Actuar con espíritu emprendedor.  | RA4, RA5, RA6, RA15 |

## Bibliografía

- Patterson (2022). Patterson D., Hennessy J.; Computer Organization and Design. The Hardware Software Interface. Second Edition. ISBN: 978-0-12-820331-6 Elsevier. 2022.
- Arikpo (2007) Arikpo I., Ogban F. and Eteng I.; Von Neumann Architecture and Modern Computers. Global Journal of Mathematical Sciences Vol.6, No.2, 2007: 97-103. Copyright(C) Bachudo Science Co.Ltd. Printed in Nigeria. ISSN 1596-6208.
- Angulo (2003) Angulo Usategui J., Gutierrez Temino J., Angulo Martínez I.; Arquitectura de microprocesadores. Los pentium a fondo. ISBN 10: 8497322428 / ISBN 13: 9788497322423. Published by Ediciones Paraninfo, S.A, 2003.
- Kurose (2017). Kurose J., Ross K. Redes de Computadoras un enfoque descendente. Séptima edición. ISBN: 978-84-9035-528-2. Pearson, S.A., 2017.