

Asignatura: **Informática y Cálculo Numérico**

Código: 10-04061

RTF

7

Semestre:

- Segundo (Ingeniería Electrónica)
- Cuarto (Ingeniería Biomédica)

Carga Horaria

96

Bloque: Ciencias Básicas de la Ingeniería

Horas de Práctica

48

Departamento: Computación

Correlativas:

- Análisis Matemático 1

Contenido Sintético:

- Algoritmos y Lenguajes de Programación.
- Entrada y Salida de la Información.
- Distintos Tipos de Datos.
- Operaciones aritméticas, relacionales y lógicas.
- Estructuras de control: condicionales e iterativas.
- Funciones de bibliotecas y definidas por el usuario.
- Verificación y validación de programas.
- Cálculo Numérico.

Competencias Genéricas:

- CG1: Identificar, formular y resolver problemas de ingeniería.
- CG4: Utilizar de manera efectiva las técnicas y herramientas de la ingeniería.
- CG9: Aprender en forma continua y autónoma.

Aprobado por HCD: 1054-HCD-2023

RES: Fecha: 29/11/2023

Competencias Específicas para la carrera de **Ingeniería Electrónica:**

- CE1.1: Diseñar, proyectar y calcular 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; para brindar soluciones óptimas de acuerdo a las condiciones técnicas, legales, económicas, humanas y ambientales.
- CE1.2.1: Modelar matemáticamente problemas de ingeniería, hallar soluciones específicas empleando algoritmos matemáticos y herramientas informáticas, y generalizar las soluciones para resolver situaciones reales de ingeniería.
- CE1.2.2: Manejar herramientas informáticas que permitan el desarrollo de soluciones en el ámbito de la ingeniería.
- CE1.4.5: Analizar, diseñar, implementar y probar sistemas embebidos y su software asociado.

Competencias Específicas para la carrera de **Ingeniería Biomédica:**

- CE8.A4: Plantear modelos matemáticos y comprender los principios para generalizar las soluciones específicas de los problemas de ingeniería biomédica mediante herramientas informáticas basadas en algoritmos matemáticos.
- CE8.A5: Manejar herramientas informáticas que permitan el desarrollo de soluciones en el ámbito de la ingeniería biomédica.

## Presentación

La asignatura Informática y Cálculo Numérico se dicta en el segundo semestre (primer año) de Ingeniería Electrónica y en el cuarto semestre (segundo año) de Ingeniería Biomédica. Al momento de transitar este espacio curricular, el estudiante ha adquirido conocimientos básicos del análisis matemático que le permitirán resolver un conjunto de problemas de ingeniería. En este contexto, la informática y el cálculo numérico se erigen como herramientas fundamentales, no solo por su capacidad para modelar y simular escenarios, sino también por su potencial para innovar y adaptar soluciones a desafíos emergentes. Desde el campo disciplinar la informática aporta la habilidad de estructurar, procesar y analizar grandes volúmenes de datos, así como la capacidad de diseñar algoritmos eficientes y sistemas robustos. Por otro lado, el cálculo numérico, brinda las técnicas y métodos para aproximar soluciones a problemas matemáticos que no tienen una solución exacta, siendo esencial para el modelado y simulación en ingeniería. De aquí la gran relevancia de esta asignatura en el contexto de ambas carreras.

La Ingeniería Electrónica, con su enfoque en sistemas electrónicos, requiere una comprensión profunda de algoritmos y estructuras de datos para optimizar soluciones. Por su parte, la Ingeniería Biomédica, al fusionar principios de medicina y biología, demanda herramientas numéricas precisas para modelar y analizar sistemas biológicos y dispositivos médicos. En ambos casos, la capacidad de programar, simular y analizar numéricamente resulta esencial.

Las competencias desarrolladas en la asignatura trascienden la mera programación. Los estudiantes aprenden a pensar lógicamente, a abordar problemas desde múltiples perspectivas y a diseñar soluciones eficientes. Estas habilidades son cruciales para cualquier ingeniero, ya que la capacidad de analizar problemas, diseñar algoritmos y aplicar soluciones técnicas son esenciales en el mundo profesional. Además, la asignatura fomenta el pensamiento crítico, la resolución de problemas y la adaptabilidad, competencias que son vitales en cualquier campo de la ingeniería.

Por último, es importante destacar que la asignatura es un cimiento para materias más avanzadas en los planes de estudio de las carreras de ingeniería. Las competencias adquiridas en esta asignatura facilitan la comprensión y el abordaje de temas más complejos, permitiendo a los estudiantes enfrentar con confianza y habilidad desafíos más avanzados en su trayecto de formación.

## Contenidos

### **Unidad 1: Fundamentos de la programación**

Resolución de problemas y algoritmos. Lenguajes de programación. Concepto de programa y sus elementos. Tipos de datos primitivos. Operaciones aritméticas. Variables, constantes y declaraciones. Operaciones de asignación. Entrada y salida estándar de información. Formato de salida. Funciones de biblioteca. Funciones definidas por el usuario: procedimientos y funciones con parámetros por valor. Alcance de variables. Aplicaciones. Verificación y validación de programas con flujo secuencial.

## **Unidad 2: Estructuras de selección**

Operaciones relacionales. Operaciones lógicas. Precedencia y asociatividad. Estructuras de selección. La estructura de decisión simple. La estructura de decisión doble. Estructuras de decisión anidadas. La estructura de decisión múltiple. Funciones definidas por el usuario: funciones con parámetros por referencia. Aplicaciones. Verificación y validación de programas con flujo selectivo.

## **Unidad 3: Estructuras de repetición**

Estructuras de repetición. Las estructuras de repetición indefinidas. La estructura de repetición definida. Estructuras de repetición anidadas. Alteraciones del flujo normal. Funciones definidas por el usuario: recursividad. Aplicaciones. Verificación y validación de programas con flujo repetitivo.

## **Unidad 4: Estructuras de datos compuestos**

Arreglos unidimensionales. Inicialización de arreglos. Arreglos bidimensionales. Algoritmos de búsqueda y ordenamiento. Funciones definidas por el usuario: arreglos como argumentos. Aplicaciones. Verificación y validación de programas con arreglos.

## **Unidad 5: Cálculo numérico**

Modelos matemáticos. Conceptos y aplicaciones del cálculo numérico en ingeniería. Métodos y algoritmos para la búsqueda de raíces. Resolución de ecuaciones por métodos numéricos. Interpolación y Extrapolación. Diferenciación numérica. Integración numérica. Reglas y algoritmos para la integración numérica. Funciones definidas por el usuario. Aplicaciones. Verificación y validación de programas de cálculo numérico.

## **Metodología de enseñanza**

La asignatura se organiza en 5 unidades didácticas, las cuales se desarrollan en un rango de entre 4 y 6 clases cada una. La propuesta del curso se lleva a cabo a través de 2 clases semanales de carácter teórico-práctico, con una duración de 3 horas cada una. La metodología de enseñanza propuesta integra las estrategias didácticas del aula invertida, el aprendizaje basado en problemas (ABP), el estudio de casos y el uso del aula virtual como complemento esencial a las clases semanales.

Los estudiantes trabajan con el material de estudio disponible en el aula virtual, que incluye videos, lecturas y casos prácticos, cubriendo conceptos teóricos y ejemplos básicos de programación. Al inicio de cada sesión en clase, se lleva a cabo una revisión y profundización de este material, asegurando que todos los estudiantes hayan comprendido los conceptos claves y estén listos para aplicarlos.

Durante las clases, el docente propone y guía la actividad de estudio de casos, que se incorpora como una herramienta para analizar situaciones reales o simuladas en el mundo de la programación. Esta actividad permite a los estudiantes aplicar y contextualizar los conceptos aprendidos. Estos casos, junto con otros problemas y programas afines propuestos por el docente, sirven como punto de partida para una serie de preguntas conceptuales que fomentan la discusión y el análisis de las posibles respuestas entre los estudiantes. Esta dinámica tiene como objetivo profundizar en conceptos fundamentales de

la programación, permitiendo a los estudiantes no solo entender las soluciones correctas, sino también identificar y discutir errores comunes.

En las clases presenciales, se dedica tiempo a resolver dudas, realizar ejercicios prácticos y trabajar en problemas específicos. Estos problemas, presentados al inicio de cada unidad, requieren soluciones a través de la programación. Los estudiantes trabajan en grupos para discutir y buscar soluciones, con el docente actuando como guía y facilitador. Al final de cada unidad, los grupos presentan sus soluciones, propiciando un espacio de retroalimentación y discusión colectiva, enriqueciendo así el proceso de aprendizaje a través de la experiencia compartida.

Además de las clases semanales, y como parte de la fase posterior de la metodología del aula invertida, se complementa la discusión de problemas y ejercicios y la atención de consultas a través de un foro disponible en el aula virtual. Este espacio permite la comunicación asincrónica pero continua entre docentes y estudiantes, fomentando un aprendizaje colaborativo y constante, y permitiendo a los estudiantes profundizar y reflexionar sobre lo aprendido en clase.

## Evaluación

La evaluación se estructura en base a un enfoque continuo e integrador, reflejando la naturaleza progresiva y acumulativa del aprendizaje en las áreas de informática y cálculo numérico. Esta metodología de evaluación tiene como objetivo no solo medir el conocimiento adquirido, sino también fomentar una comprensión profunda y aplicada de los conceptos y habilidades aprendidos. En este contexto, la evaluación se realiza a través de una evaluación parcial de cada una de las 5 unidades didácticas a través de las cuales se organiza la asignatura, y una evaluación integradora una vez aprobadas las 5 evaluaciones parciales.

- **Evaluación Parcial:** cada una de las cinco unidades didácticas culmina con una instancia de evaluación la cual es precedida por una actividad práctica.
  - ◆ **Trabajo Práctico (TP):** Los estudiantes deberán realizar un trabajo práctico que refleje la aplicación de los conceptos y habilidades aprendidos en la unidad. Este trabajo tiene como objetivo desarrollar la capacidad del estudiante para aplicar de manera práctica y efectiva lo aprendido, y puede incluir ejercicios, problemas, proyectos o simulaciones, según lo que sea más pertinente para la unidad en cuestión. Se dispone de al menos 2 semanas para poder realizar esta actividad, debiendo la misma ofrecer realimentación inmediata al estudiante de forma que pueda realizarla de forma asincrónica sin la supervisión del docente dentro del aula virtual.
  - ◆ **Evaluación Parcial (EP):** Una vez completado el TP con al menos el 60%, los estudiantes se realizan una evaluación parcial para medir su comprensión teórica y práctica de los contenidos de la unidad. Esta evaluación puede incluir preguntas teóricas, problemas prácticos y/o análisis de casos, asegurando una evaluación completa de los conocimientos y habilidades adquiridos en la unidad. La duración de esta actividad no debe superar los 30 minutos y la misma debe ser realizada bajo la supervisión del docente durante el horario de clase dentro del aula virtual. Se prevé una instancia de

recuperación de esta evaluación al finalizar cada unidad didáctica, y una adicional antes de la evaluación integradora (totalizando 6 instancias de recuperación). En cada instancia será posible recuperar una evaluación parcial de cualquier unidad didáctica ya finalizada.

→ **Evaluación Integradora:** una vez aprobadas las 5 evaluaciones parciales, se propone una instancia de evaluación integradora, de la cual consta de dos partes:

- ◆ *Práctica Integradora (PI):* Al finalizar cada unidad didáctica, se propone al estudiante una actividad integradora, la cual integra todas las unidades ya trabajadas en el curso y consiste en el desarrollo de un programa informático cuya especificación es dada a los estudiantes como enunciado. Esta actividad es temporizada a los fines de poner en práctica la gestión de los tiempos y prioridades en el desarrollo de un programa. Es posible realizarla de forma asincrónica a través del aula virtual, sin necesidad de supervisión docente, y la misma deberá ofrecer realimentación inmediata al estudiante.
- ◆ *Evaluación Integradora (EI):* Una vez aprobadas todas las Evaluaciones Parciales, los estudiantes deben presentarse a una evaluación integradora de todas las unidades didácticas. Esta evaluación tiene como objetivo medir la capacidad del estudiante para conectar y aplicar de manera integrada los conceptos y habilidades aprendidos a lo largo del curso. Esta evaluación es temporizada y tiene una duración de 2 horas, debiendo ser realizada en una clase bajo la supervisión de un docente dentro del aula virtual. Se prevé una instancia de recuperación de esta evaluación.

La combinación de evaluaciones parciales y una evaluación integradora garantiza que los estudiantes no solo adquieran y retengan el conocimiento, sino que también desarrollen habilidades críticas y aplicadas esenciales para su futuro profesional en el campo de la ingeniería.

## Condiciones de aprobación

### Condiciones de regularización

Para alcanzar la condición de regular, el estudiante debe cumplir las siguientes condiciones:

1. Asistir al menos al 80% de las clases. La asistencia es registrada en el aula virtual mediante la realización de los TPs y las EPs.
2. Alcanzar un rendimiento no inferior a 60% en cada uno de los Trabajos Prácticos (TP).
3. Alcanzar un rendimiento global no inferior a 60% en cada uno de las Evaluaciones Parciales (EP).

### Condiciones de promoción

Para alcanzar la promoción, el alumno debe cumplir las siguientes condiciones:

1. Alcanzar la condición de alumno regular para lo cual se deben cumplir las condiciones indicadas al respecto.
2. Alcanzar un rendimiento no inferior al 60% en la Evaluación Integradora (EI).

Cumplidas estas condiciones, la nota final de promoción se calculará según la siguiente fórmula:

$$\text{Nota Final} = \text{redondear}( (20\% \cdot \text{TPs} + 20\% \cdot \text{EPs} + 60\% \cdot \text{EI}) / 10 )$$

Donde TPs y EPs corresponden al resultados de promediar los TP y EP de cada unidad respectivamente.

## Examen final

### Estudiantes Regulares

Los estudiantes regulares deben rendir un examen equivalente a la actividad Evaluación Integradora (EI), el cual será calificado del mismo modo que para los estudiantes que alcanzaron la promoción, considerando para las variables TPs y EPs el rendimiento alcanzado durante la cursada, y para la variable EI el rendimiento alcanzado en el examen final.

### Estudiantes Libres

Los estudiantes libres deben rendir un examen que consta de dos partes:

- Una prueba de competencias con la misma metodología y objetivos que la Evaluación Parcial (EP), acotada a ejercicios que serán evaluados automáticamente en el aula virtual. La aprobación de esta primera parte es requisito excluyente para la prosecución del examen, y se deberá obtener un rendimiento no inferior al 60%.
- Un examen de Evaluación Integradora (EI), con la misma modalidad y objetivos que el requerido a los estudiantes regulares.

La Nota Final final para los estudiantes libres se obtendrá por la siguiente expresión:

$$\text{Nota Final} = \text{redondear}( (40\% \text{ EP} + 60\% \text{ EI}) / 10 )$$

## Actividades prácticas y de laboratorio

Las actividades prácticas se desarrollan a partir de trabajos prácticos disponibles en el aula virtual. Cada una de las 5 unidades sobre la que se organiza el recorrido formativo de la asignatura comprende un Trabajo Práctico (TP) compuesto por ejercicios que proponen el desarrollo de programas como solución a diferentes problemas generales. Cada problema es descrito a través de un enunciado y al menos una solución particular es ejemplificada indicando el resultado esperado. Para cada problema enunciado se espera que el estudiante provea un código en lenguaje de programación que será evaluado de forma inmediata a través de diferentes pruebas predefinidas. Estas pruebas comprenden soluciones a diferentes casos que el programa debe resolver de forma satisfactoria, y ofrece al estudiante una realimentación inmediata sobre las pruebas superadas y las fallidas. De esta forma, el estudiante puede realizar un diagnóstico sobre los casos donde falla y buscar así modificar su solución para que la misma sea de carácter general, responda a la consigna, y pueda entonces superar todas las pruebas propuestas.

## Resultados de aprendizaje

Los resultados de aprendizaje a promover en el desarrollo de la asignatura son veintisiete, en relación con los descriptores de las carreras de:

- Ingeniería Electrónica: Fundamentos de Programación de Sistemas Informáticos (Ciencias Básicas), Cálculo y Análisis Numérico (Ciencias Básicas).
- Ingeniería Biomédica: Fundamentos de Programación de Sistemas Informáticos (Ciencias Básicas), Informática y Cálculo Numérico (Tecnologías Básicas).

los cuales representan una amplia gama de habilidades y conocimientos relacionados con la programación y su aplicación en la resolución de problemas, incluyendo aquellos relevantes para la ingeniería.

- RA01: Comprender y aplicar algoritmos y resolución de problemas en la programación.
- RA02: Identificar y utilizar lenguajes de programación.
- RA03: Definir y gestionar elementos de un programa, como variables y constantes.
- RA04: Realizar operaciones aritméticas y manipular tipos de datos.
- RA05: Gestionar entrada/salida de información y formato de salida.
- RA06: Utilizar funciones de biblioteca y crear funciones definidas por el usuario.
- RA07: Comprender el alcance de las variables y su aplicación.
- RA08: Verificar y validar programas con flujo secuencial.
- RA09: Aplicar operaciones relacionales y lógicas en el control de flujo.
- RA10: Implementar estructuras de selección (simple, doble, múltiple) y anidadas.
- RA11: Crear funciones definidas por el usuario con parámetros por referencia.
- RA12: Aplicar estructuras de selección en diferentes contextos.
- RA13: Comprender y aplicar estructuras de repetición (indefinidas y definidas).
- RA14: Utilizar estructuras de repetición anidadas y gestionar alteraciones del flujo normal.
- RA15: Implementar la recursividad en funciones definidas por el usuario.
- RA16: Aplicar estructuras de repetición en la solución de problemas.
- RA17: Trabajar con arreglos unidimensionales y bidimensionales.
- RA18: Inicializar y manipular arreglos.
- RA19: Aplicar algoritmos de búsqueda y ordenamiento en arreglos.
- RA20: Utilizar funciones definidas por el usuario con arreglos como argumentos.
- RA21: Aplicar arreglos en la solución de problemas específicos.
- RA22: Comprender conceptos y aplicaciones del cálculo numérico en ingeniería.
- RA23: Definir y gestionar errores en cálculos numéricos.
- RA24: Implementar métodos para la búsqueda de raíces y algoritmos de integración numérica.
- RA25: Crear funciones definidas por el usuario en cálculos numéricos.
- RA26: Aplicar el conocimiento de cálculo numérico en la solución de problemas específicos.
- RA27: Verificar y validar programas de cálculo numérico.

A continuación, se indican las Competencias Genéricas, Competencias Específicas y los resultados de aprendizaje relacionados, para Ingeniería Electrónica e Ingeniería Biomédica, que serán evaluados con las rúbricas correspondientes.



<b>Competencias Genéricas</b>	<b>Resultados de Aprendizaje</b>
CG1: Competencia para identificar, formular y resolver problemas de ingeniería.	RA01, RA03, RA09, RA10, RA13, RA16, RA17, RA19
CG4: Competencia para utilizar de manera efectiva las técnicas y herramientas de la ingeniería.	RA02, RA04, RA05, RA06, RA07, RA08, RA20, RA22, RA23, RA24, RA25, RA26, RA27
CG9: Competencia para aprender en forma continua y autónoma.	RA01, RA02, RA08, RA16, RA22, RA26

<b>Competencias Específicas (Ingeniería Electrónica)</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.	RA01, RA03, RA09, RA10, RA13, RA16, RA17, RA19
CE1.2.1: Modelar matemáticamente problemas de ingeniería, hallar soluciones específicas empleando algoritmos matemáticos y herramientas informáticas, y generalizar las soluciones para resolver situaciones reales de ingeniería.	RA04, RA08, RA19, RA22, RA23, RA24, RA25, RA26, RA27
CE1.2.2: Manejar herramientas informáticas que permitan el desarrollo de soluciones en el ámbito de la ingeniería.	RA02, RA06, RA08, RA20, RA27
CE1.4.5: Analizar, diseñar, implementar y probar sistemas embebidos y su software asociado.	RA01, RA02, RA03, RA04, RA05, RA06, RA08, RA10, RA12, RA17

<b>Competencias Específicas (Ingeniería Biomédica)</b>	<b>Resultados de Aprendizaje</b>
CE8.A4: Plantear modelos matemáticos y comprender los principios para generalizar las soluciones específicas de los problemas de ingeniería biomédica mediante herramientas informáticas basadas en algoritmos matemáticos.	RA04, RA08, RA19, RA22, RA23, RA24, RA25, RA26, RA27
CE8.A5: Manejar herramientas informáticas que permitan el desarrollo de soluciones en el ámbito de la ingeniería biomédica.	RA02, RA06, RA08, RA20, RA27

## Bibliografía

- Bronson, Gary, *C++ para Ingeniería y Ciencias (2da. edición)*, International Thomson Editores, México, 2007
- Marzal, Andrés, Gracia Luengo, Isabel, García, Isabel, *Introducción a la programación con Python 3*, Universitat Jaume I, 2014
- Deitel, H. M., Deitel, P. J., *Cómo programar en C++*, Pearson Educación, 2015