

Programa del curso CA-2026

## **Introducción a la computación**

Escuela de Ingeniería Electromecánica  
Carrera de Ingeniería Electromecánica

## I parte: Aspectos relativos al plan de estudios

### 1. Datos generales

<b>Nombre del curso:</b>	Introducción a la computación
<b>Código:</b>	CA-2026
<b>Tipo de curso:</b>	Teórico
<b>Obligatorio o electivo:</b>	Obligatorio
<b>Nº de créditos:</b>	2
<b>Nº horas de clase por semana:</b>	4
<b>Nº horas extraclase por semana:</b>	2
<b>Ubicación en el plan de estudios:</b>	Curso de 2 <sup>do</sup> semestre en Ingeniería Electromecánica
<b>Requisitos:</b>	EE-0108 Introducción a la ingeniería electromecánica
<b>Correquisitos:</b>	Ninguno
<b>El curso es requisito de:</b>	EE-0305 Transductores
<b>Asistencia:</b>	Libre
<b>Suficiencia:</b>	Si
<b>Posibilidad de reconocimiento:</b>	Si
<b>Aprobación y actualización del programa:</b>	01/01/2026 en sesión de Consejo de Escuela 01-2026

## 2. Descripción general

El curso de *Introducción a la computación* aporta en el desarrollo de los siguientes rasgos del plan de estudios: desarrollar soluciones de software para el control y procesamiento de datos en sistemas electromecánicos; y desarrollar soluciones de hardware usando microcontroladores para el control y procesamiento de datos en sistemas electromecánicos.

Los aprendizajes que los estudiantes desarrollarán en el curso son: comprender los fundamentos de las ciencias de la computación y su aplicación en la resolución de problemas; desarrollar habilidades en el diseño y análisis de algoritmos, así como en la implementación y manipulación de estructuras de datos básicas; programar en lenguajes de alto nivel, utilizando variables, tipos de datos, operadores, estructuras de control y principios de programación modular; y integrar conocimientos de software para el diseño de sistemas computacionales desde el modelado con UML, diagramas de flujo hasta su implementación.

Para desempeñarse adecuadamente en este curso, los estudiantes deben poner en práctica lo aprendido en el curso de: Matemática general.

Una vez aprobado este curso, los estudiantes podrán emplear algunos de los aprendizajes adquiridos en los cursos de: Transductores, Métodos numéricos para ingeniería, y Microcontroladores.

## 3. Objetivos

Al final del curso la persona estudiante será capaz de:

### Objetivo general

- Desarrollar soluciones de software que permitan el control y procesamiento de datos en sistemas electromecánicos

### Objetivos específicos

- Comprender los fundamentos de las ciencias de la computación y su aplicación en la resolución de problemas
- Desarrollar habilidades en el diseño y análisis de algoritmos, así como en la implementación y manipulación de estructuras de datos básicas
- Programar en lenguajes de alto nivel, utilizando variables, tipos de datos, operadores, estructuras de control y principios de programación modular
- Integrar conocimientos de software para el diseño de sistemas computacionales desde el modelado con UML, diagramas de flujo hasta su implementación

## 4. Contenidos

En el curso se desarrollaran los siguientes temas:

1. Introducción a las ciencias de la computación
  - 1.1. Surgimiento de computación moderna
  - 1.2. Campos de aplicación
  - 1.3. Arquitectura de un sistema computacional
2. Lenguajes de programación
  - 2.1. Paradigmas de programación, tipos, diferencias.

- 2.2. Lenguajes de alto nivel interpretados
- 2.3. Lenguajes de alto nivel interpretados
- 2.4. Variables, tipos de datos y operadores
- 2.5. Representación digital de datos
- 3. Estructuras de control
  - 3.1. Condicionales
  - 3.2. Bucles
  - 3.3. Manejo de excepciones
  - 3.4. Representación gráfica de las estructuras
- 4. Estructuras de datos
  - 4.1. Arreglos
  - 4.2. Listas y tuplas
  - 4.3. Pilas y colas
  - 4.4. Acceso a memoria y reserva de memoria
  - 4.5. Punteros y aritmética de punteros
- 5. Funciones y programación modular
  - 5.1. Definición y uso de funciones
  - 5.2. Modularización del código
  - 5.3. Memoria temporal de las funciones
- 6. Modelado con UML
  - 6.1. Introducción a UML
  - 6.2. Diagramas de casos de uso
  - 6.3. Diagramas de flujo
  - 6.4. Diagrama de estados
  - 6.5. Modelado de base de datos
- 7. Algoritmos
  - 7.1. Definición y características
  - 7.2. Diseño y análisis de algoritmos
  - 7.3. Algoritmos de ordenamiento y búsqueda
  - 7.4. Implementación de máquinas de estado
  - 7.5. Rendimiento y comparación según el orden de complejidad.
- 8. Compiladores
  - 8.1. Proceso de compilación

## 8.2. Fases de un compilador

### II parte: Aspectos operativos

#### 5. Metodología

En este curso, se utilizará el enfoque sistémico-complejo para la ejecución de las sesiones magistrales y se integrará la investigación práctica aplicada para las asignaciones extraclase. Esta última se implementará mediante técnicas como el estudio de casos, el aprendizaje basado en proyectos, el modelado y la simulación.

**Las personas estudiantes podrán desarrollar actividades en las que:**

- Desarrollarán habilidades en la programación utilizando lenguajes de alto nivel, aplicando variables, tipos de datos, operadores, y estructuras de control.
- Utilizarán diagramas UML para modelar y diseñar sistemas computacionales.
- Integrarán conocimientos de software para desarrollar e implementar soluciones.

Este enfoque metodológico permitirá a la persona estudiante desarrollar soluciones de software que permitan el control y procesamiento de datos en sistemas electro-mecánicos

Si un estudiante requiere apoyos educativos, podrá solicitarlos a través del Departamento de Orientación y Psicología.

#### 6. Evaluación

La evaluación se distribuye en los siguientes rubros:

- Exámenes parciales: evaluaciones formales que miden el nivel de comprensión y aplicación de los conceptos clave del curso. Generalmente cubren una parte significativa del contenido visto hasta la fecha y pueden incluir problemas teóricos y prácticos.
- Tareas: evaluaciones que tienen el propósito de reforzar, aplicar o evaluar el aprendizaje de un tema específico. Pueden requerir investigación, resolución de problemas, desarrollo de habilidades prácticas o aplicación de conocimientos teóricos.
- Proyecto(s) grupal(es): actividad integradora donde los estudiantes aplican conocimientos teóricos y prácticos para resolver un problema real o simulado. Fomenta el desarrollo de habilidades analíticas, de investigación y trabajo en equipo.

Exámenes parciales (3)	60 %
Tareas (6)	25 %
Proyecto(s) grupal(es) (1)	15 %
Total	100 %

De conformidad con el artículo 78 del Reglamento del Régimen Enseñanza-Aprendizaje del Instituto Tecnológico de Costa Rica y sus Reformas, en este curso la persona estudiante tiene derecho a presentar un examen de reposición si su nota luego de redondeo es 60 o 65.

## 7. Bibliografía

- [1] T. H. Cormen, C. E. Leiserson, R. L. Rivest y C. Stein, *Introduction to Algorithms*. MIT Press, 2009.
- [2] A. S. Tanenbaum y T. Austin, *Structured Computer Organization*. Pearson, 2016.
- [3] D. A. Patterson y J. L. Hennessy, *Computer Organization and Design: The Hardware/Software Interface*. Morgan Kaufmann, 2020.
- [4] A. Silberschatz, H. F. Korth y S. Sudarshan, *Database System Concepts*. McGraw-Hill, 2020.
- [5] A. V. Aho, M. S. Lam, R. Sethi y J. D. Ullman, *Compilers: Principles, Techniques, and Tools*. Pearson, 2006.
- [6] R. E. Bryant y D. R. O'Hallaron, *Computer Systems: A Programmer's Perspective*. Pearson, 2015.

## 8. Persona docente

El curso será impartido por:

**Mag. Rellenar**  
Rellenar

Correo: xxx@itcr.ac.cr Teléfono: 0

Oficina: 0 Escuela: Ingeniería en Producción Industrial Sede: Cartago