



## IIC2343 - Arquitectura de Computadores (I/2025)

### Programa de curso

<b>Profesor(es)</b>	: Felipe Valenzuela ( <a href="mailto:frvalenzuela@alumni.uc.cl">frvalenzuela@alumni.uc.cl</a> ) Germán Contreras ( <a href="mailto:glcontreras@uc.cl">glcontreras@uc.cl</a> )
<b>Foro de dudas</b>	: <a href="#">Github</a>
<b>Talleres</b>	: Martes, Miércoles y Jueves módulos 5 y 6 (14:50 - 17:20) - Labs 1 y 5, Edificio San Agustín
<b>Horario de atención</b>	: Coordinar por correo
<b>Ayudante en jefe</b>	: Catalina Miranda ( <a href="mailto:ccmiranda1@uc.cl">ccmiranda1@uc.cl</a> )
<b>Coordinadora laboratorio</b>	: Javiera Pinto ( <a href="mailto:jpints@uc.cl">jpints@uc.cl</a> )

## 1. Presentación del curso

Los computadores son parte fundamental de nuestra sociedad: los ocupamos como herramienta de trabajo, comunicación y entretenimiento, controlan nuestros autos y medios de transporte, almacenan nuestra información personal y financiera. En resumen, permiten que el mundo tal como lo conocemos funcione. El propósito de este curso es que el alumno entienda qué es y cómo funciona un computador y conozca los principales conceptos relacionados con la arquitectura y programación de estos.

El curso considera aspectos teóricos y prácticos de la arquitectura de computadores. En relación a los elementos teóricos, los alumnos aprenderán los fundamentos, técnicas y herramientas que permiten diseñar y programar un computador, poniendo énfasis en la construcción, comunicación y coordinación de los elementos internos. Por su parte, los elementos prácticos del curso consideran en el diseño y construcción de distintos elementos de un computador utilizando hardware programable.

## 2. Objetivos de aprendizaje

Al final del curso, los alumnos serán capaces de:

- Conocer los distintos esquemas de representación de datos basados en codificación numérica binaria.
- Comprender el funcionamiento de un computador, con énfasis en los componentes internos y sus mecanismos de comunicación.
- Controlar un computador mediante la creación de programas en lenguajes de bajo nivel.
- Diseñar computadores en base al análisis y la evaluación de requerimientos, usando distintos tipos de microarquitecturas y conjuntos de instrucciones.
- Implementar físicamente componentes de un computador, utilizando herramientas computacionales de diseño y síntesis de hardware.

### 3. Contenido

A continuación, se presenta un desglose detallado de los contenidos del curso:

#### **Capítulo 1: Fundamentos**

- Representación de datos.
- Lógica booleana y circuitos digitales.
- Operaciones aritméticas y lógicas.
- Almacenamiento de datos.

#### **Capítulo 2: Diseño y programación de un computador**

- Programabilidad.
- Control de flujo.
- Subrutinas.
- Lenguaje de máquina y Assembly.

#### **Capítulo 3: Arquitecturas de Computadores**

- Microarquitectura.
- Arquitectura del conjunto de instrucciones (ISA).
- Ejemplos de arquitecturas.

#### **Capítulo 4: Comunicación**

- Dispositivos de entrada/salida (I/O).
- Comunicación entre CPU, memoria y dispositivos de I/O.
- Arquitectura de buses.

#### **Capítulo 5: Extensiones y mejoras al modelo de computador básico**

- Jerarquía de memoria.
- Multiprogramación.
- Paralelismo.

### 4. Metodología

La metodología del curso se basa en clases expositivas, fortaleciendo los conceptos mediante la discusión de ejemplos específicos y la realización de ejercicios teóricos y prácticos. La asistencia a la cátedra es voluntaria y como tal se espera que quienes asistan lleguen con puntualidad al comienzo de la clase y participen durante la sesión. Se asumirá que todo ejercicio propuesto será resuelto por los alumnos en su tiempo de estudio.

## 5. Evaluaciones

Las evaluaciones del curso estarán enfocadas al logro de los objetivos de aprendizaje. El desglose de las evaluaciones es el siguiente:

- Evaluaciones escritas ( $E$ ): el curso tendrá tres evaluaciones escritas de carácter individual, presencial y **obligatorio**. Si falta a una o dos evaluaciones escritas como máximo durante el semestre, podrá rendir una evaluación recuperativa siempre y cuando su inasistencia tenga una justificación emitida **por su unidad académica**. Si falta a las tres evaluaciones escritas del curso, una de ellas será evaluada con la nota mínima independiente de que exista justificación.
  - $I_1$ : Jueves 24 de abril, 17:30 hrs.
  - $I_2$ : Jueves 29 de mayo, 17:30 hrs.
  - $I_3$ : Jueves 03 de julio, 17:30 hrs.
  - $I_{\text{Recuperativa}}$ : Martes 08 de julio, horario a confirmar. Reemplaza la nota de la evaluación escrita a la que haya faltado con justificación.
  - $I_{\text{Recuperativa} \times 2}$ : Martes 08 de julio, horario a confirmar. Si recupera dos evaluaciones, rendirá una única evaluación que combinará preguntas de ambas interrogaciones y cubrirá ambas notas. No obstante, en este escenario, se le evaluará **con un 70 % de exigencia**.
- Actividad de programación ( $I_P$ ): Esta actividad, de carácter individual, presencial y obligatoria, será evaluada el día lunes 26 de mayo en el horario de clases de su sección y consistirá en el desarrollo de un problema de programación, por lo que requerirá el **uso de computador**. Si no asiste a esta instancia y **no justifica con su unidad académica**, se le calificará con la nota mínima. Si justifica su inasistencia, podrá optar a rendir una actividad recuperativa el martes 08 de julio, en un horario diferido respecto a las evaluaciones escritas. Esta inasistencia **es independiente** de la permitida en las interrogaciones.
- Actividades prácticas: el curso tendrá una evaluación práctica que corresponderá a la ponderación de las actividades realizadas en laboratorio durante el horario de taller y las presentaciones de avances del subsiguiente proyecto grupal. El objetivo es llegar a implementar un computador con funcionalidad básica y hacer programas que su implementación pueda ejecutar.

Fechas de actividades **obligatorias y con nota**:

- Actividad 1 ( $A_1$ ) : Semana del 17 de marzo.
- Actividad 2 ( $A_2$ ): Semana del 24 de marzo.

Finalizadas las actividades en laboratorio, los estudiantes se dividirán en grupos de un máximo de dos integrantes **dentro de su horario asignado** y continuarán con el proyecto. Deberán presentar sus avances en tres fechas:

- Entrega 1 de proyecto ( $P_1$ ): Lunes 07 de abril.
- Entrega 2 de proyecto ( $P_2$ ): Lunes 12 de mayo.
- Entrega 3 de proyecto ( $P_3$ ): Lunes 23 de junio.

## 6. Exigencias del curso

Para aprobar el curso, se debe cumplir con los siguientes requisitos:

- Las evaluaciones escritas son de carácter obligatorio.
- El promedio de las evaluaciones escritas ( $\bar{E}$ ) debe ser mayor o igual a 3,951 y se calcula de la siguiente manera:

$$\bar{E} = 0,3 * I_1 + 0,3 * I_2 + 0,3 * I_3 + 0,1 * I_P$$

.

- La nota de la actividades prácticas ( $A_P$ ) debe ser mayor o igual a 3,951 y se calcula de la siguiente manera:

$$A_P = 0,05 * A_1 + 0,05 * A_2 + 0,1 * P_1 + 0,5 * P_2 + 0,3 * P_3$$

- La nota final ( $N_F$ ) debe ser mayor o igual a 3,951 y se calcula de la siguiente manera:

$$N_F = 0,5 * \bar{E} + 0,5 * A_P$$

En caso de no cumplir alguno de los criterios, se reprueba con  $N_F$ :

$$N_F = \min(3,9; N_F)$$

## Política de Integridad Académica

Los alumnos de la Escuela de Ingeniería deben mantener un comportamiento acorde al Código de Honor de la Universidad:

*“Como miembro de la comunidad de la Pontificia Universidad Católica de Chile me comprometo a respetar los principios y normativas que la rigen. Asimismo, prometo actuar con rectitud y honestidad en las relaciones con los demás integrantes de la comunidad y en la realización de todo trabajo, particularmente en aquellas actividades vinculadas a la docencia, el aprendizaje y la creación, difusión y transferencia del conocimiento. Además, velaré por la integridad de las personas y cuidaré los bienes de la Universidad.”*

En particular, se espera que mantengan altos estándares de honestidad académica. Cualquier acto deshonesto o fraude académico está prohibido; los alumnos que incurran en este tipo de acciones se exponen a un procedimiento sumario. Específicamente, para los cursos del Departamento de Ciencia de la Computación, rige obligatoriamente la siguiente política de integridad académica. Todo trabajo presentado por un alumno (grupo) para los efectos de la evaluación de un curso debe ser hecho individualmente por el alumno (grupo), sin apoyo en material de terceros. Por “trabajo” se entiende en general las interrogaciones escritas, las tareas de programación u otras, los trabajos de laboratorio, los proyectos, el examen, entre otros. Si un alumno (grupo) copia un trabajo, los antecedentes serán enviados a la Dirección de Docencia de la Escuela de Ingeniería para evaluar posteriores sanciones en conjunto con la Universidad, las que pueden incluir reprobación del curso y un procedimiento sumario. Por “copia” se entiende incluir en el trabajo presentado como propio partes hechas por otra persona. Está permitido usar material disponible públicamente, por ejemplo, libros o contenidos tomados de Internet, siempre y cuando se incluya la cita correspondiente. Lo mismo aplica para herramientas de inteligencia artificial externas al curso.

En este curso, cualquier infracción Código de Honor derivará en la reprobación automática del curso y nota final  $N_F = 1, 1$ .

## 7. Bibliografía

- D. Patterson, Computer Organization and Design RISC-V Edition: The Hardware Software Interface. Morgan Kaufmann, 2020.
- Tanenbaum, A.; Structured Computer Organization, 6 Ed., Pearson, 2013.
- S. Harris, Digital design and computer architecture, RISC-V edition. Morgan Kaufmann, 2021.
- Apuntes de semestres pasados disponibles en el sitio web del curso.