



PROGRAMA DE ASIGNATURA

I. IDENTIFICACIÓN

Carrera o programa: Ingeniería en Tecnologías de Información

Unidad responsable: Escuela de Ingeniería

Nombre de la asignatura: Arquitectura y Organización de Computadores

Código: ECIN-00464

Semestre en la malla¹: 4

Créditos SCT – Chile: 5

Ciclo de Formación	Básico		Profesional	X
Tipo de Asignatura	Obligatoria	X	Electiva	

Clasificación de área de conocimiento²

Área: Ingeniería y Tecnología **Subárea:** Ingeniería Informática

Requisitos:

Prerrequisitos:	Requisitos para:
Programación Orientada a Objetos	Sistemas Operativos

II. ORGANIZACIÓN SEMESTRAL

Horas Dedicación Semanal (Cronológicas)	Docencia Directa	4.5		Trabajo Autónomo		3.5	Total	8
Detalle Horas Directas	Cátedra	Ayudantía	Laboratorio	Taller		Terreno	Exp. Clínica	Supervisión
	3			1.5				

III. APOORTE AL PERFIL DE EGRESO

La asignatura contribuye al dominio 1 del perfil de egreso, “Conocimiento científico y disciplinario”. Además, contribuye al dominio 4 “Habilidades para la Práctica de la Ingeniería”. Al finalizar la asignatura las y los estudiantes serán capaces de aplicar los conceptos fundamentales de la arquitectura y organización de computadores digitales del tipo Von Neumann y Harvard, de tal manera de poder definir, modificar y mantener la plataforma computacional de la empresa donde se desarrolle, sea ésta desde un Data Center hasta alternativas de plataformas computacionales dedicadas a la adquisición y distribución de datos en ambientes Industriales.

IV. COMPETENCIAS

La carrera declara las siguientes habilidades:

- 1.2. Aplicar conocimientos de ciencias de la ingeniería a la solución de problemas complejos de ingeniería.
- 1.3. Aplicar conocimientos, métodos y herramientas con un enfoque sistémico principalmente en la ejecución de proyectos TI en el ámbito de las ciencias de la computación, infraestructura TI e ingeniería de software.
- 4.4. Diseñar soluciones que involucren, por ejemplo, aplicaciones TI, infraestructura TI, toma de decisiones, gestión de datos y gestión de proyectos.

¹ Este campo sólo se completa en caso de carreras con programas semestrales.

² Clasificación del curso de acuerdo a la OCDE



4.5. Implementar soluciones que involucren, por ejemplo, aplicaciones TI, infraestructura TI, toma de decisiones, gestión de datos y gestión de proyectos.

V. RESULTADOS DE APRENDIZAJE

1. Distinguir el funcionamiento del computador a nivel de bloques funcionales.
2. Resolver la ruta de datos necesaria para la ejecución de una instrucción de bajo nivel al interior del procesador.
3. Construir un programa simple de bajo nivel.
4. Identificar los componentes internos de la tarjeta madre.
5. Identificar arquitecturas de alto rendimiento y tolerantes a fallos distintas configuraciones de computadores.
6. Aplicar pruebas de rendimiento entre distintas configuraciones de computadores.
7. Seleccionar los procesos, técnicas y herramientas adecuados de acuerdo a los requerimientos.
8. Desarrollar la solución tecnológica más adecuada en base a las características del problema y los recursos disponibles.

VI. ÁREAS TEMÁTICAS

1. Organización de Computadores
 - 1.1. Conceptos Computacionales: bit, byte, palabra, puertas lógicas, registros, multiplexor.
 - 1.2. Computador de Propósitos Generales.
 - 1.3. Computador de Propósitos Específico.
 - 1.4. Esquema funcional de un computador.
2. Arquitectura del Procesador
 - 2.1. Formato de una instrucción de bajo nivel.
 - 2.2. Ciclo de una instrucción.
 - 2.3. Estructura de la CPU a nivel de registro.
 - 2.4. Ruta de datos al interior del procesador para resolución de instrucciones de bajo nivel.
 - 2.5. Diseño de la fase de ejecución de las instrucciones de bajo nivel.
 - 2.6. Unidad de control micro programada.
 - 2.7. Implementación de instrucciones de bajo nivel a partir de un set de instrucciones micro programadas.
3. Taller de Programación de Bajo Nivel
 - 3.1. Programación de bajo nivel en SimuProc 4.15.
 - 3.2. Ejercicios de programación de bajo nivel.
 - 3.3. Seguimiento de la ejecución de un programa de bajo nivel al interior del procesador.
 - 3.4. Programación en bajo nivel en Emu8086.
4. Componentes de la Organización de un Computador.
 - 4.1. Unidad aritmética lógica.
 - 4.2. Interrupciones de software y hardware.
 - 4.3. Pipeline.
 - 4.4. Hyperthreading
 - 4.5. Arquitectura dual core.



- 4.6. Chip Set modernos.
- 4.7. Jerarquía de memoria en el computador.
- 4.8. Métodos de acceso a la memoria.
5. Arquitecturas de Alto rendimiento y Tolerantes a Fallos.
 - 5.1. Procesadores Multicore.
 - 5.2. Procesadores GPU.
 - 5.3. Clúster de Computadores.
 - 5.4. Máquinas Virtuales
 - 5.5. Arquitecturas Tolerantes a fallos
 - 5.6. Raid de discos.
6. Benchmark de Computadores
 - 6.1. Técnicas de comparación de procesadores
 - 6.2. Técnicas de comparación de memoria RAM
 - 6.3. Técnicas de comparación de comportamiento de la memoria.
 - 6.4. Comparación del sistema como un todo.

VII. ORIENTACIONES METODOLÓGICAS

1. La metodología a desarrollar en esta asignatura debe favorecer la interacción entre las y los estudiantes a través de trabajos prácticos colaborativos que permitan la solución a problemas específicos contextualizados a la asignatura.
 - Se sugiere el uso de clases expositivas y participativas con método combinado, es decir, clases expositivas con alternancia de trabajos en grupo de corta duración para responder preguntas.
 - Se sugiere la utilización de la metodología activa de análisis de casos para desarrollar experiencias que permitan incorporar los elementos teórico prácticos asociados a los resultados de aprendizaje de la asignatura.
2. Las experiencias de cátedra/laboratorio/taller deben ser realizadas por medio de la utilización de software moderno aplicable a la asignatura.
3. Se recomienda que las y los estudiantes realicen presentaciones periódicas sobre el trabajo realizado que incluya: contextualización, desarrollo y conclusiones.
4. Actividades prácticas recomendadas: cápsulas teóricas, reuniones de trabajo, taller de trabajo en equipo y liderazgo, presentaciones e informes escritos de avance en español, revisión del estado del arte asociado al problema, lluvia de ideas, análisis de alternativas y descripción detallada de la solución.

VIII. ORIENTACIONES Y CRITERIOS PARA EVALUACIÓN

1. Se recomienda la aplicación de una evaluación diagnóstica al inicio de la asignatura.
2. La asignatura podría contemplar dos instancias de evaluación de los resultados de aprendizaje: cátedra y taller/laboratorio.
 - En el caso de existir, ambas debieran ser aprobadas por separado: el porcentaje de cada una de ellas deberá ser de 60% para cátedra y 40% para taller/laboratorio.
 - En el caso que la asignatura tenga actividades de taller/laboratorio, éstas deben ser realizadas en grupos de estudiantes y se recomienda la elaboración por parte de los estudiantes de un informe sobre la actividad desarrollada.



3. Se evaluará el conocimiento conceptual y procedimental mediante el desarrollo de al menos dos pruebas sumativas de carácter presencial.
 - Se recomienda además la aplicación de una evaluación mediante la entrega de un trabajo desarrollado en las horas indirectas asociadas a la asignatura.
 - Se recomienda que las y los estudiantes realicen una o más presentaciones de los trabajos realizados, la evaluación de la misma debe ser por medio de la aplicación de una rúbrica.
4. Se recomienda realizar evaluaciones de carácter formativo. Esto permite al docente introducir correcciones, añadir alternativas y reforzar los aspectos para ayudar al estudiantado en el logro de sus habilidades.
5. La asistencia y condiciones de aprobación de la asignatura debe ser acorde a la aplicación del Reglamento de Docencia de Pregrado.

IX. RECURSOS BIBLIOGRÁFICOS

Bibliografía Mínima

- Stallings, W. (2006). Organización y Arquitectura de Computadores. Prentice Hall.
- Tanenbaum, A. S., Escalona García, L. R. (2000). Organización de computadoras: un enfoque estructurado. España: Prentice Hall Hispanoamericana.

Bibliografía Complementaria

- Ortega, J., Mancía Anguita, M., & Prieto, A. (2005). Arquitectura de Computadores. Thomson.
- Brey, B. B. (2001). Los microprocesadores intel: arquitectura, programación e interfaz de los procesadores 8086/8088, 80186/80188, 80286, 80386, 80486, Pentium, Pentium Pro y Pentium II. México: Pearson Educación.
- Palacios Municio, E. (2009). Microcontrolador PIC16F84. Desarrollo de proyectos. 3a edición. España: Rama S.A. Editorial y Publicaciones.