



PROGRAMA DE ASIGNATURA

I. IDENTIFICACIÓN				
Carrera o programa: Ingeniería en Tecnologías de Información				
Unidad responsable: Escuela de Ingeniería				
Nombre de la asignatura: Sistemas Operativos				
Código: ECIN-00564				
Semestre en la malla¹: 5				
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:	
<ul style="list-style-type: none">Arquitectura y Organización de Computadores			<ul style="list-style-type: none">Redes de Computadores I	

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”. También contribuye al dominio 3 “Habilidades Interpersonales”. Por último, contribuye al dominio 4 “Habilidades para la Práctica de la Ingeniería”. Al finalizar la asignatura las y los estudiantes serán capaces de administrar los recursos del computador a través del manejo de los procesos que desarrollan las tareas de los usuarios, siempre desde la perspectiva de otorgar el mejor servicio posible, en tiempo de respuesta, disponibilidad y seguridad, con énfasis en sistemas operativos modernos Linux y Windows, y su desarrollo sobre arquitecturas de multi núcleo que facilitan la ejecución paralela de las tareas.

IV. COMPETENCIAS
La carrera declara las siguientes habilidades: 1.1. Aplicar conocimientos de matemáticas y ciencias naturales a la solución de problemas complejos de ingeniería. 1.2. Aplicar conocimientos de ciencias de la ingeniería a la solución de problemas complejos de ingeniería.

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

² Clasificación del curso de acuerdo a la OCDE



- 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.
- 3.2. Comunicar comprensivamente información técnica en español, en forma oral, escrita, y gráfica, a nivel avanzado.
- 4.5. Implementar soluciones que involucren, por ejemplo, aplicaciones TI, infraestructura TI, toma de decisiones, gestión de datos y gestión de proyectos.
- 4.6. Operar soluciones que involucren, por ejemplo, aplicaciones TI, infraestructura TI, toma de decisiones, gestión de datos y gestión de proyectos.

V. RESULTADOS DE APRENDIZAJE

1. Diferenciar las funciones del sistema operativo como gestor de recursos del computador y la interacción con los usuarios.
2. Aplicar técnicas para la caracterización, creación y administración de procesos e hilos como flujos de tareas creando programas capaces de realizar la función de sincronización de tareas.
3. Asociar las técnicas utilizadas por el sistema operativo para la administración de recursos.
4. Distinguir la estructura, medios de acceso, seguridad y respaldo de los sistemas de archivos.
5. Analizar el comportamiento de cada componente del sistema operativo y del sistema computacional en su conjunto, desde la perspectiva de la calidad de servicio para los usuarios.
6. Desarrollar la solución tecnológica más adecuada en base a las características del problema y los recursos disponibles.
7. Evaluar el desempeño de la arquitectura TI, los sistemas de software, y las plataformas de cómputo y comunicaciones

VI. ÁREAS TEMÁTICAS

1. Introducción
 - 1.1 Definición y clasificación de sistemas operativos
 - 1.2 Definición y caracterización de procesos
 - 1.3 Identificación de funciones del sistema operativo: administración de recursos, comunicación con los usuarios, registro de actividades.
 - 1.4 Estructura en capas del sistema operativo.
2. Administración y sincronización de procesos concurrentes.
 - 2.1 Definición de concurrencia.
 - 2.2 Concepto de paralelismo
 - 2.3 Región crítica
 - 2.4 Sincronización de tareas
 - 2.5 interbloqueo
 - 2.6 Presentación de casos.
3. Administración de recursos.
 - 3.1 Planificación del procesador.
 - 3.2 Gestión de la memoria virtual.
 - 3.3 Operación de periféricos.
4. Casos de estudio Linux y MSWindows



- 4.1 Núcleo del sistema
 - 4.2 Lenguaje de comandos
 - 4.3 Administración de usuarios
 - 4.4 Estructura y seguridad del sistema de archivos
 - 4.5 Comunicación con aplicaciones (API)
 - 4.6 Creación y sincronización de procesos e hilos.
 - 4.7 Virtualización de plataformas.
-
5. Análisis de rendimiento.
 - 5.1 Parámetros de rendimiento.
 - 5.2 Modelo de carga de trabajo.
 - 5.3 Ecuaciones del modelo.
 - 5.4 Asíntotas de rendimiento: Tiempo de respuesta y productividad.

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 un método combinado, es decir, clases expositivas alternadas con 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 la realización 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

- Tanenbaum, A. S. (2003). Sistemas operativos modernos. España: Pearson Educación.
- Stallings, W. (2005). Sistemas operativos: aspectos internos y principios de diseño. España: Pearson Educación.

Bibliografía Complementaria

- Silberschatz, A., Gagne, G., Galvin, P. B. (2006). Fundamentos de sistemas operativos. Colombia: McGraw-Hill Interamericana de España S.L.
- Hart, J. M. (1997). Win32 System Programming. Reino Unido: Addison-Wesley Developers Press.
- Matthew, N., Stones, R. (2011). Beginning Linux Programming. Alemania: Wiley.
- Steven, R., Robbins, K. A. (2000). Unix Programación Práctica. Guía Para la Concurrencia, la Comunicación Y Los Multihilos. (n.p.): Prentice Hall.
- Bach, M. J. (1986). The Design of the UNIX Operating System. Reino Unido: Prentice-Hall.