

PLAN DE ESTUDIOS: 2004 Ajuste 2011
ANO ACADEMICO: 2013
CARRERA: Ingeniería en Electrónica.

1. OBJETIVOS:

Los objetivos de la asignatura de Sistemas Operativos pueden agruparse de la siguiente manera:

a- De la cátedra

- ❖ Orientar y coordinar el Aprendizaje por parte del alumno que curse la materia en los conceptos fundamentales de la arquitectura y diseño de los Sistemas Operativos (SOp) vigentes en el mercado.
- ❖ Introducirlo en el desarrollo de Programas y Aplicación de Utilidades que conforman los Sistemas Operativos, en los ambientes de trabajo del futuro Ing. en Telecomunicaciones.

b- Del alumno

- ❖ Conformar un Aprendizaje teórico-práctico, sobre los SOp en uso, llevado de una intensa actividad de cursado e investigación que volcará luego en la confección y aprobación de un conjunto de trabajos prácticos y evaluaciones parciales teórico-prácticas.
- ❖ Lograr una concepción global y un enfoque selectivo, para realizar soluciones algorítmicas, de desarrollo de programas y aplicación de métodos de solución, de los diferentes problemas que se presentan entre la interacción del Hardware y los SOp, con la orientación de solución de problemas de electrónica y telecomunicaciones.

2. Contenidos:

La Cátedra de Sistemas Operativos se encuentra orientada a que los Alumnos desarrollen las siguientes bases conceptuales y prácticas.

- ❖ Estructura y funcionalidad de un Sistema Operativo. (Unidades 1 y 2)
- ❖ Conformación de Administradores, y su lógica de trabajo.(Unidades 3, 4, 5)
- ❖ Práctica y desarrollo de aplicaciones que conforman los diversos Administradores de un Sistema Operativo. (Unidades 6 y 7)

Unidad I: Arquitectura de los Sistemas Operativos

1. Sistema Operativo: Concepto. Estructura. Kernel y MicroKernel.
2. Terminología básica: monoprogramación, multiprogramación, monousuario, multiusuario, multiprocesamiento, multitarea, portabilidad, sistema de tiempo real.
3. Evolución histórica de los Sistemas Operativos. Clasificación: monolítica, jerárquica, virtual, Cliente/Servidor, de tiempo real.
4. Entorno de desarrollo.
 - ❖ Proceso de compilación. Compilador GNU (gcc).
 - ❖ Control del proceso de compilación, make de GNU.
 - ❖ Administración de errores.

- ❖ Ejemplos de desarrollo.

Unidad II: Procesos - Administración de Procesos

1. Introducción a los problemas del procesador: modelo de estados de los procesos. Planificación de trabajos. Ordenamiento de los procesos. Orden de ejecución de procesos, trabajos y thread. Colas y prioridades.
2. Estructuras de control del Administrador de Procesos (Tabla de Memoria, Tabla de E/S, Tabla de Archivos, Tabla de Procesos y PCB). Modos de Ejecución.
3. Estados de los Procesos: Sistema de dos estados, creación y terminación. Modelo de cinco estados.
4. Planificación de Procesos
Conceptos básicos. Introducción. Planificación en Largo, Mediano y Corto Plazo.
Planificación de los Procesos: No-preemptive y Preemptive scheduling.
Algoritmos de Planificación No-Preemptive: FCFS, SJF y Prioridad. Comentarios.
Algoritmos de Planificación Preemptive: Round-Robin, Múltiples colas fijas, Múltiples colas dinámicas. Comentarios.
5. Gestión de Procesos en Unix - Linux.
6. Entorno de desarrollo de Procesos:
 - ❖ Conceptos y definiciones.
 - ❖ Atributos de un proceso.
 - ❖ Manipulación de Procesos.
 - ❖ Ejemplos de desarrollo.

Unidad III: Sincronización e IPC

1. Concurrencia. Principios Generales.
2. Exclusión Mútua. Soluciones por Sw y Hw. Abrazo Mortal (Deadlocks, Interbloqueo o bloqueo mutuo): Condiciones. Inanición. Métodos de Solución.
3. IPCs: Pipes & FIFO, Espacios de Memoria Compartida, Semáforos, Monitores, Pasaje de Mensajes y Sockets. Problema de Productor-Consumidor.
4. Entorno de Desarrollo: Solución a problemas de sincronización y de uso de IPC.

Unidad IV: Señales y Llamadas del Sistema

1. Conceptos sobre señales e Interrupciones.
2. Envío de señales. Intercepción de señales.
3. Llamadas al sistema. Utilización de llamadas al sistema.
4. Entorno de Desarrollo. Ejemplos aplicación de señales e interrupciones.

Unidad V: Administración de Memoria

1. Conceptos de la Administración de Memoria. Concepto de rutinas reusables y reentrantes.
Esquemas de administración de memoria: monoprogamación y multiprogamación.

2. Protección de memoria: necesidad de importancia de protección de memoria en los SOP multiprogramables, esquemas de protección (registros base y límite, clave de protección).
3. Administración de Memoria por Particiones. Fijas, Variable, Sistema de Colegas. Problemas de Fragmentación.
4. Administración de Memoria Virtual. El swapping: concepto. swap-in, swap-out, el tiempo del multiplexado de los recursos (trashing).
5. La paginación: concepto de página y cuadro de página. Las tablas y el direccionamiento. Funciones implementadas por el Hardware y por el SOP. Fragmentación interna. Distintos algoritmos de reemplazo de página. .
6. Segmentación Combinación de segmentación y administración por demanda. Ventajas y desventajas de los sistemas de segmentación.
7. Entorno de Desarrollo. Ejemplos aplicación administración de Memoria.

Unidad VI: Sistema de Archivo

1. Introducción. Organización y acceso a archivos: Pilas, archivos secuenciales, archivos secuenciales indexados, archivos indexados y archivos directos o de dispersión. Organización de directorios: Contenidos, estructura y designación.
2. Compartición de archivos: Derechos de acceso, acceso simultáneos, agrupación de registros.
3. Gestión del almacenamiento secundario: Asignación de archivos, gestión del espacio libre y fiabilidad.
4. Gestión de archivos en Unix/Linux: I-nodo y asignación de archivos.
5. Entorno de Desarrollo. Ejemplos aplicación administración de Archivos..

Unidad VII: Administración de dispositivos de entrada-salida.

1. Introducción: funciones generales.
2. Necesidad de la administración de periféricos. Controladores y canales. Implementación de controles.
3. Esquema de asignación de periféricos: exclusiva, compartida o virtual, automática o por operador. Consideraciones a tener en cuenta para compartir discos entre varios SOP
4. Principios del Hardware entrada-salida: *device controllers* y *drivers*. DMA y dispositivos de entrada-salida. Discos. Terminales. *Clocks*. *Handlers* de dispositivos y de *IRQs*. Concepto de RAID.
5. Concepto de método de acceso: acceso como interface standard entre programas y Hardware, funciones de los métodos de acceso, clases de métodos de acceso, invocación a los métodos de acceso, ejemplos.
6. Concepto de Buffer de datos. Concepto de bloqueo y desbloqueo de registros. Concepto de programa de Canal.
7. Funciones de supervisor de entrada-salida. Administración de canales y periféricos: algoritmo de selección, encolamiento de requerimiento de entrada-salida (por periférico, por canal, por red).
8. Ordenamiento de la entrada-salida: estado y asignación.

3. BIBLIOGRAFIA

3.1. Secundaria

- ❖ Administración de Sistemas Linux. Guía Avanzada. M.Carling, Stephen Degler y James Dennis. Prentice Hall, 2000.
- ❖ Operating Systems, 3th edition. William Stallings. Prentice Hall, 2001.
- ❖ Sistemas Operativos, Diseño e implementación. Andrew Tanenbaum. Prentice Hall, 1994.
- ❖ Sistemas Operativos Modernos. Andrew Tanenbaum, Prentice Hall, 1996.
- ❖ Cátedra de Sistemas Operativos, Apuntes. Ing. Netzeel. UTN. 1997.
- ❖ El Libro de Unix. Syed M. Sarwar, Robert Koretsky y Syed A. Sarwar. Ed. Addison Wesley, 2004. España.
- ❖ UNIX Sistema V Versión 4. Rosen, Rozinsky y Farber. McGraw Hill. Edic. 2002.
- ❖ LINUX, Edición especial. Tackett Jack, Gunter David y Brown Lance. Ed. Prentice Hall. 1998.
- ❖ UNIX para impacientes. Abrahams. Addison Wesley Iberoamericana.
- ❖ La Arquitectura de Windows NT a fondo. Solomon.
- ❖ Manual de Seguridad en Windows NT. Sheldon.
- ❖ Windows NT Server 4.0. Kit de desarrollo. Microsoft.
- ❖ Asi es Microsoft Windows 2000 Server. Northrup.
- ❖ Linux firewalls. Ziegler. New Riders Publishing.
- ❖ Manual de Actualización y reparación de PCs, 12 edición. Scott Mueller. Que, Prentice Hall, 2001.
- ❖ Sistemas Operativos una Visión Aplicada. Jesús Carretero Pérez, Pedro de Miguel Anasagasti, Félix García Carballeira y Fernando Pérez Costoya. McGraw-Hill. 2001.

3.2. Primaria

- ❖ Programación en Linux, con ejemplos. Kurt Wall. QUE, Prentice Hall. Madrid. 2000.
- ❖ Sistemas Operativos, principios de diseño e interioridades. Stallings William. 5ta Edición (Esp). Prentice Hall. Madrid. 2006.
- ❖ Sistemas Operativos Modernos. Andrew. S. Tanenbaum. Prentice-Hall. Interamericana S.A. Madrid, 2009.

3.3. Consulta

- ❖ El Libro de Unix. Syed M. Sarwar, Robert Koretsky y Syed A. Sarwar. Ed. Addison Wesley, 2004. España.
- ❖ El Libro de Linux. Syed M. Sarwar, Robert Koretsky y Syed. A. Sarwar. Ed. Addison Wesley. 2007. España.
- ❖ Sistemas operativos. Conceptos fundamentales. Silberschatz-Peterson-Galvin. 3ra. Edición Addison Wesley. 1994.

4. METODOLOGIA DE LA ENSEÑANZA

El alumno tendrá acceso a un conjunto de Actividades que le permitirá conformar su entorno de Aprendizaje, los cuales se desarrollarán en los siguientes lugares:

- a. Actividades de Enseñanza en el Aula. Clases Grupales de tipo Teórico.
- b. Actividades de Práctica en Laboratorios de Computadoras, que le permitirán familiarizarse con el ambiente de trabajo, y desde allí construir en la operación su Aprendizaje.
- c. Actividades de Investigación aplicada en los Trabajos Prácticos, de tipo Grupal. Cada Investigación deberá concluir con la correspondiente **presentación de la documentación** escrita y con el correspondiente medio de almacenamiento (disquette – cd – dvd). En todos los casos la funcionalidad es la base de la corrección y aprobación de la asignatura.

Lo anterior será posible materializarlo con trabajos de investigación, desarrollo de aplicaciones y evaluaciones de tipo Individual y Grupal. Estas pueden listarse de la siguiente forma:

- ❖ **Prácticas de Resolución de problemas**
- ❖ **Prácticas de Laboratorio**
- ❖ **Prácticas de Simulación**
- ❖ **Prácticas de Programación**
- ❖ **Prácticas de Diseño y Proyecto**
- ❖ **Presentaciones de temas específicos por Alumnos.**
- ❖ **Trabajos de Campo y Visitas a Plantas**

Conformación de los Trabajos Prácticos y Ensayos a realizar por los Alumnos a lo largo de su cursación.

Unidad temática	Trabajo Práctico	Res. Probl	P.Lab	Pr. Simul.	Pr. Program	Pr. Dis. Y Proy.	Hs.
I. Arquitectura de Sistemas Operativos	TP1: Introducción Operación de Sistema de Archivos	50%	80%	30%	0%	0%	6hs
	TP2 Programación de Shell Script	20%	100%	0%	100%	15%	12hs
II. Procesos	TP3: Procesos y Programación de Utilidades de Desarrollo	40%	50%	40%	60%	10%	6hs

III. Sincronización e IPC	TP4: Sincronización e IPC	30%	30%	25%	60%	30%	6hs
IV: Señales y Llamadas al Sistema	TP5: Señales y Llamadas de Sistemas.	40%	100%	0%	100%	60%	6hs
V: Administración de Memoria							
VI: Sistemas de Archivos							
VII: Sistemas de Entrada/Salida	TP 6: Sistemas de Entrada/Salida	25%	100%	25%	75%	25%	12hs
						Total Hs Practicas	48hs

Esto permitirá al alumno:

- ❖ Adquirir vocabulario técnico-informático y utilizarlo con precisión
- ❖ Conocer en forma amplia y general el funcionamiento de las partes de un SOp.
- ❖ Evaluar, a nivel de implementación, cualquier SOP. sobre equipos existentes en plaza.
- ❖ Desarrollar en el alumno el interés por la investigación; utilizando publicaciones, libros y sistemas reales que sean propuestos por el Profesor.
- ❖ Ayudar a desarrollar en el alumno la actitud de detector de posibles soluciones de problemas, realizando una adecuada ejercitación práctica.

Los Alumnos además deberán desarrollar una serie de Ensayos, con formato predefinido, en los cuales se explorarán sobre temas concernientes a los vistos en Aula, y deberán ser entregados, con una frecuencia de 15 días. Los temas propuestos son:

- ❖ ¿Es la virtualización una salida práctica y de bajo costo para compensación de Servidores?
- ❖ ¿El Software Libre puede ser una opción hoy para el segmento bancario y financiero.?
- ❖ ¿Pueden desarrollarse arquitecturas de SOs sin administradores ni drivers?
- ❖ Para trabajar con las herramientas de SOs qué prefiere: ¿Shell carácter o Shell gráfico?
¿Por qué?
- ❖ ¿Porqué las empresas del medio argentino prefieren BSD sobre Linux?
- ❖ ¿Qué es más complejo, programar un juego, o un scheduler?

Todos los ensayos deben declarar bibliografía y otras fuentes, además no pueden superar el 30% del escrito, lo textual de terceros. Aquí se puede plantear del criterio restante; presentación de temas específicos por alumno.

5. CRITERIOS DE EVALUACION

La primera evaluación que debe rendir un Alumno que cursa Sistemas Operativos es la Evaluación Diagnóstica.

El alumno deberá rendir un parcial y un recuperatorio (para el caso de no aprobación del primer nombrado) con fecha determinada por cronograma general de la Universidad.

Entrega de TP resueltos, por grupos de 2 o 3 alumnos.

Examen Final de toda la materia cursada.

Cualquiera de los exámenes se puede rendir en aula o en computadora.