

2015

**Universidad de
Belgrano**

**Facultad de Ing. y
Tecnología en
Informática**

**SISTEMAS OPERATIVOS
PROGRAMA ANALITICO
ING. EN INFORMATICA**

Ing. Sergio Omar Aguilera – sergio.aguilera@comunidad.ub.edu.ar

ASIGNATURA:	Sistemas Operativos
PLAN DE ESTUDIOS:	2010
ANO ACADEMICO:	2015
CARRERA/S :	Ingeniería en Informática.
PROFESOR a CARGO:	<i>Ing. Sergio Omar Aguilera</i>
CUATRIMESTRE:	<i>Primer Cuatrimestre.</i>

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 profesional en Informática.

b- Del alumno

- ❖ Conformar un aprendizaje teórico-práctico, sobre los Sistemas Operativos 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 instalación, configuración y Explotación de Sistemas Informáticos y Computacionales.

c- Según Contenidos Mínimos

Completar los siguientes CMs:

Sistemas Operativos, concepto, evolución, estructura y servicios que brindan. Inicialización y configuración de un Sistema Operativo. Virtualización. Concepto de proceso, comunicación entre procesos, problemas de sincronización y comunicación, uso de IPC, planificación de procesos. Sistema de archivos, protección. Administración de memoria, memoria virtual, su administración. Administración de entrada/salida. Seguridad del Sistema Operativo.

2. Contenidos

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

- ❖ Repaso de la estructura general y funcionalidad de un Sistema Operativo (Unidad 1). Procesos como unidad de ejecución. Thread o hilos de Ejecución. (Unidad 2)
- ❖ Problemas y métodos de Comunicación entre módulos de ejecución. Estructuras de Programación aplicadas a la intercomunicación (Unidades 3 y 4)

- ❖ Estructura, operación y diseño de un Sistema de Archivos.(Unidad 6)
- ❖ Práctica y desarrollo de aplicaciones que conforman los diversos Administradores de un Sistema Operativo. (Unidades 7 y 8)
- ❖ Seguridad en los Sistemas Operativos, tanto en Clientes como en Servidores. (Unidad 9)

Unidad 1: Arquitecturas de Sistemas Operativos.

1. Arquitectura de los Sistemas Operativos. Conceptos y funcionalidades.
2. Ambientes de trabajo. Centralizados , Descentralizados y Distribuidos. Inicialización y configuración de Sistemas Operativos. Virtualización.
3. Conceptos básicos de los Sistemas Operativos.
4. Administradores de un SOp.
5. Software de Sistema..

Unidad 2: Introducción y planificación de procesadores

1. Procesos. Concepto, creación y muerte de procesos. Traza.
2. Ambiente de trabajo de los Procesos.
3. Introducción a los problemas del procesador: modelo de estados de los procesos.
4. Estados de Suspensión.
5. Traza de ejecución de los procesos.

Unidad 3: Planificación de los Procesos

1. Repaso de los estados de los procesos.
2. Planificación de Procesos. Condiciones de contexto.
3. Planificación. Tipos.
4. Ordenamiento de los procesos. Orden de ejecución de procesos, trabajos y thread. Colas y prioridades.
5. Planificación de los Procesos: No-preemptive y Preemptive scheduling.
6. Algoritmos de Planificación No-Preemptive: FCFS, SJF y Prioridad. Comentarios.
7. Algoritmos de Planificación Preemptive: Round-Robin, Múltiples colas fijas, Múltiples colas dinámicas. Comentarios.
8. Gestión de Procesos en Unix - Linux.
9. Entorno de desarrollo de Procesos:

Unidad 4: Sincronización de Procesos

1. Concurrencia. Principios Generales. Interacción entre los procesos.
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:
5. Proceso de compilación. Compilador GNU (gcc).
 - ❖ Control del proceso de compilación, make de GNU.
 - ❖ Administración de errores.

- ❖ Ejemplos de desarrollo.
- ❖ Solución a problemas de sincronización y de uso de IPC.

Unidad 5: 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 6: 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 Sistemas Operativos multiprogamables, 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 7: Sistema de Archivos

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 8: Administración 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 controlers y drivers. DMA y dispositivos de entrada-salida. Discos. Terminales. Cloks. Handlers de dispositivos y de IRQs. Concepto de RAID.
5. Concepto de método de acceso: acceso como interfase estándar 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.

Unidad 9: Seguridad en Sistemas Operativos

1. Protección: Introducción al problema, diferencia entre protección y seguridad. Objetivos de la protección, dominios, matriz de accesos, lista de controles de accesos, revocación de derechos de acceso. Sistemas basados en capacidades y objetos. Protección basada en lenguajes.
2. Seguridad: Introducción al problema. El problema de la seguridad. La autenticación. Seguridad externa: física y administración. Seguridad interna: de operación, supervisión de amenazas. Protección por contraseñas. Controles de acceso.
3. Seguridad por Hardware, por Software. Hilos de programas (program threads). Hilos de sistema (system free). Monitoreo de los hilos (threads monitoring). Encartación. Virus, troyanos, gusanos.

3. BIBLIOGRAFIA

3.1. Principal - 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.

3.2. Consulta - Secundaria

- ❖ El Libro de Linux. Syed M. Sarwar, Robert Koretsky y Syed. A. Sarwar. Ed. Addison Wesley. 2007. España.
- ❖ Sistemas Operativos. Gary Nutt. 3er. Edicion. Ed. Pearson Educación. 2004.
- ❖ Sistemas operativos. Conceptos fundamentales. Silberschatz-Peterson-Galvin. 3ra. Edición Addison Wesley. 1994.
- ❖ Sistemas Operativos, un enfoque basado en conceptos. D.M. Dhamdhere. Ed. Mc Graw Hill. Mexico DF. 2008.
- ❖ Creación de Distribuciones Linux. Personalización para cada necesidad. Claudio Peña Millahual. Edit. MP. Manuales de Users. 2009.

3.3. Referencia

- ❖ 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. Microsoft. 2000
- ❖ Manual de Seguridad en Windows NT. Sheldon. Microsoft. 2001
- ❖ Linux firewalls. Ziegler. New Riders Publishing. 2002
- ❖ 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. 1er Edición. Editorial McGraw-Hill. 2001.

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** por mail y con el correspondiente medio de almacenamiento (disquette – cd – dvd). En todos los casos la correcta funcionalidad es la base de la corrección y aprobación de la asignatura.

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.
- ❖ Iniciarse en las actividades de trabajo que conlleva el desarrollo colaborativo con el propio grupo o con grupos más extensos, a nivel local y remoto.

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	TP – Aplicativo	Comentarios
1. Estructuras de Ejecución	TP1. Utilidades de Aplicación	Instalación y Comandos de Linux.
	TP 2. Procesos.	Uso de gcc y resolución de problemas de procesos.
2. Sincronización e IPC	TP 3. Sincronización e IPC.	Planificación y Solución a problemas de diseño.
3. Señales y Llamadas al Sistema	TP 4. Señales y Llamadas de Sistema.	Programación de Señales.
4: Administración de Memoria	TP 5. Sistemas de I/O y Memoria.	Resolución de problemas de memoria e I/O, usando métodos pre-establecidos
5: Administración del Sistema de Archivos	TP 6. Archivos y Entrada/Salida.	Solución a problemas de administración de archivos. Entrada y Salida al Sistema.
6: Administración de Entrada / Salida		
7: Seguridad en Sistemas Operativos	TP 7. Seguridad en SOp.	Aplicación de problemas de transferencia de información.

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 explicitarán 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 SOps sin administradores ni drivers?
- ❖ Para trabajar con las herramientas de SOps que prefiere, Shell carácter o Shell gráfico, porque?
- ❖ Porque las empresas del medio argentino prefieren BSD sobre Linux?
- ❖ Que es más complejo, programar un juego, o un administrador de 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 primer 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.

6. AUTORIDADES DE LA UNIVERSIDAD DE BELGRANO

Dr. Avelino Porto Presidente

Dr. Eustaquio Castro Vicepresidente de Gestión Técnica y Administrativa

Prof. Aldo J. Pérez Vicepresidente de Gestión Institucional

7. AUTORIDADES DE LA FACULTAD DE INGENIERIA.

Ing. Alberto Guerci. Decano

Lic. Paula Angelleri. Directora de Carrera.

Ing. Marcelo Romeo. Coordinador de Area.