

Universidad de Costa Rica  
Recinto Tacaes de Grecia  
Bachillerato en Informática Empresarial  
Sigla del curso: IF-4001  
Nombre del curso: Sistemas Operativos  
Requisitos: IF-3000, IF-3001.  
Horas lectivas semanales: 8  
Créditos: 4  
Horario: L: 8:00 a.m. – 11:50 a.m., M: 1:00 p.m. – 4:50 p.m.  
Horas consulta: M: 5:00 p.m. - 7:30 p.m.  
Profesor: MCI. Alberto Elías Ramírez Molina  
Correo electrónico: [alberto.ramirezmolina@ucr.ac.cr](mailto:alberto.ramirezmolina@ucr.ac.cr),  
[rammoli@gmail.com](mailto:rammoli@gmail.com)  
MSN: [rammoli@live.com](mailto:rammoli@live.com)  
Tel: 8816-2466  
II Semestre, 2010.

---



## DESCRIPCIÓN DEL CURSO

Este curso se orienta al estudio del diseño de los sistemas operativos y de sus mecanismos de implementación interna. Partiendo de lo anterior, se analizan los sistemas operativos más comunes, su estructura interna y los mecanismos de diseño adecuados para construir sistemas operativos. Además, se estudian los efectos que tienen las nuevas tecnologías informáticas sobre los sistemas operativos.

En lo que se refiere a casos de estudio, se contemplan varios sistemas operativos: Locales, de red y distribuidos.

## OBJETIVOS

### OBJETIVO GENERAL

1. Familiarizar al estudiante con la estructura y funciones características de un sistema operativo de manera que identifique sus componentes principales, cómo han sido implementados y la manera en que estos afectan las aplicaciones y su ambiente de ejecución.

### OBJETIVOS ESPECÍFICOS

1. Estudiar los diferentes componentes de la estructura de un sistema operativo.
2. Conocer la máquina abstracta que proporcionan los sistemas operativos a través de los servicios que ofrecen.
3. Formular en virtud de las características, las repercusiones que tiene el sistema operativo sobre las aplicaciones que en él se ejecuten o desarrollen.
4. Comprender diferentes mecanismos y algoritmos para implementar los diferentes componentes de un sistema operativo.
5. Estudiar las características básicas de varios sistemas operativos comerciales populares.

6. Analizar los diferentes detalles de implementación de sistemas operativos distribuidos.

# CONTENIDOS

## 1. Introducción

- 1.1. ¿Qué es un Sistema Operativo?
- 1.2. Breve Repaso Histórico.
  - 1.2.1. Perspectiva del Usuario.
  - 1.2.2. Perspectiva del Sistema.
- 1.3. Servicios de un Sistema Operativo
- 1.4. Funciones de los Sistemas Operativos
- 1.5. Desarrollo de los Sistemas Operativos
- 1.6. Tipos de Sistemas Operativos.
- 1.7. Estructura de los Sistemas Operativos.

## 2. Administración de CPU y procesos

- 2.1. Procesos
  - 2.1.1. La problemática generada por la interacción de los procesos y sus posibles soluciones
  - 2.1.2. El concepto de proceso
  - 2.1.3. Creación de procesos
  - 2.1.4. Interacción de procesos
  - 2.1.5. Sincronización
  - 2.1.6. Semáforos
  - 2.1.7. Otros métodos de Sincronización
- 2.2. Comunicación y Sincronización
  - 2.2.1. El problema del Deadlock
  - 2.2.2. El problema "económico" de los recursos y sus implicaciones.
- 2.3. Planificación de procesos
  - 2.3.1. Procesos y Recursos
  - 2.3.2. Caracterización del problema del deadlock
  - 2.3.3. Prevención, detección y recuperación de deadlocks
  - 2.3.4. Control de Procesos y Administración del Procesador
- 2.4. Problemas clásicos de comunicación
  - 2.4.1. Análisis de las políticas de asignación del procesador
  - 2.4.2. Estructuras de datos para la administración de procesos.
  - 2.4.3. Esquemas simples de asignación del procesador.
  - 2.4.4. Asignación bajo multiprogramación.

## 3. Administración de la memoria

- 3.1. Introducción
  - 3.1.1. Análisis de los diferentes esquemas y políticas de asignación de la memoria
- 3.2. Memoria sin intercambio
- 3.3. Memoria con intercambio
- 3.4. Memoria virtual
  - 3.4.1. Esquemas simples de administración.
  - 3.4.2. Memoria Virtual.

- 3.4.3.Algoritmos de Pila.
- 3.4.4.Algoritmos de Reloj.

#### **4. Sistemas de archivos**

- 4.1.Introducción
  - 4.1.1.Administración de Información.
  - 4.1.2.El sistema de archivos y otros elementos importantes en la administración de información
  - 4.1.3.Concepto de Archivo.
  - 4.1.4.Métodos de Acceso.
  - 4.1.5.Métodos de Asignación.
  - 4.1.6.Estructuras de Directorios.
  - 4.1.7.Protección de Archivos
  - 4.1.8.Compresión de Datos.
- 4.2.Archivos y directorios
- 4.3.Estructura del almacenamiento secundario
- 4.4.Implementación

#### **5. Entrada y salida**

- 5.1.Fundamentos de entrada y salida
  - 5.1.1.El papel preponderante de las interrupciones en la comunicación con los dispositivos y los Device Drivers.
- 5.2.Hardware de entrada y salida
  - 5.2.1.Comunicación CPU-Dispositivo
  - 5.2.2.Interrupciones
  - 5.2.3.Acceso Directo a Memoria (DMA)
  - 5.2.4.Canales de entrada y salida
  - 5.2.5.Device Drivers
  - 5.2.6.Asignación de Disco

#### **6. Bloqueos**

- 6.1.Detección, evasión y prevención

#### **7. Introducción a los sistemas distribuidos**

- 7.1.Definición de sistema distribuido y un sistema paralelo
  - 7.1.1.Características (ambos)
  - 7.1.2.Arquitectura (ambos)
- 7.2.Interconexión de procesos
  - 7.2.1.Interfaz de sockets
  - 7.2.2.Concurrencia en los servicios
- 7.3.Invocación remota
  - 7.3.1.Aspectos generales
  - 7.3.2.Llamadas a procedimientos remotos aplicadas a un caso de uso.
  - 7.3.3.Otras implementaciones de la invocación remota
- 7.4.Sincronización
  - 7.4.1.Problemática
  - 7.4.2.Sincronización de relojes lógicos
  - 7.4.3.Sincronización de relojes físicos
- 7.5.Otros aspectos de los sistemas operativos distribuidos
  - 7.5.1.Seguridad
  - 7.5.2.Sistemas de nombrado
  - 7.5.3.Memoria compartida distribuida

## **8. Casos de estudio**

- 8.1.PVM (Paralel Virtual Machine) (Investigación)
- 8.2.Estudio comparativo de sistemas operativos vigentes para cliente y para servidor (Investigación)
- 8.3.Sistemas de archivos utilizados en la actualidad, características, ventajas y desventajas (Investigación)
- 8.4.Rendimiento de los diferentes sistemas operativos (Investigación)
- 8.5.Virtualización (Investigación)
- 8.6.Otros temas relacionados

# METODOLOGÍA

La asignatura se impartirá durante 8 horas semanales. 4 Horas de corte teórico y 4 horas de aplicación práctica.

El profesor utilizará fundamentalmente la tradicional pizarra para exponer sus explicaciones y razonamientos. También se utilizará en clase el proyector multimedia y computadora, que resulta muy útil para la introducción de conceptos o la explicación de algoritmos.

En las clases prácticas se realizarán diferentes ejercicios de programación y otras prácticas relacionadas con los temas del curso.

Durante las clases teóricas se podrán realizar pruebas cortas o quices sin previo aviso. Dichas pruebas podrán ser de comprobación de tareas cortas, tareas programadas, laboratorios, exposiciones, comprobaciones de lectura, comprobaciones de la materia vista durante la clase o en clases anteriores.

Los laboratorios consisten en prácticas que deben ser realizadas durante la clase de laboratorio y entregados al final de la misma, a no ser que el profesor indique una fecha posterior para entregarlo. Los laboratorios también pueden ser realizados sin previo aviso y pueden ser de práctica sobre algún tema visto en una clase anterior o sobre alguna investigación corta relacionada con algún tema del curso.

El valor de cada quiz y laboratorio depende de la cantidad de pruebas que se realicen durante el curso.

Las tareas consisten en asignaciones basadas en la materia del curso, pueden ser de investigación. El profesor indicará un día y hora máxima de entrega. El valor de cada tarea estará indicado en el enunciado de la misma.

La investigación será sobre algún tema asignado por el profesor, se realizará en grupos y deberá exponerse al grupo en la fecha que el profesor indique. Estos temas de investigación formarán parte de la materia a evaluar en el curso.

El proyecto consistirá en dar solución a un problema que abarque varios de los temas vistos en el curso y requerirá investigación de otros temas por parte de los estudiantes.

En cualquier evaluación que consista en la entrega de un programa se debe entregar el código fuente completo en el formato adecuado para poder ser revisado, compilado

y ejecutado. Si un programa tiene errores de compilación o algún otro tipo de error que impida su ejecución será calificado con nota cero.

Los exámenes pueden ser pruebas teóricas o teórico-prácticas (esto queda a criterio del profesor). Los exámenes serán realizados en el aula o en la clase de laboratorio en caso de tener una parte práctica.

Sólo se reponen evaluaciones según lo estipulado en el artículo 24 del Reglamento de Régimen Académico de la UCR y utilizando el procedimiento allí descrito.

## EVALUACIÓN

Rubro	Porcentaje
2 Exámenes (15% C/U)	30
Quices	25
Laboratorios	10
Proyecto	15
Investigación	10
Tareas	10

### Bibliografía

- A. Silberschatz, P. Galvin,  
*Sistemas Operativos*, 5ta. Ed.  
Pearson, 1999.
- A.S. Tanenbaum,  
*Sistemas Operativos Modernos*, 2da. Ed.  
Prentice Hall, 2003.
- A.S. Tanenbaum,  
*Sistemas Operativos Distribuidos*.  
Prentice Hall, 1996.
- Bach. Maurice.  
*The Design of the Unix Operation System*.  
Prentice-Hall. Englewood Cliffs. EEUU. 1990.
- George Coulouris, Jean Dollimore y Tim Kindberg.  
*Sistemas Distribuidos. Conceptos y Diseño*.  
Addison-Wesley, 2001, 3ª edición.
- J. Carretero, P. de Miguel, F. García, F. Pérez,  
*Sistemas Operativos. Una Visión Aplicada*,  
McGraw-Hill, 2001.

- H.M. Deitel,  
*Introducción a los Sistemas Operativos*,  
Addison-Wesley, 1987.
- W. Stallings,  
*Sistemas Operativos*,  
Prentice Hall, 2001