



UNIVERSIDAD DE LAS PALMAS  
DE GRAN CANARIA

PROYECTO DOCENTE CURSO: 2005/06

12687 - SISTEMAS OPERATIVOS

**ASIGNATURA:** 12687 - SISTEMAS OPERATIVOS

**CENTRO:** Escuela de Ingeniería Informática

**TITULACIÓN:** Ingeniero en Informática

**DEPARTAMENTO:** INFORMÁTICA Y SISTEMAS

**ÁREA:** Ciencia De La Comp. E Intel. Artificial

**PLAN:** 10 - Año 199 **ESPECIALIDAD:**

**CURSO:** Segundo curso **IMPARTIDA:** Segundo semestre **TIPO:** Troncal

**CRÉDITOS:** 7,5

**TEÓRICOS:** 4,5

**PRÁCTICOS:** 3

## Descriptores B.O.E.

Organización, Estructura y Servicio de los Sistemas Operativos. Gestión y Administración de Memoria y Procesos. Gestión de Entrada/Salida. Sistemas de Ficheros.

## Temario

1. Conceptos sobre sistemas operativos (3h) [3]
  - 1.1.- Funciones y objetivos de los sistemas operativos
  - 1.2.- Evolución de los sistemas operativos
2. Estructura del computador (2h) [3]
  - 2.1.- Estructura de la Entrada/Salida
  - 2.2.- Interrupciones
  - 2.3.- Jerarquía de memorias
  - 2.4.- Protección del hardware
3. Estructura del sistema operativo (2h) [3]
  - 3.1.- Componentes del sistema operativo
  - 3.2.- Servicios del sistema operativo
  - 3.3.- Llamadas al sistema
  - 3.4.- Programas del sistema
  - 3.5.- Modelos arquitectónicos de sistemas operativos
  - 3.6.- Diseño e implementación de sistemas
4. Procesos (2h) [3]
  - 4.1.- Concepto de proceso
  - 4.2.- Planificación de procesos
  - 4.3.- Operaciones sobre procesos
  - 4.4.- Comunicación entre procesos
  - 4.5.- Procesos vs Hilos
5. Planificación de procesos (3h) [3]
  - 5.1.- Criterios de planificación
  - 5.2.- Algoritmos de planificación
  - 5.3.- Evaluación de algoritmos
6. Concurrencia: exclusión mutua y sincronización (10h) [3]
  - 6.1.- Conceptos básicos
  - 6.2.- Problema de la sección crítica: soluciones por software y hardware

- 6.3.- Semáforos
- 6.4.- Regiones Críticas
- 6.5.- Monitores
- 7. Memoria (3h) [3]
  - 8.1.- Antecedentes
  - 8.2.- Sistemas básicos de gestión de memoria
  - 8.3.- Segmentación
  - 8.4.- Paginación
  - 8.5.- Sistemas mixtos
- 8. Memoria Virtual (6h) [3]
  - 9.1.-Principios de operación
  - 9.2.-Paginación por demanda
  - 9.3.-Reemplazo de páginas
  - 9.4.-Asignación de marcos
  - 9.5.-Hiperpaginación
- 9. Archivos (6h) [3]
  - 10.1.- Concepto de archivo
  - 10.2.- Métodos de acceso
  - 10.3.- Organización de directorios
  - 10.4.- Protección
  - 10.5.- Gestión del almacenamiento secundario
- 10. Entrada/salida (5h) [3]
  - 11.1.- Dispositivos de entrada/salida
  - 11.2.- Organización de las funciones de entrada/salida
  - 11.3.- Planificación de discos
- 11. Interbloqueo (3h) [3]
  - 7.1.- Caracterización del interbloqueo
  - 7.2.- Estrategias de tratamiento del interbloqueo
  - 7.3.- Métodos de prevención
  - 7.4.- Métodos de evitación
  - 7.5.- Detección de interbloqueos
  - 7.6.- Recuperación de interbloqueos

## Conocimientos Previos a Valorar

- Organización de computadores
- Programación estructurada y modular. Tipos abstractos de datos.
- Lenguaje C, a nivel principiante-intermedio.
- Inglés técnico: comprensión lectora de documentación técnica.

NOTA importante: si no se conoce previamente el lenguaje C, el estudiante tendrá serias dificultades para realizar los trabajos prácticos.

## Objetivos

Objetivos teóricos: que el estudiante conozca la estructura general, funciones y principios de diseño de los sistemas operativos.

Objetivos prácticos: que el estudiante consolide sus conocimientos teóricos mediante la programación de componentes de un sistema operativo.

## Metodología de la Asignatura

En las clases teóricas se explicarán los fundamentos sobre sistemas operativos descritos en el temario. La teoría se impartirá mediante clases magistrales basadas en el texto de referencia de la asignatura. El estudiante dispondrá de fichas o guías de cada unidad temática para que conozca los puntos más importantes de ella. Además se propondrán ejercicios para realizar en casa, con el fin de que los estudiantes repasen o profundicen en estos contenidos.

Los alumnos aplicarán los conceptos más importantes por medio de la implementación de componentes en un sistema operativo.

## Evaluación

Exámenes teóricos (75% de la calificación)

Un examen de convocatoria y un examen parcial. El examen parcial se realizará en la mitad del cuatrimestre y abarcará la mitad del temario teórico. Dicho parcial supondrá el 50% de la nota de la parte teórica y su nota se guardará únicamente hasta la convocatoria de JUNIO. Aquellos que hayan superado este parcial solo deberán examinarse de la segunda parte en el examen final de la convocatoria de Junio. El examen final podrá contener preguntas sobre el sistema Nachos empleado en prácticas. Para superar la parte teórica de la asignatura es necesario superar ambas partes por separado, lo que es aplicable también a las prácticas de la asignatura.

- Nueve trabajos prácticos obligatorios (25% de la calificación, puntuando todos lo mismo)

Para superar la parte práctica es necesario obtener más del 50% de la nota máxima en cada uno de los dos trabajos prácticos obligatorios.

- Trabajo práctico optativo (10% de puntuación adicional)

## Descripción de las Prácticas

Práctica número 1

Descripción: Introducción al sistema operativo Nachos

Objetivos: Presentar la arquitectura del s.o. instruccional Nachos, así como las instrucciones básicas para instalarlo, explorarlo y modificarlo. También se explicaran aquellas habilidades de operatoria en entorno Linux que son necesarias para desenvolverse correctamente en las prácticas. Por último, basándonos en el propio código fuentes del Nachos se explicarán los aspectos básicos del C++ utilizados en el código fuente del Nachos.

Material de Laboratorio recomendado: Por cada persona, un ordenador con Linux conectado a la red y que tenga instalados los paquetes de desarrollo en C/C++ y un entorno gráfico

Nº horas estimadas en laboratorio: 4

Práctica número 2:

Descripción: Implementación del gestor de llamadas al sistema del Nachos (primer trabajo obligatorio)

Objetivos: Desarrollar el gestor de llamadas al sistema del Nachos. Para ello los alumnos deberán conocer la arquitectura de registros y memoria de la máquina hardware que el Nachos simula, así como las funciones que nos suministra para manejar estos recursos.

Material de Laboratorio recomendado: Por cada 2 personas, un ordenador con Linux conectado a la red y que tenga instalados los paquetes de desarrollo en C/C++ y un entorno gráfico.

Nº horas estimadas en laboratorio: 3

Práctica número 3

Descripción: manejo de hilos y concurrencia en Nachos (segundo trabajo obligatorio)

Objetivos: Desarrollar distintos programas basados en la ejecución concurrente de hilos con la

finalidad de adiestrarse en programación concurrente.

Material de laboratorio recomendado: Por cada dos personas, un pecé con Linux conectado a la red y que tenga instalados los paquetes de desarrollo en C/C++ y un entorno gráfico

Nº horas estimadas en Laboratorio: 3

#### Práctica número 4

Descripción: políticas de planificación de hilos en Nachos (tercer trabajo obligatorio)

Objetivos: Comprobar el funcionamiento de al menos dos políticas de planificación del procesador, implementando parte de una de ellas, vistas en la teoría. Para ello se utilizarán los programas desarrollados en el trabajo anterior.

Material de laboratorio recomendado: Por cada dos personas, un pecé con Linux conectado a la red y que tenga instalados los paquetes de desarrollo en C/C++ y un entorno gráfico

Nº horas estimadas en Laboratorio: 3

#### Práctica número 5

Descripción: Implementación de las variables tipo cerradura (cuarto trabajo obligatorio)

Objetivos: Implementar este mecanismo de sincronización utilizando los recursos que Nachos proporciona.

Material de laboratorio recomendado: Por cada dos personas, un pecé con Linux conectado a la red y que tenga instalados los paquetes de desarrollo en C/C++ y un entorno gráfico

Nº horas estimadas en Laboratorio: 2

#### Práctica número 6

Descripción: Implementación de las variables tipo condición (quinto trabajo obligatorio)

Objetivos: Implementar este mecanismo de sincronización utilizando los recursos que Nachos proporciona.

Material de laboratorio recomendado: Por cada dos personas, un pecé con Linux conectado a la red y que tenga instalados los paquetes de desarrollo en C/C++ y un entorno gráfico

Nº horas estimadas en Laboratorio: 3

#### Práctica número 7

Descripción: resolución de dos problemas clásicos de concurrencia (sexto trabajo obligatorio)

Objetivo: Resolver dos problemas clásicos de concurrencia vistos en teoría, utilizando los mecanismos de sincronización construidos en las dos prácticas anteriores.

Material de laboratorio recomendado: Por cada dos personas, un pecé con Linux conectado a la red y que tenga instalados los paquetes de desarrollo en C/C++ y un entorno gráfico

Nº horas estimadas en Laboratorio: 3

#### Práctica número 8

Descripción: Implementar la llamada al sistema Exec sin multiprogramación (séptimo trabajo obligatorio)

Objetivo: Desarrollar la llamada al sistema Exec, de forma que el programa cargado mediante esta llamada al sistema sustituye al programa que la invoca. El alumno deberá entender la organización de los programas en memoria del Nachos, para posteriormente poder desarrollar la carga de éstos en memoria.

Material de laboratorio recomendado: Por cada dos personas, un pecé con Linux conectado a la red y que tenga instalados los paquetes de desarrollo en C/C++ y un entorno gráfico

Nº horas estimadas en Laboratorio: 3

#### Práctica número 9

Descripción: Implementar la llamada al sistema Exec para que Nachos proporcione un entorno multiprogramado (octavo trabajo obligatorio)

Objetivo: Desarrollar la llamada al sistema Exec de forma que el programa cargado mediante esta

llamada al sistema comparta la memoria con el programa que la invoca. El alumno deberá modificar el procedimiento de carga de programas en memoria para posibilitar que ésta sea compartida por varios programas. También deberá modificar la implementación de la conmutación de contexto entre programas.

Material de laboratorio recomendado: Por cada dos personas, un pecé con Linux conectado a la red y que tenga instalados los paquetes de desarrollo en C/C++ y un entorno gráfico

Nº horas estimadas en Laboratorio: 3

#### Práctica número 10

Descripción: Implementar las llamadas al sistema Exit y Join para el Nachos (noveno trabajo obligatorio)

Objetivo: Desarrollar las llamadas al sistema Join y Exec para completar el entorno multiprogramado de Nachos. El alumno deberá incorporar nuevas estructuras de control con el objeto de implementar estas llamadas de manejo de procesos de usuario.

Material de laboratorio recomendado: Por cada dos personas, un pecé con Linux conectado a la red y que tenga instalados los paquetes de desarrollo en C/C++ y un entorno gráfico

Nº horas estimadas en Laboratorio: 3

#### Práctica número 11 (opcional)

Descripción: memoria virtual (trabajo opcional)

Objetivos: Incorporar memoria virtual al Nachos y evaluar el rendimiento de las políticas de reemplazo de páginas.

Material de laboratorio recomendado: Por cada dos personas, un pecé con Linux conectado a la red y que tenga instalados los paquetes de desarrollo en C/C++ y un entorno gráfico

Nº horas total estimadas para la realización de la práctica: 10

## Bibliografía

---

### [1] Sistemas operativos /

*Abraham Silberschatz, Peter Galvin, Greg Gagne.*

*Limusa,, México : (2002) - (6ª ed.)*

*968-18-6168-X*

---

### [2] El lenguaje de programación C++ /

*Bjarne Stroustrup.*

*Addison Wesley,, Madrid : (2001) - (Edición especial.)*

*847829046X*

---

### [3] Edición especial Linux 4ª edición /

*Jack Tackett, Steven Burnett.*

*Prentice Hall,, Madrid : (2000)*

*8483221675*

---

### [4] Libro de prácticas de sistemas operativos /

*Jesús Carretero Pérez, Félix García Carballeira ; Fernando Pérez Costoya.*

*McGraw-Hill,, Madrid : (2002)*

*84-481-3662-4*

---

### [5] Sistemas operativos: una visión aplicada /

*Jesús Carretero Pérez, Félix García Carballeira, Pedro de Miguel Anasagasti, Fernando Pérez Costoya.*

*McGraw Hill,, Madrid : (2001)*

*84-481-3001-4*

---

**[6] Sistemas operativos: Teoría y problemas /**

*Joaquín Aranda Almansa... [et al.].*

*Sanz y Torres,, Madrid : (2002)*

*84-88667-81-7*

---

**[7] Programación concurrente /**

*José Tomás Palma Méndez, María del Carmen Garrido Carrera, Fernando Sánchez Figueroa, Alexis Quesada Arencibia.*

*Thomson,, Australia ; [Madrid] : (2003)*

*8497321847*

---

**[8] Sistemas operativos: principios de diseño e interioridades /**

*William Stallings ; traducción Amalia Oñate Gómez, Ángel González del Alba Baraja.*

*Prentice Hall,, Madrid : (2001) - (4ª ed.)*

*84-205-3177-4*

---

## Equipo Docente

**CARMELO RUBÉN GARCÍA RODRÍGUEZ**

(COORDINADOR)

**Categoría:** *CATEDRÁTICO DE ESCUELA UNIVERSITARIA*

**Departamento:** *INFORMÁTICA Y SISTEMAS*

**Teléfono:** *928458753* **Correo Electrónico:** *ruben.garcia@ulpgc.es*

**FRANCISCO ALEXIS QUESADA ARENCIBIA**

**Categoría:** *PROFESOR CONTRATADO DOCTOR, TIPO 1*

**Departamento:** *INFORMÁTICA Y SISTEMAS*

**Teléfono:** *928454572* **Correo Electrónico:** *alexis.quesada@ulpgc.es*