

Contenido

Semana 1 – Introducción a Sistemas Operativos.....	7
📌 Objetivo de la Semana	7
📁 Contenido del Directorio.....	7
🧠 Temas cubiertos.....	8
📌 Introducción	8
⚙️ Arquitectura de un sistema operativo	8
📌 Gestión de recursos	8
🛠 Material de práctica.....	8
📖 Cómo usar este material.....	8
📌 Licencia y Atribución.....	9
👨‍💻 Recursos adicionales	9
Semana 2 – Gestión de Procesos en Sistemas Operativos.....	10
📌 Objetivo de la Semana	10
📁 Contenido del Directorio.....	10
🧠 Temas abordados	11
📌 ¿Qué es un proceso?.....	11
⌚ Estados de un proceso	11
🚚 Cambio de contexto	11
🕒 Planificación de CPU.....	11
📝 Actividades y ejercicios.....	12
🛠 Cómo usar este material	12
📖 Recursos adicionales (tema de procesos)	12
📌 Licencia y Atribución.....	12
Semana 3 – Hilos (Threads) y Concurrencia	13
📌 Objetivo de la Semana	13
📁 Contenido del Directorio.....	13
🧠 Temas Abordados.....	14
🕒 ¿Qué es un Hilo (Thread)?.....	14
⌚ Modelos de Hilos	14
⚠️ Concurrencia.....	14

👉 Sincronización Básica	14
📝 Actividades y Ejercicios Sugeridos.....	15
✖ Cómo Usar Este Material.....	15
📌 Licencia y Atribución.....	15
Semana 4 – Gestión de Memoria en Sistemas Operativos.....	16
📌 Objetivo de la Semana	16
📁 Contenido del Directorio.....	16
🧠 Temas abordados esta semana.....	17
📌 ¿Qué es la gestión de memoria?	17
🧱 Paginación.....	17
📐 Segmentación.....	17
🧠 Memoria virtual.....	17
📝 Actividades y ejercicios sugeridos	17
✖ Cómo usar este material	18
📚 Recursos adicionales	18
📌 Licencia y Atribución.....	18
Semana 5 – Seguridad y Protección en Sistemas Operativos	19
📌 Objetivo de la Semana	19
📁 Contenido del Directorio.....	19
🧠 Temas abordados esta semana.....	20
🔒 Seguridad vs Protección	20
⚠ Amenazas a un sistema operativo.....	20
🔑 Control de acceso.....	20
👤 Autenticación de usuarios.....	20
🛡 Mecanismos de protección en sistemas reales	20
📝 Actividades y ejercicios sugeridos	21
✖ Cómo usar este material	21
📌 Licencia y Atribución.....	21
Semana 6 – Sistemas de Archivos y Entrada/Salida.....	22
📌 Objetivo de la Semana	22
📁 Contenido del Directorio.....	22

🧠 Temas abordados esta semana.....	23
📋 Sistemas de archivos	23
💾 Organización de almacenamiento	23
📤 Entrada/Salida (E/S) en sistemas operativos	23
📊 Gestión de dispositivos	23
📝 Actividades y ejercicios sugeridos	23
✖️ Cómo usar este material	24
📌 Licencia y Atribución.....	24
Semana 7 – Comunicación entre Procesos y Sincronización	25
📌 Objetivo de la Semana	25
📁 Contenido del Directorio.....	25
🧠 Temas abordados esta semana.....	26
⌚ Comunicación entre Procesos (IPC)	26
🧩 Memoria Compartida	26
🤝 Sincronización	26
📝 Actividades y Ejercicios Sugeridos.....	26
✖️ Cómo usar este material	27
📌 Licencia y Atribución.....	27
Semana 8 – Deadlocks y Exclusión Mutua.....	28
📌 Objetivo de la Semana	28
📁 Contenido del Directorio.....	28
🧠 Temas abordados esta semana.....	29
🚫 ¿Qué es un Deadlock?.....	29
📌 Condiciones necesarias para deadlocks	29
🛡️ Prevención y evitación	29
⌚ Detección y recuperación.....	29
🔒 Exclusión Mutua	30
📝 Actividades y ejercicios sugeridos	30
✖️ Cómo usar este material	30
📌 Licencia y Atribución.....	30
Semana 9 – Planificación de Disco y Gestión de Entrada/Salida	31

📌 Objetivo de la Semana	31
📁 Contenido del Directorio	31
🧠 Temas abordados esta semana.....	32
↳ Controladores de dispositivos	32
📅 Planificación de disco	32
📦 Buffering y Caching	32
📝 Actividades y ejercicios sugeridos	33
✖️ Cómo usar este material	33
📌 Licencia y Atribución.....	33
Semana 10 – Administración de Sistemas de Archivos y Comandos de Linux	34
📌 Objetivo de la Semana	34
📁 Contenido del Directorio	34
🧠 Temas abordados esta semana.....	35
💻 Administración de Sistemas de Archivos.....	35
🔒 Permisos y Control de Acceso	35
🐧 Comandos de Linux para Gestión de Archivos.....	35
📝 Actividades y ejercicios sugeridos	36
✖️ ¿Cómo usar este material?	36
📌 Licencia y Atribución.....	36
Semana 11 – Redes, Sockets y Sistemas Distribuidos (o Temas Avanzados de SO).....	37
📌 Objetivo de la Semana	37
📁 Contenido del Directorio	37
🧠 Temas abordados esta semana.....	38
🌐 Redes en el contexto de Sistemas Operativos.....	38
🔌 Programación de Sockets.....	38
💻 Arquitectura Cliente-Servidor	38
🤝 Sistemas Distribuidos	38
📝 Actividades y ejercicios sugeridos	38
✖️ Cómo usar este material	39
📌 Licencia y Atribución.....	39
Semana 12 – Repaso Final y Proyecto Integrador.....	40

📌 Objetivo de la Semana	40
📁 Contenido del Directorio	40
🧠 Temas integrados en esta semana	41
✖ Repaso de conceptos esenciales.....	41
📌 Proyecto integrador (ejemplo de enfoque)	41
📝 Actividades sugeridas esta semana.....	42
📌 Cómo usar este material.....	42
📚 Recursos adicionales	42
📌 Licencia y Atribución.....	42
Semana 13 – Monitoreo y Control de Recursos en Sistemas Operativos.....	43
📌 Objetivo de la Semana	43
📁 Contenido del Directorio	43
🧠 Temas abordados esta semana.....	44
💻 Monitoreo de Procesos.....	44
📊 Gestión de Memoria	44
📦 Control de Entrada/Salida y Discos.....	44
⚙️ Optimización y Priorización de Recursos.....	44
📝 Actividades y ejercicios sugeridos	44
✖ Cómo usar este material	45
📌 Licencia y Atribución.....	45
Semana 14 – Gestión de Entrada/Salida Avanzada y Dispositivos	46
📌 Objetivo de la Semana	46
📁 Contenido del Directorio	46
🧠 Temas abordados esta semana.....	47
🔌 Gestión de Entrada/Salida (E/S) Avanzada	47
📦 Dispositivos de Bloque y de Carácter	47
🧠 Módulos del Núcleo y Drivers	47
💡 `sysfs` , `udev` y Administración Dinámica	48
📝 Actividades y ejercicios sugeridos	48
✖ Cómo usar este material	48
📌 Licencia y Atribución.....	48

Semana 15 – Seguridad, Protección y Gestión de Permisos en el Sistema Operativo	49
📌 Objetivo de la Semana	49
📁 Contenido del Directorio	49
🧠 Temas abordados esta semana.....	50
🔒 Seguridad y Protección en el SO	50
👤 Usuarios, Grupos y Permisos.....	50
💻 ACLs y Políticas Avanzadas	50
🔒 Autenticación y Criptografía Básica	51
🛡 Herramientas de Seguridad en Linux	51
📝 Actividades y ejercicios sugeridos	51
✖ Cómo usar este material	51
📌 Licencia y Atribución.....	52
Semana 16 – Seguridad Avanzada y Técnicas de Protección en Sistemas Operativos	53
📌 Objetivo de la Semana	53
📁 Contenido del Directorio	53
🧠 Temas abordados esta semana.....	54
🔍 Amenazas y Riesgos en Sistemas Operativos	54
🛡 Mecanismos de Protección del SO	54
🔒 Seguridad en Procesos	54
📌 Protección de Memoria	54
📝 Actividades y ejercicios sugeridos	55
✖ Cómo usar este material	55
📌 Licencia y Atribución.....	55

Semana 1 – Introducción a Sistemas Operativos

Este directorio contiene los recursos, materiales y ejercicios correspondientes a la **Semana 1** del curso *Sistemas Operativos 2025*.

Objetivo de la Semana

El objetivo de esta semana es:

- Introducir los conceptos fundamentales sobre sistemas operativos.
- Establecer la base teórica y práctica para el resto del curso.
- Asegurar que todos los estudiantes comprendan los temas introductorios esenciales.

Contenido del Directorio

| Archivo | Descripción |

|-----|-----|

| **01-Introducción.md / PDF** | Explica qué es un sistema operativo, su función y objetivos. |

| **02-Arquitectura.tex / PDF** | Documento con la arquitectura hardware-software y la interacción con el OS. |

| **03-EjerciciosSemana1.md** | Lista de ejercicios y actividades para practicar esta semana. |

| **04-EjemploCódigo/** | Carpeta con ejemplos de aplicación o scripts para ilustrar conceptos básicos. |

| **05-Presentación1.pptx / PDF** | Presentación de la primera clase. |

| ` README.md` | Archivo de descripción (este documento). |

> **Nota:** La estructura puede variar dependiendo de cómo se organice el curso o las plataformas utilizadas para compartir material.

Temas cubiertos

Durante esta semana se abordan los siguientes conceptos principales:

Introducción

- ¿Qué es un sistema operativo?
- Historia y evolución de los OS.
- Roles y funciones principales de un OS.

Arquitectura de un sistema operativo

- Relación entre hardware y software.
- Niveles de abstracción y capas del sistema.

Gestión de recursos

- Procesos, memoria, entrada/salida.
- Gestión de archivos y dispositivos.

Material de práctica

- Ejemplos simples de código o diagramas explicativos asociados a los temas.

Cómo usar este material

1. **Leer** los documentos teóricos en orden.
2. **Revisar** las presentaciones anexas.
3. **Practicar** con los ejercicios recopilados.
4. **Ejecutar y estudiar** los ejemplos de código.
5. Si hay dudas, consultar con el instructor o foro del curso.

Licencia y Atribución

Este material forma parte del repositorio *Sistemas-Operativos-2025* y sigue la licencia general del curso, que permite el uso educativo y la modificación con la debida atribución al autor original.

Recursos adicionales

Para ampliar tus conocimientos en sistemas operativos, puedes consultar otros repositorios y documentación especializada sobre temas como planificación de CPU, concurrencia, memoria virtual, hilos, sincronización y sistemas de archivos.

:contentReference[oaicite:3]{index=3}

Semana 2 – Gestión de Procesos en Sistemas Operativos

Este directorio contiene los recursos, actividades y ejemplos correspondientes a la **Semana 2** del curso *Sistemas Operativos 2025*.

Objetivo de la Semana

El objetivo de esta semana es:

- Introducir el concepto de **proceso** y su gestión dentro de un sistema operativo.
- Entender los **estados de un proceso**, cómo se crea y finaliza.
- Analizar la **planificación de CPU** básica y el **cambio de contexto**.
- Practicar con ejemplos y ejercicios sobre estos temas fundamentales.

Contenido del Directorio

| Archivo / Carpeta | Descripción |

|-----|-----|

| **01-Procesos.md / PDF** | Introducción a procesos: definición, atributos y ciclo de vida. |

| **02-EstadosDeProceso.md** | Diagrama de estados de proceso y transición entre ellos. |

| **03-CambioDeContexto.md** | Explicación del cambio de contexto y su importancia. |

| **04-PlanificacionCPU.md** | Planificación básica de CPU (FCFS, Round Robin, SJF). |

| **05-EjerciciosSemana2.md** | Actividades y ejercicios prácticos de esta semana. |

| **06-CodigoEjemplo/** | Código de ejemplo en C / Shell / pseudocódigo. |

| **07-Presentación2.pdf / PPTX** | Presentación / diapositivas relacionadas con los temas. |

| ` README.md` | **Este documento**, con descripción general. |

>  *La estructura de archivos puede variar dependiendo de la organización del curso.*

Temas abordados

¿Qué es un proceso?

- Concepto y diferencias con un programa.
- Atributos esenciales de un proceso.

Estados de un proceso

- Nuevo
- Listo
- Ejecutando
- Bloqueado / Esperando
- Terminado
- Transiciones entre estados (con diagramas).

Cambio de contexto

- Qué sucede en un cambio de contexto.
- Registro de CPU, pila, memoria, etc.

Planificación de CPU

- Algoritmos sencillos:
 - **FCFS** (First-Come, First-Served)
 - **Round Robin**
 - **SJF** (Shortest Job First)

- Comparación de ventajas y desventajas.

Actividades y ejercicios

Se recomienda:

1. **Leer todos los apuntes teóricos** en el orden propuesto.
2. **Estudiar los diagramas de estados de procesos** y practicar con ejemplos.
3. **Resolver los ejercicios** de ` EjerciciosSemana2.md` .
4. **Probar los ejemplos de código** y modificarlos para ver diferentes escenarios de planificación y cambio de contexto.

Cómo usar este material

1. Abre los documentos en formato Markdown o PDF.
2. Revisa el código de ejemplo en la carpeta ` 06-CodigoEjemplo/` .
3. Completa los ejercicios propuestos.
4. Si existen presentaciones, úsalas como complemento visual de la teoría.

Recursos adicionales (tema de procesos)

Para ampliar tus conocimientos, puedes consultar referencias de gestión de procesos en sistemas operativos clásicos, incluyendo estado de un proceso, manejo de colas y algoritmos de planificación CPU — temas que suelen abordarse en unidades didácticas alineadas con el temario de Sistemas Operativos.

:contentReference[oaicite:0]{index=0}

Licencia y Atribución

Este material forma parte del repositorio *Sistemas-Operativos-2025* y sigue la licencia general del curso, permitiendo su uso educativo con la debida atribución al autor original.

Semana 3 – Hilos (Threads) y Concurrency

Este directorio contiene los recursos, apuntes, ejercicios y ejemplos correspondientes a la **Semana 3** del curso *Sistemas Operativos 2025*.

📣 Objetivo de la Semana

Durante esta semana se busca:

- Introducir el concepto de **hilos (threads)** como unidad de ejecución dentro de un proceso.
- Diferenciar entre hilos y procesos.
- Entender los **beneficios de la concurrencia** y las **problemáticas que surgen** al ejecutar múltiples hilos de forma simultánea.
- Presentar mecanismos básicos de **sincronización** para coordinar acceso a recursos compartidos.

📁 Contenido del Directorio

Archivo / Carpeta	Descripción
----- -----	
01-IntroduccionThreads.md / PDF	Conceptos fundamentales de hilos y diferencias con procesos.
02-ModelosDeHilos.md / PDF	Modelos de hilos (1:1, N:1, M:N) y su implementación.
03-Concurrencia.md / PDF	Concurrencia, condiciones de carrera y problemas clásicos.
04-SincronizacionBasica.md	Semáforos, mutexes y exclusión mutua.
05-EjerciciosSemana3.md	Ejercicios prácticos para aplicar los conceptos de hilos y concurrencia.
06-EjemplosCodigo/	Ejemplos de código en C / pseudocódigo que ilustran hilos y sincronización.
07-Presentacion3.pdf / PPTX	Presentación visual con diagramas y ejemplos clave.
`README.md`	Este documento — descripción general de la semana.

Temas Abordados

¿Qué es un Hilo (Thread)?

Un **hilo** es una unidad de ejecución **más ligera que un proceso**, que comparte contexto de memoria y recursos con otros hilos del mismo proceso, permitiendo **conurrencia más eficiente** dentro de un mismo programa.
:contentReference[oaicite:2]{index=2}

Modelos de Hilos

Los sistemas operativos pueden implementar hilos de varias formas:

- **1:1** — Cada hilo de usuario corresponde a un hilo del kernel.
- **N:1** — Varios hilos de usuario se mapean a un solo hilo del kernel.
- **M:N** — Mapeos flexibles entre hilos de usuario y hilos del kernel.

Estos modelos influyen en **desempeño y concurrencia real** de las aplicaciones.

:contentReference[oaicite:3]{index=3}

Concurrencia

La **concurrencia** se refiere a la ejecución de múltiples secuencias de instrucciones (hilos/procesos) que pueden **compartir recursos**. Esto puede generar:

- **Condiciones de carrera** (*race conditions*) cuando dos hilos acceden y modifican un mismo recurso sin coordinación.
 - Resultados **no deterministas** si no hay control de acceso.
- :contentReference[oaicite:4]{index=4}

Sincronización Básica

Para resolver problemas de concurrencia se usan mecanismos de sincronización como:

- **Mutexes** (bloqueos exclusivos).
- **Semáforos** (contadores que regulan acceso).
- **Barreras** y otros mecanismos de coordinación.

Estos métodos garantizan que sólo un hilo acceda a una **sección crítica** a la vez.
:contentReference[oaicite:5]{index=5}

Actividades y Ejercicios Sugeridos

1. **Leer los apuntes teóricos** para comprender qué son hilos y por qué se usan.
2. **Ejecutar los ejemplos de código** para ver cómo se crean y sincronizan hilos.
3. **Resolver los ejercicios de la semana** para practicar condiciones de carrera y exclusión mutua.
4. **Comparar resultados** obtenidos con y sin sincronización adecuada.

Cómo Usar Este Material

1. Abre los documentos ` .md` o ` .pdf` en orden para estudiar cada concepto.
2. Usa la presentación visual para reforzar lo principal.
3. Practica con los ejemplos de código sobre un sistema real o en máquina virtual.
4. Completa los ejercicios y guarda tu documentación.

Licencia y Atribución

Este material forma parte del repositorio *Sistemas-Operativos-2025* y se distribuye bajo la licencia educativa correspondiente, **permitiendo su uso con atribución al autor original**.

Semana 4 – Gestión de Memoria en Sistemas Operativos

Este directorio contiene los recursos, ejercicios y ejemplos correspondientes a la **Semana 4** del curso *Sistemas Operativos 2025*.

Objetivo de la Semana

El objetivo de esta semana es:

- Entender los **conceptos fundamentales de la gestión de memoria** en un sistema operativo.
- Analizar **técnicas y estructuras** como memoria contigua, paginación y segmentación.
- Conocer mecanismos como la memoria virtual y su impacto en rendimiento.
- Practicar con ejercicios y ejemplos relacionados a la administración de memoria en sistemas reales.

Contenido del Directorio

| Archivo / Carpeta | Descripción |

|-----|-----|

| **01-Memoria-Introducción.md / PDF** | Introducción a la gestión de memoria y objetivos principales. |

| **02-Paginación.md / PDF** | Documento que cubre la paginación simple, paginación con tablas y TLB. |

| **03-Segmentación.md / PDF** | Explica la segmentación, ventajas, desventajas y aplicación. |

| **04-Memoria-Virtual.md / PDF** | Conceptos de memoria virtual, fallos de página y swapping. |

| **05-EjerciciosSemana4.md** | Lista de ejercicios prácticos y problemas teóricos. |

| **06-EjemplosCodigo/** | Código de ejemplo (C / Shell / pseudocódigo) relacionado con administración de memoria. |

| **07-Presentación4.pdf / PPTX** | Presentación visual con diagramas y explicaciones clave. |

| ` README.md` | **Este documento**, con descripción general. |

>  *La estructura de archivos puede variar dependiendo de cómo se haya organizado el contenido del curso.*

Temas abordados esta semana

¿Qué es la gestión de memoria?

- Función de la memoria principal del sistema.
- Diferencia entre memoria física y lógica.
- Importancia de administrar memoria de forma eficiente.

Paginación

- División de memoria en páginas y marcos de página.
- Tablas de páginas y acceso a dirección física.
- TLB (Translation Lookaside Buffer).

Segmentación

- División en segmentos lógicos (código, datos, pila).
- Comparación con paginación.
- Ventajas y limitaciones de la segmentación.

Memoria virtual

- Concepto y propósito de la memoria virtual.
- Fallos de página (*page faults*) y manejo.
- Swapping y políticas de reemplazo.

Actividades y ejercicios sugeridos

1. **Leer todos los apuntes teóricos** siguiendo los documentos numerados.
2. **Resolver los ejercicios** propuestos en `05-EjerciciosSemana4.md` .

3. **Revisar y ejecutar** los ejemplos de código, si los hay, para ver el impacto de diferentes esquemas de memoria.

4. **Comparar técnicas** (paginación vs segmentación) y describir casos de uso reales.

Cómo usar este material

1. Explora los documentos en el orden sugerido.
2. Usa las presentaciones como apoyo visual.
3. Completa y revisa tus respuestas a los ejercicios.
4. Para código ejemplos, compílalos si aplica (p.ej., con GCC) y analiza su salida para comprender mejor cada mecanismo de memoria.

Recursos adicionales

Para profundizar en gestión de memoria y sistemas operativos, hay muchos recursos didácticos disponibles, incluyendo apuntes y ejemplos de instituciones académicas y repositorios comunitarios que cubren estos temas fundamentales.

:contentReference[oaicite:1]{index=1}

Licencia y Atribución

Este material forma parte del repositorio *Sistemas-Operativos-2025* y sigue la licencia general del curso, permitiendo el uso educativo con la debida atribución al autor original.

Semana 5 – Seguridad y Protección en Sistemas Operativos

Este directorio contiene los recursos, ejercicios y ejemplos correspondientes a la **Semana 5** del curso *Sistemas Operativos 2025*.

Objetivo de la Semana

El objetivo de esta semana es:

- Introducir los conceptos fundamentales sobre **seguridad y protección** dentro de un sistema operativo. :contentReference[oaicite:1]{index=1}
- Entender modelos de amenazas, mecanismos de protección y políticas de acceso.
- Presentar técnicas de **control de acceso**, autenticación y protección de recursos.
- Practicar con ejercicios y ejemplos concretos sobre mecanismos de seguridad del sistema.

Contenido del Directorio

| Archivo / Carpeta | Descripción |

|-----|-----|

| **01-Seguridad-Protección.md / PDF** | Documento teórico sobre conceptos de seguridad y protección. :contentReference[oaicite:2]{index=2} |

| **02-AmenazasSistemas.md / PDF** | Tipos de amenazas a sistemas operativos y vectores de ataque. :contentReference[oaicite:3]{index=3} |

| **03-ControlAcceso.md** | Explicación de mecanismos de control de acceso y permisos. :contentReference[oaicite:4]{index=4} |

| **04-AutenticaciónUsuarios.md** | Autenticación de usuarios y gestión de credenciales. :contentReference[oaicite:5]{index=5} |

| **05-EjerciciosSemana5.md** | Lista de ejercicios prácticos sobre seguridad y protección. |

| **06-EjemplosCodigo/** | Ejemplos de código (scripts o programas) relacionados con autenticación o control de accesos. |

| **07-Presentación5.pdf** | Presentación visual con diagramas y explicaciones clave. |

| ` README.md` | **Este documento**, con descripción general. |

>  *La estructura de archivos puede variar dependiendo de cómo esté realmente organizado el repositorio.*

Temas abordados esta semana

Seguridad vs Protección

- **Seguridad**: conjunto de técnicas para defender un sistema contra accesos no autorizados.
- **Protección**: mecanismos específicos para garantizar que los recursos se usan de forma segura. :contentReference[oaicite:6]{index=6}

Amenazas a un sistema operativo

- Accesos no autorizados.
- Malware y exploits.
- Fallas de diseño o errores de configuración. :contentReference[oaicite:7]{index=7}

Control de acceso

- Gestión de permisos sobre recursos (lectura, escritura, ejecución).
- Listas de control de acceso (*ACLs*) y modelos de seguridad.
:contentReference[oaicite:8]{index=8}

Autenticación de usuarios

- Uso de credenciales y métodos de verificación.
- Autenticación de múltiples factores o basada en tokens.
:contentReference[oaicite:9]{index=9}

Mecanismos de protección en sistemas reales

- Políticas vs mecanismos.
- Ejemplos en UNIX / Linux (permisos de archivos, roles, etc.).
:contentReference[oaicite:10]{index=10}

Actividades y ejercicios sugeridos

1. **Leer los apuntes teóricos** de seguridad y protección.
2. **Analizar ejemplos de amenazas** y proponer técnicas de mitigación.
3. **Resolver los ejercicios** de `05-EjerciciosSemana5.md` .
4. **Ejecutar ejemplos de código** para comprobar permisos y control de acceso en un sistema real.

Cómo usar este material

1. Explora cada documento en el orden sugerido.
2. Usa las presentaciones como apoyo visual.
3. Completa y revisa tus respuestas a los ejercicios.
4. Ejecuta los ejemplos de código en un entorno seguro (p.ej., una máquina virtual Linux) para observar cómo funcionan los mecanismos de seguridad.

Licencia y Atribución

Este material forma parte del repositorio *Sistemas-Operativos-2025* y sigue la licencia general del curso, permitiendo **su uso educativo con la debida atribución al autor original**.

Semana 6 – Sistemas de Archivos y Entrada/Salida

Este directorio contiene los recursos, materiales y ejercicios correspondientes a la **Semana 6** del curso *Sistemas Operativos 2025*.

Objetivo de la Semana

El objetivo de esta semana es:

- Entender los conceptos fundamentales de **sistemas de archivos** dentro de un sistema operativo.
- Conocer los **mecanismos de entrada/salida (E/S)** y cómo el SO gestiona dispositivos y operaciones de E/S.
- Revisar estructuras de almacenamiento, organización y métodos de acceso.
- Practicar con ejercicios, ejemplos y análisis relacionados con sistemas de archivos y operaciones de E/S.

Contenido del Directorio

| Archivo / Carpeta | Descripción |

|-----|-----|

| **01-SistemasDeArchivos.md / PDF** | Introducción a los sistemas de archivos: estructura, componentes y funciones. |

| **02-EntradaSalida.md / PDF** | Explicación de los mecanismos de entrada y salida gestionados por el sistema operativo. |

| **03-OrganizacionDiscos.md** | Organización física y lógica de discos, particiones y tablas de asignación. |

| **04-EstructurasArchivos.md** | Estructuras internas de archivos (inodos, FAT, directorios, metadatos). |

| **05-EjerciciosSemana6.md** | Lista de ejercicios prácticos sobre sistemas de archivos y E/S. |

| **06-EjemplosCodigo/** | Ejemplos de código (C, scripts o pseudocódigo) relacionados con sistemas de archivos y operaciones de E/S. |

| **07-Presentación6.pdf / PPTX** | Presentación visual con diagramas y explicaciones clave. |

| ` README.md` | **Este documento**, con descripción general. |

Temas abordados esta semana

Sistemas de archivos

- **Definición y propósito.**
- Organización de archivos y directorios.
- Tipos comunes de sistemas de archivos (FAT, NTFS, ext4, etc.).
- Metadatos de archivo y permisos.

Organización de almacenamiento

- Discos y particiones.
- Tablas de asignación (FAT), inodos (UNIX).
- Bloques, clusters y acceso secuencial vs aleatorio.

Entrada/Salida (E/S) en sistemas operativos

- **Mecanismos de comunicación** con dispositivos.
- Buffering, caching y controladores de dispositivos.
- Operaciones básicas de lectura/escritura.

Gestión de dispositivos

- Controladores de disco y dispositivos de E/S.
- Políticas de planificación de E/S.
- Ejemplos con dispositivos reales.

Actividades y ejercicios sugeridos

1. **Revisar la teoría** sobre sistemas de archivos y E/S en los documentos provistos.
2. **Ejercicios prácticos** en `05-EjerciciosSemana6.md` .
3. **Revisar los ejemplos de código** en la carpeta `06-EjemplosCodigo/` .
4. **Analizar casos reales** de cómo funciona un sistema de archivos y las operaciones de E/S en un sistema UNIX o Linux (ext4, inodos, syscalls).

5. **Comparar diferentes formatos** de sistemas de archivos y discutir ventajas/inconvenientes.

Cómo usar este material

1. Abre los documentos en formato Markdown o PDF.
2. Explora cada tema en el orden sugerido (de teoría a práctica).
3. Utiliza las presentaciones como apoyo visual.
4. Ejecuta los ejemplos de código en un entorno de desarrollo si es posible (p. ej., Linux).
5. Resuelve los ejercicios y compáralos con las soluciones sugeridas.

Licencia y Atribución

Este material forma parte del repositorio *Sistemas-Operativos-2025* y sigue la licencia general del curso, permitiendo su uso educativo con la debida atribución al autor original.

Semana 7 – Comunicación entre Procesos y Sincronización

Este directorio contiene los recursos, explicaciones, ejercicios y ejemplos correspondientes a la **Semana 7** del curso *Sistemas Operativos 2025*.

Objetivo de la Semana

El objetivo de esta semana es:

- Explorar los mecanismos de **Comunicación entre Procesos (IPC)** y **Sincronización**.
- Entender cómo los procesos pueden intercambiar información y coordinar su ejecución.
- Analizar herramientas como semáforos, mutexes, variables compartidas y barreras.
- Practicar con ejemplos y ejercicios que implementan estas técnicas clave en sistemas concurrentes.

Contenido del Directorio

| Archivo / Carpeta | Descripción |

|-----|-----|

| **01-IPC-Conceptos.md / PDF** | Introducción a la comunicación entre procesos, objetivos y escenarios de uso. |

| **02-Pipes-y-FIFOs.md / PDF** | Comunicación mediante pipes y FIFOs (named pipes). |

| **03-MemoriaCompartida.md / PDF** | Uso de memoria compartida para pasar datos entre procesos. |

| **04-Semaforos-Mutexes.md / PDF** | Sincronización de procesos con semáforos y mutexes. |

| **05-EjerciciosSemana7.md** | Actividades y ejercicios prácticos para aplicar los conceptos. |

| **06-EjemplosCodigo/** | Código de ejemplo (C, scripts o pseudocódigo) implementando IPC y sincronización. |

| **07-Presentación7.pdf / PPTX** | Presentaciones o diagramas clave para esta semana. |

| ` README.md` | **Este documento**, con descripción general. |

>  *La estructura de archivos puede variar dependiendo de cómo se organiza el curso dentro del repositorio.*

Temas abordados esta semana

Comunicación entre Procesos (IPC)

- **Proceso a proceso:** intercambio de datos y mensajes.
- **Pipes (tuberías):** canal simple para transmitir datos entre procesos relacionados.
- **FIFOs (named pipes):** pipes con nombre para procesos no relacionados.
- **Sockets básicos:** concepto de comunicación orientada a red para procesos locales o remotos.

Memoria Compartida

- Área de memoria accesible por múltiples procesos.
- Cómo se establece y controla el acceso seguro.
- Ventajas y riesgos de la memoria compartida.

Sincronización

- Problemas clásicos: carrera crítica (*race conditions*).
- **Semáforos** y **mutexes** para coordinar acceso a recursos compartidos.
- Concepto de **sección crítica** y cómo protegerla.

Actividades y Ejercicios Sugeridos

1. **Leer cuidadosamente** cada uno de los apuntes teóricos.
2. **Estudiar y aplicar** los mecanismos de IPC en ejemplos de código.
3. **Resolver los ejercicios** propuestos en `05-EjerciciosSemana7.md` .

4. **Ejecutar los ejemplos de código** en un entorno real (Linux/Unix) para observar la sincronización y comunicación en acción.
5. **Comparar diferentes técnicas** y analizar ventajas y desventajas de cada una en distintos escenarios.

Cómo usar este material

1. Abre los documentos en formato Markdown o PDF para comprender cada concepto básico.
2. Usa las presentaciones para repasar visualmente los mecanismos de IPC y sincronización.
3. Explora los ejemplos de código en la carpeta `06-EjemplosCodigo/` .
4. Completa todos los ejercicios y revisa tus soluciones.

Licencia y Atribución

Este material forma parte del repositorio *Sistemas-Operativos-2025* y sigue la licencia general del curso, **permitiendo su uso educativo con la debida atribución al autor original**.

Semana 8 – Deadlocks y Exclusión Mutua

Este directorio contiene los recursos, materiales y ejercicios correspondientes a la **Semana 8** del curso *Sistemas Operativos 2025*.

Objetivo de la Semana

El objetivo de esta semana es:

- Entender el fenómeno de **deadlock (bloqueo mutuo)** en sistemas concurrentes.
- Estudiar las **condiciones necesarias** para que ocurra un deadlock.
- Analizar **técnicas de prevención, evitación y recuperación** ante deadlocks.
- Revisar cómo se logra la **exclusión mutua** y por qué es esencial para sincronización segura de procesos.

Contenido del Directorio

Archivo / Carpeta	Descripción
----- -----	
01-Deadlocks-Conceptos.md / PDF	Introducción al concepto de deadlock y causas.
02-CondicionesDeadlock.md	Análisis de las condiciones de Coffman para deadlocks.
03-PrevencionYEvicion.md	Técnicas para prevenir y evitar deadlocks.
04-RecuperacionDeadlock.md	Estrategias para detectar y recuperar sistemas bloqueados.
05-EjerciciosSemana8.md	Actividades y ejercicios prácticos sobre deadlocks.
06-EjemplosCodigo/	Ejemplos de código (C / pseudocódigo) que ilustran deadlocks y soluciones.
07-Presentacion8.pdf / PPTX	Presentación visual con diagramas explicativos de deadlocks y exclusión mutua.
` README.md`	**Este documento**, con descripción general.

>  *La estructura real puede variar dependiendo de cómo esté organizado tu repositorio.*

Temas abordados esta semana

¿Qué es un Deadlock?

Un **deadlock** o bloqueo mutuo ocurre cuando dos o más procesos quedan permanentemente esperando recursos que están siendo retenidos por otros procesos, creando una situación de bloqueo circular.

Condiciones necesarias para deadlocks

Para que ocurra un deadlock se deben cumplir simultáneamente las siguientes condiciones (conocidas como *condiciones de Coffman*):

1. **Exclusión mutua** – al menos un recurso no compatible.
2. **Retención y espera** – procesos que retienen recursos mientras esperan otros.
3. **No apropiación** – los recursos no pueden ser forzados a salir de un proceso.
4. **Espera circular** – existe una cadena circular de procesos esperando recursos.

Entender estas condiciones ayuda a diseñar mecanismos para evitarlos.

Prevención y evitación

- **Prevención:** Impedir que alguna de las condiciones necesarias pueda ocurrir.
- **Evitación:** Técnicas como el *algoritmo del banquero* permiten decidir si conceder o no recursos basados en estados seguros.

Estas técnicas son parte de la teoría clásica de planificación y asignación de recursos.

Detección y recuperación

- **Detección de deadlocks:** Monitoreo del estado de asignación de recursos para encontrar ciclos.
- **Recuperación:** Estrategias para liberar procesos o recuperar recursos (abortos, preemptions).

Exclusión Mutua

La **exclusión mutua** garantiza que una sola entidad (proceso/hilo) acceda a una sección crítica a la vez, evitando condiciones de carrera. Mecanismos comunes incluyen:

- Semáforos
- Mutexes (bloqueos mutuos)
- Monitores

Actividades y ejercicios sugeridos

1. **Leer los apuntes teóricos** en cada documento numerado.
2. **Estudiar los diagramas** y ejemplos de deadlock.
3. **Resolver los ejercicios** en `05-EjerciciosSemana8.md` .
4. **Ejecutar y analizar los ejemplos de código** para ver cómo pueden producirse deadlocks y cómo evitarlos.

Cómo usar este material

1. Abre los documentos en Markdown o PDF para comprender los conceptos básicos.
2. Usa las presentaciones como apoyo visual para repasar.
3. Revisa los ejemplos de código para comprender cómo se manifiestan deadlocks en programas reales.
4. Completa los ejercicios y compara tus soluciones.

Licencia y Atribución

Este material forma parte del repositorio *Sistemas-Operativos-2025* y sigue la licencia general del curso, permitiendo su uso educativo con la debida atribución al autor original.

Semana 9 – Planificación de Disco y Gestión de Entrada/Salida

Este directorio contiene los recursos, apuntes, ejercicios y ejemplos correspondientes a la **Semana 9** del curso *Sistemas Operativos 2025*.

Objetivo de la Semana

El objetivo de esta semana es:

- Profundizar en los mecanismos de **entrada/salida (E/S)** del sistema operativo y la **planificación de dispositivos**.
- Comprender cómo el SO gestiona el acceso a dispositivos de almacenamiento.
- Estudiar algoritmos de **planificación de disco**, **buffering**, **caching** y **controladores de E/S**.
- Practicar con ejercicios y ejemplos que muestren el funcionamiento de estos mecanismos.

Contenido del Directorio

Archivo / Carpeta	Descripción
----- -----	
01-EntradaSalida.md / PDF	Introducción a los mecanismos de E/S del sistema operativo.
02-ControladoresEYS.md / PDF	Explicación del papel de los controladores para dispositivos de E/S.
03-PlanificacionDisco.md	Algoritmos de planificación de disco (FCFS, SSTF, SCAN, C-SCAN).
04-BufferingYCaching.md	Técnicas de buffering y caching para mejorar rendimiento de E/S.
05-EjerciciosSemana9.md	Actividades y ejercicios prácticos relacionados con E/S y planificación de discos.
06-EjemplosCodigo/	Código de ejemplo (C / pseudocódigo / scripts) ilustrando conceptos clave.

| **07-Presentacion9.pdf** | Presentación con diagramas y ejemplos visuales. |

| ` README.md` | **Este documento**, con descripción general. |

>  *La estructura exacta puede variar dependiendo de cómo se haya organizado el repositorio.*

Temas abordados esta semana

 Entrada/Salida (E/S) en sistemas operativos

Los sistemas operativos no solo gestionan programas y memoria, sino también **operaciones de entrada y salida** hacia dispositivos como discos, teclados y redes. El SO actúa como intermediario entre aplicaciones y hardware.

:contentReference[oaicite:1]{index=1}

Controladores de dispositivos

Los **drivers o controladores** son componentes especializados que permiten al sistema operativo comunicarse con el hardware concreto (discos, interfaces USB, etc.). Gestionan interrupciones, transferencia de datos y errores.

Planificación de disco

Los discos y dispositivos de almacenamiento tienen algoritmos específicos para decidir **qué solicitud de E/S atender primero**, para mejorar rendimiento y reducir tiempo de búsqueda:

- **FCFS (First Come First Serve)**
- **SSTF (Shortest Seek Time First)**
- **SCAN / C-SCAN**

Estos algoritmos son especialmente relevantes para dispositivos tradicionales y se estudian con simuladores y cálculos de tiempos de acceso.

:contentReference[oaicite:2]{index=2}

Buffering y Caching

Para mejorar la eficiencia de E/S, los sistemas operativos utilizan técnicas de **buffering** (almacenamiento temporal de datos) y **caching** (almacenamiento de datos frecuentemente usados en memoria más rápida).

Actividades y ejercicios sugeridos

1. **Leer la teoría** de cada documento en orden.
2. **Estudiar ejemplos de planificación de disco** y simular con valores reales de acceso.
3. **Resolver los ejercicios** del archivo `05-EjerciciosSemana9.md` .
4. **Probar los ejemplos de código** para observar cómo se manejan las operaciones de E/S y planificación en un SO.

Cómo usar este material

1. Abre los documentos en formato Markdown o PDF para entender cada concepto básico.
2. Usa las presentaciones como apoyo visual para repasar algoritmos de disco.
3. Explora los ejemplos de código en la carpeta `06-EjemplosCodigo/` .
4. Resuelve los ejercicios y compara tus respuestas para consolidar el aprendizaje.

Licencia y Atribución

Este material forma parte del repositorio *Sistemas-Operativos-2025* y sigue la licencia general del curso, **permitiendo su uso educativo con la debida atribución al autor original**.

Semana 10 – Administración de Sistemas de Archivos y Comandos de Linux

Este directorio contiene los recursos, apuntes, ejercicios y ejemplos correspondientes a la **Semana 10** del curso *Sistemas Operativos 2025*.

Objetivo de la Semana

El objetivo de esta semana es:

- Profundizar en la **administración de sistemas de archivos** dentro de un sistema operativo. :contentReference[oaicite:0]{index=0}
- Explorar comandos y herramientas de Linux para crear, montar, desmontar y gestionar sistemas de archivos. :contentReference[oaicite:1]{index=1}
- Entender conceptos como **permisos, enlaces y cuotas de disco**, así como técnicas para controlar el espacio en disco. :contentReference[oaicite:2]{index=2}

Contenido del Directorio

| Archivo / Carpeta | Descripción |

|-----|-----|

| **01-AdministracionDeFicheros.md / PDF** | Introducción al manejo de sistemas de archivos: creación, estructura y gestión. |

| **02-Montaje-Desmontaje.md / PDF** | Comandos para montar y desmontar sistemas de archivos en Linux. |

| **03-Permisos-Enlaces-Cuotas.md** | Explicación de permisos, enlaces simbólicos/físicos y cuotas de disco. |

| **04-ComandosLinux.md** | Lista y explicación de comandos Linux útiles para administración de archivos. |

| **05-EjerciciosSemana10.md** | Actividades prácticas y ejercicios sobre administración de sistemas de archivos y comandos. |

| **06-EjemplosCodigo/** | Ejemplos de uso de comandos y scripts de administración (bash u otros). |

| **07-Presentacion10.pdf / PPTX** | Presentación visual con ejemplos y comandos paso a paso. |

| ` README.md` | **Este documento**, con una descripción completa de la semana. |

>  *La estructura real puede variar dependiendo de cómo esté organizado el repositorio y qué archivos específicos tenga tu carpeta `Semana10`.*

Temas abordados esta semana

Administración de Sistemas de Archivos

- **¿Qué es un sistema de archivos?**

Sistema que organiza y almacena datos en dispositivos de almacenamiento.

:contentReference[oaicite:3]{index=3}

- **Creación y eliminación de sistemas de archivos**

Cómo formatear y preparar discos o particiones para uso del sistema operativo.

:contentReference[oaicite:4]{index=4}

- **Montaje y desmontaje**

Asociar un sistema de archivos a un punto de montaje y liberar ese punto cuando no se usa. :contentReference[oaicite:5]{index=5}

Permisos y Control de Acceso

- **Permisos de archivos y directorios**

Lectura, escritura y ejecución para usuario, grupo y otros.

:contentReference[oaicite:6]{index=6}

- **Enlaces físicos y simbólicos**

Referencias alternativas a archivos en el sistema de archivos.

:contentReference[oaicite:7]{index=7}

- **Cuotas de disco**

Control de cuánto espacio de almacenamiento pueden usar usuarios o grupos.

:contentReference[oaicite:8]{index=8}

Comandos de Linux para Gestión de Archivos

- `ls`, `cp`, `mv`, `rm`, `chmod`, `chown`, `df`, `du`

Comandos básicos para explorar, copiar, mover, borrar y gestionar atributos.

:contentReference[oaicite:9]{index=9}

Actividades y ejercicios sugeridos

1. **Revisar los apuntes teóricos** sobre sistemas de archivos y comandos de Linux.
2. **Practicar con los ejercicios** propuestos en `05-EjerciciosSemana10.md` .
3. **Ejecutar ejemplos de código y scripts** para afianzar el uso de comandos de administración.
4. **Experimentar en un entorno Linux** (por ejemplo, una máquina virtual) con montaje, permisos y cuotas.

¿Cómo usar este material?

1. Abre los documentos ` .md` o ` .pdf` para entender cada tema.
2. Observa las presentaciones para reforzar los conceptos visualmente.
3. Ejecuta los ejemplos de comandos en un sistema real o simulado.
4. Completa los ejercicios, guarda tus soluciones y compáralas con posibles guías de solución.

Licencia y Atribución

Este material forma parte del repositorio *Sistemas-Operativos-2025* y sigue la licencia general del curso, **permitiendo su uso educativo con la debida atribución al autor original**.

Semana 11 – Redes, Sockets y Sistemas Distribuidos (o Temas Avanzados de SO)

Este directorio contiene los recursos, apuntes, actividades y ejemplos correspondientes a la **Semana 11** del curso *Sistemas Operativos 2025*.

Objetivo de la Semana

El objetivo de esta semana es:

- Introducir los conceptos fundamentales de **comunicaciones de red y programación de sockets** desde la perspectiva de un sistema operativo.
- Analizar cómo los sistemas operativos interactúan con redes y la pila TCP/IP.
- Explorar mecanismos de comunicación entre procesos a través de redes y la coordinación en sistemas distribuidos.
- Practicar con ejemplos de código y ejercicios que demuestren transmisión y recepción de datos en red.

Contenido del Directorio

Archivo / Carpeta	Descripción
----- -----	
01-Redes-Introduccion.md / PDF	Conceptos básicos de redes y cómo los SO gestionan interfaces y comunicaciones.
02-SocketsTCPUDP.md / PDF	Programación de sockets TCP y UDP para comunicación entre procesos en red.
03-ClienteServidor.md / PDF	Patrón cliente-servidor, puertos, direcciones y protocolos comunes.
04-SistemasDistribuidos.md / PDF	Conceptos de sistemas distribuidos y coordinación entre nodos.
05-EjerciciosSemana11.md	Lista de ejercicios prácticos para afianzar los temas.
06-EjemplosCodigo/	Ejemplos de código (C / Python / pseudocódigo) para comunicación en red.

| **07-Presentacion11.pdf / PPTX** | Presentación visual con diagramas, ejemplos y casos de uso. |

| ` README.md` | **Este documento**, con descripción general de la semana. |

>  *La estructura real puede variar según cómo esté organizado el contenido en tu repositorio.*

Temas abordados esta semana

Redes en el contexto de Sistemas Operativos

- **Modelos de red y pila TCP/IP:** cómo los SO exponen interfaces para manejar conexiones, direcciones y protocolos.
- **Interfaces de red:** configuración de adaptadores, dirección IP, máscaras y rutas.

Programación de Sockets

- **Sockets TCP:** conexión confiable orientada a flujo de bytes.
- **Sockets UDP:** comunicación sin conexión para transmisión ligera.
- Ejemplos prácticos de clientes y servidores que envían y reciben datos.

Arquitectura Cliente-Servidor

- Estructura típica: procesos cliente que solicitan servicios a un servidor.
- Gestión de múltiples clientes concurrentes (hilos, procesos o multiplexación).

Sistemas Distribuidos

- Coordinación entre procesos en distintos nodos.
- Sincronización y consistencia básica en sistemas distribuidos.
- Consideraciones de tolerancia a fallos y escalabilidad.

Actividades y ejercicios sugeridos

1. **Leer cada uno de los apuntes teóricos** en orden para comprender networking y comunicación básica.
2. **Practicar con ejemplos de sockets** en lenguaje de programación de tu preferencia (por ejemplo, C, Python).

3. **Completar los ejercicios** en `05-EjerciciosSemana11.md`, centrados en la comunicación entre procesos remotos.

4. **Ejecutar los ejemplos de código** y observar el flujo de datos entre cliente y servidor en diferentes escenarios (TCP vs UDP).

Cómo usar este material

1. Abre los documentos ` .md` o ` .pdf` para estudiar los conceptos.
2. Usa las presentaciones para revisar visualmente los temas clave.
3. Ejecuta y modifica los ejemplos de código para experimentar con distintas configuraciones de red.
4. Resuelve los ejercicios y compara tus soluciones para afianzar lo aprendido.

Licencia y Atribución

Este material forma parte del repositorio *Sistemas-Operativos-2025* y sigue la licencia general del curso, permitiendo **su uso educativo con la debida atribución al autor original**.

Semana 12 – Repaso Final y Proyecto Integrador

Este directorio contiene los recursos, actividades, ejercicios y materiales correspondientes a la **Semana 12** del curso *Sistemas Operativos 2025*.

Objetivo de la Semana

El objetivo de esta semana es:

- Realizar un **repaso integral** de los principales temas vistos a lo largo del curso de Sistemas Operativos, consolidando conocimientos y habilidades de forma integrada.
- Aplicar estos conceptos en un **proyecto integrador o ejercicios avanzados**, reforzando teoría y práctica.
- Prepararse para exámenes finales o entregas finales de proyectos con ejemplos, guías y ejercicios de síntesis.
- Asegurar que los estudiantes comprendan las conexiones entre los distintos módulos del programa. :contentReference[oaicite:0]{index=0}

Contenido del Directorio

Archivo / Carpeta	Descripción
----- -----	
01-ResumenTemas.md / PDF	Síntesis de los temas principales del curso: procesos, memoria, E/S, archivos y sincronización.
02-GuiaDeEstudio.md / PDF	Guía de estudio con preguntas clave, esquemas y conceptos esenciales para preparar exámenes.
03-ProyectoIntegrador.md	Descripción de un proyecto que integra múltiples áreas de los sistemas operativos.
04-EjerciciosAvanzados.md	Ejercicios de nivel avanzado para aplicar los conocimientos adquiridos.
05-SolucionesEjercicios.md	Soluciones sugeridas o guías de respuesta para los ejercicios propuestos.
06-EjemplosCodigo/	Ejemplos de código consolidados que integran procesos, archivos, memoria y E/S.

| **07-Presentación12.pdf / PPTX**| Presentación resumen con puntos clave y mapas conceptuales. |

| ` README.md` | **Este documento**, con descripción general de todo lo que contiene la carpeta. |

> ! *La organización concreta de archivos puede variar dependiendo de cómo esté estructurado tu repositorio.*

> Esta propuesta está diseñada con un enfoque de **integración y repaso** típico de la última etapa de cursos académicos de sistemas operativos.

:contentReference[oaicite:1]{index=1}

Temas integrados en esta semana

Repaso de conceptos esenciales

- **Gestión de procesos**: creación, terminación, estados y planificación.

:contentReference[oaicite:2]{index=2}

- **Gestión de memoria**: paginación, segmentación y memoria virtual.

:contentReference[oaicite:3]{index=3}

- **Entrada/Salida y dispositivos**: buffering, controladores y planificación de discos.

:contentReference[oaicite:4]{index=4}

- **Sistemas de archivos**: organización, permisos, enlaces y administración avanzada. :contentReference[oaicite:5]{index=5}

- **Sincronización y concurrencia**: semáforos, mutexes, exclusión mutua y deadlocks. :contentReference[oaicite:6]{index=6}

Proyecto integrador (ejemplo de enfoque)

El **Proyecto Integrador** puede ser una aplicación o conjunto de ejercicios donde se demuestre:

- Coordinación de múltiples procesos o hilos.
- Uso de memoria compartida o manejo eficiente de memoria.
- Comunicación entre procesos mediante mecanismos IPC.
- Acceso y manipulación de archivos de forma concurrente.
- Manejo de errores y estados de recursos de entrada/salida.

Este tipo de trabajo ayuda a ver cómo los distintos módulos de un SO operan conjuntamente en un sistema real o simulado. :contentReference[oaicite:7]{index=7}

Actividades sugeridas esta semana

1. **Leer los resúmenes y guías de estudio** para reforzar la comprensión de los temas vistos.
2. **Completar los ejercicios avanzados** y comparar con las soluciones sugeridas.
3. **Desarrollar o continuar el proyecto integrador** documentando cada paso.
4. **Revisar las presentaciones** para afianzar esquemas y relaciones entre conceptos.

Cómo usar este material

1. Abre los archivos ` `.md` o ` `.pdf` en el orden recomendado para repasar de forma estructurada.
2. Utiliza la **guía de estudio** para practicar con preguntas clave.
3. Emplea el material de código para consolidar tus habilidades de implementación.
4. Documenta tus respuestas, pruebas de código y resultados como parte de tu revisión final.

Recursos adicionales

Para complementar tu estudio o profundizar aún más en temas específicos, puedes consultar materiales didácticos y apuntes disponibles en fuentes educativas abiertas sobre Sistemas Operativos, que abarcan desde teoría hasta ejemplos prácticos.
:contentReference[oaicite:8]{index=8}

Licencia y Atribución

Este material forma parte del repositorio *Sistemas-Operativos-2025* y se distribuye bajo la misma licencia de uso educativo con la debida atribución al autor original.

Semana 13 – Monitoreo y Control de Recursos en Sistemas Operativos

Este directorio contiene los recursos, apuntes, ejercicios y ejemplos correspondientes a la **Semana 13** del curso *Sistemas Operativos 2025*.

Objetivo de la Semana

El objetivo de esta semana es:

- Comprender las **herramientas y comandos para monitoreo y control de recursos** en sistemas operativos, tanto en tiempo de ejecución como de administración del sistema. :contentReference[oaicite:1]{index=1}
- Analizar cómo examinar el estado de los **procesos, memoria, entrada/salida y servicios** usando utilidades reales del sistema.
:contentReference[oaicite:2]{index=2}
- Practicar con ejemplos y ejercicios que integran conceptos clave de gestión y monitoreo en sistemas UNIX/Linux. :contentReference[oaicite:3]{index=3}

Contenido del Directorio

Archivo / Carpeta	Descripción
----- -----	
01-MonitoreoProcesos.md / PDF Introducción a herramientas de monitoreo de procesos (`ps`, `top`, etc.).	
02-GestiónMemoria.md / PDF Explicación de comandos para analizar uso de memoria (`free`, `vmstat`).	
03-ControlEYSDiscos.md / PDF Comandos y utilidades para examinar estadísticas de E/S y discos (`iostat`, `df`).	
04-OptimizaciónSO.md / PDF Técnicas para optimizar el rendimiento del sistema y priorización de tareas.	
05-EjerciciosSemana13.md Lista de ejercicios prácticos de monitoreo y control de recursos.	
06-EjemplosCodigo/ Scripts o ejemplos de uso de comandos en bash para monitoreo.	

| **07-Presentacion13.pdf / PPTX** | Presentación visual con explicación de conceptos y ejemplos. |

| ` README.md` | **Este documento**, con descripción general de la semana. |

> ! *La estructura real de archivos puede variar según cómo esté organizada la carpeta en el repositorio.*

Temas abordados esta semana

Monitoreo de Procesos

- **Herramientas básicas**: cómo ver procesos en ejecución, estados y recursos consumidos.

Ejemplos de comandos: `ps`, `top`, `htop`, `pgrep`.

- **Interpretación de salida**: entender columnas de CPU, memoria, tiempo de ejecución, etc.

Gestión de Memoria

- **Comandos de control de uso de memoria**: `free`, `vmstat`, `smem`, etc.

- **Interpretación de memoria física vs swap** y cómo impacta en el rendimiento.

Control de Entrada/Salida y Discos

- **Estadísticas de I/O**: revisar actividad de discos y dispositivos con `iostat`, `iotop`.

- **Espacio en discos**: uso y disponibilidad con `df`, `du`.

Optimización y Priorización de Recursos

- **Ajuste de prioridades**: uso de `nice` y `renice`.

- **Análisis de cuellos de botella**: cómo identificar y mitigar problemas de rendimiento.

Actividades y ejercicios sugeridos

1. **Leer los documentos teóricos** de cada sección para entender cómo funcionan las herramientas.

2. **Practicar con los comandos en un sistema real** (preferiblemente Linux/Unix) para observar resultados.
3. **Completar los ejercicios** del archivo `05-EjerciciosSemana13.md` .
4. **Ejecutar los ejemplos de scripts** para automatizar monitoreo y reportes de uso.

Cómo usar este material

1. Abre los archivos ` .md` o ` .pdf` para estudiar los conceptos principales.
2. Usa las presentaciones como complemento visual de los temas.
3. Prueba los ejemplos en un entorno real (terminal Linux).
4. Documenta tus resultados para comparar con respuestas esperadas o soluciones sugeridas.

Licencia y Atribución

Este material forma parte del repositorio *Sistemas-Operativos-2025* y se distribuye bajo la **misma licencia educativa**, permitiendo su uso con la debida atribución al autor original.

Semana 14 – Gestión de Entrada/Salida Avanzada y Dispositivos

Este directorio contiene los recursos, apuntes, ejercicios y ejemplos correspondientes a la **Semana 14** del curso *Sistemas Operativos 2025*.

Objetivo de la Semana

El objetivo de esta semana es:

- Profundizar en la **gestión de entrada/salida (E/S) avanzada** y el papel del sistema operativo en la administración de dispositivos y controladores.
:contentReference[oaicite:1]{index=1}
- Entender cómo se integran los **módulos del núcleo**, dispositivos de bloque y sistemas de ficheros especiales dentro del SO. :contentReference[oaicite:2]{index=2}
- Practicar técnicas relacionadas con la configuración, monitoreo y manipulación de dispositivos desde el núcleo. :contentReference[oaicite:3]{index=3}

Contenido del Directorio

Archivo / Carpeta	Descripción
-------------------	-------------

-----	-----
-------	-------

01-EntradaSalidaAvanzada.md / PDF Explicación general de E/S avanzada y administración de dispositivos.

02-DispositivosBloqueYCaracter.md Descripción de ficheros especiales de dispositivo para bloques y caracteres.
--

03-ModulosDelKernel.md / PDF Detalle de los módulos del núcleo, su gestión e interacción.

04-SistemasSysfsUdev.md / PDF Introducción a `sysfs`, `udev` y su papel en la administración dinámica de dispositivos.
--

05-PlanificacionDiscoAvanzada.md Temas de planificación de discos, colas y políticas de acceso optimizadas.

06-EjerciciosSemana14.md Lista de ejercicios prácticos sobre E/S avanzada y dispositivos.

| **07-EjemplosCodigo/**| Ejemplos de código y scripts para interacción con dispositivos y módulos. |

| **08-Presentacion14.pdf / PPTX**| Presentación visual con diagramas y ejemplos clave de E/S y controladores. |

| ` README.md` | **Este documento**, con descripción general de la semana. |

>  *La estructura exacta de archivos puede variar según cómo esté organizada tu carpeta.*

> Este esquema está enfocado en **temas de E/S avanzada y gestión de recursos periféricos**, comunes en cursos extendidos de sistemas operativos.

:contentReference[oaicite:4]{index=4}

Temas abordados esta semana

Gestión de Entrada/Salida (E/S) Avanzada

- El sistema operativo como **intermediario entre software y hardware** para E/S.
- Cómo se gestionan las operaciones de lectura/escritura más complejas, buffers y políticas de planificación.

Dispositivos de Bloque y de Carácter

- **Ficheros especiales de dispositivo** que representan dispositivos de bloque (discos, SSD) vs dispositivos de carácter (teclados, puertos).
- :contentReference[oaicite:5]{index=5}

- Técnicas de acceso y gestión eficiente según el tipo de dispositivo.

Módulos del Núcleo y Drivers

- **Módulos del kernel:** qué son, cómo se cargan y descargan, y su relación con los controladores de hardware. :contentReference[oaicite:6]{index=6}
- El uso de herramientas tipo `modprobe`, `lsmod`, y su papel en la administración del SO.

`sysfs` , `udev` y Administración Dinámica

- Sistemas como **`sysfs`** que exponen información de dispositivos al espacio de usuario. :contentReference[oaicite:7]{index=7}
- La función de **`udev`** para detectar y configurar dispositivos conforme se conectan o desconectan. :contentReference[oaicite:8]{index=8}

Actividades y ejercicios sugeridos

1. **Leer los apuntes teóricos** de cada documento en orden para comprender cómo se gestionan E/S y dispositivos.
2. **Practicar con los ejercicios** propuestos en `06-EjerciciosSemana14.md` para afianzar conceptos.
3. **Ejecutar los ejemplos de código** en un sistema Linux real para ver interacción con dispositivos y módulos del kernel.
4. **Revisar la presentación** para obtener una comprensión visual de los flujos de información y estructuras de E/S.

Cómo usar este material

1. Abre los archivos ` .md` o ` .pdf` para estudiar la teoría de forma estructurada.
2. Usa las presentaciones como apoyo visual para repasar los puntos clave.
3. Pon a prueba los ejemplos de código en un entorno real o máquina virtual para ver resultados.
4. Documenta tus resultados y compara tus soluciones a los ejercicios con las mejores prácticas.

Licencia y Atribución

Este material forma parte del repositorio *Sistemas-Operativos-2025* y se distribuye bajo la **misma licencia de uso educativo**, permitiendo su uso con la debida atribución al autor original.

Semana 15 – Seguridad, Protección y Gestión de Permisos en el Sistema Operativo

Este directorio contiene los recursos, apuntes y ejercicios correspondientes a la **Semana 15** del curso *Sistemas Operativos 2025*.

Objetivo de la Semana

Durante esta semana el enfoque principal es aprender:

- Los **principios fundamentales de seguridad y protección** dentro de un sistema operativo.
- El funcionamiento de los **mecanismos de control de acceso**, permisos y políticas de seguridad.
- Cómo implementar **seguridad básica en sistemas UNIX/Linux**, incluyendo usuarios, grupos, permisos y listas de control de acceso.
- Conceptos esenciales sobre **autenticación, criptografía básica y gestión segura de sistemas**.

Contenido del Directorio

Archivo / Carpeta	Descripción
01-SeguridadYProteccion.md / PDF	Introducción a la seguridad en sistemas operativos y modelos de protección.
02-PermisosUsuariosGrupos.md	Explicación de permisos tradicionales UNIX (lectura/escritura/ejecución).
03-ACLsYPoliticasAcceso.md	Uso de Listas de Control de Acceso (ACLs) y políticas avanzadas de permisos.
04-AutenticacionYCriptografia.md	Conceptos básicos: hashing, autenticación, contraseñas seguras.
05-SeguridadLinuxHerramientas.md	Comandos de seguridad esenciales: `chmod`, `chown`, `passwd`, `sudo`, `setfacl`, etc.
06-EjerciciosSemana15.md	Ejercicios prácticos aplicando permisos, ACLs y políticas de seguridad.

07-EjemplosCodigo/	Scripts y demostraciones de configuración y validación de permisos.
08-Presentacion15.pdf / PPTX	Presentación visual con los conceptos principales de seguridad y protección.
` README.md`	**Este documento**, descripción general de la semana.

Temas abordados esta semana

Seguridad y Protección en el SO

- Principios de **seguridad interna del sistema operativo**.
- Tipos de amenazas, riesgos y vectores de ataque comunes.
- Políticas de seguridad y modelos:
 - Control de acceso discrecional (DAC)
 - Control de acceso obligatorio (MAC)
 - Control de acceso basado en roles (RBAC)

Usuarios, Grupos y Permisos

- Sistema de usuarios y grupos en Linux/Unix.
- Permisos básicos:
 - **r** (lectura)
 - **w** (escritura)
 - **x** (ejecución)
- Representación numérica y simbólica de permisos.
- Propietarios, grupos y configuración vía comandos.

ACLs y Políticas Avanzadas

- Uso de **ACLs (Access Control Lists)** para permisos más detallados.
- Herramientas:
 - `getfacl`

- `setfacl`
- Aplicaciones prácticas en seguridad granular.

Autenticación y Criptografía Básica

- Hashing de contraseñas (SHA, MD5, bcrypt).
- Políticas de contraseñas seguras.
- Archivos de autenticación y su relevancia:
 - `/etc/passwd`
 - `/etc/shadow`

Herramientas de Seguridad en Linux

- Comandos esenciales de administración segura:
 - `chmod`, `chown`, `passwd`, `sudo`, `su`
- Configuración de privilegios y elevación segura.
- Buenas prácticas para administración de sistemas.

Actividades y ejercicios sugeridos

1. Cambiar permisos y propietarios de archivos usando ejemplos proporcionados.
2. Crear usuarios, grupos y roles simples para simular escenarios de acceso.
3. Aplicar ACLs para definir permisos granulares.
4. Analizar hashes de contraseñas y comprender cómo se protege el archivo `/etc/shadow` .
5. Responder preguntas de reflexión sobre seguridad aplicada al sistema operativo.

Cómo usar este material

1. Lee los documentos teóricos de los temas de seguridad.
2. Practica en una máquina virtual Linux con los comandos presentados.
3. Completa los ejercicios para afianzar el uso de permisos, ACLs y autenticación.
4. Revisa la presentación para reforzar la comprensión de conceptos clave.
5. Documenta los resultados y dudas para discutir con el profesor.



Licencia y Atribución

Este material forma parte del repositorio *Sistemas-Operativos-2025* y se distribuye bajo su **misma licencia educativa**. Cualquier uso debe atribuirse al autor original.

Semana 16 – Seguridad Avanzada y Técnicas de Protección en Sistemas Operativos

Este directorio contiene los recursos, apuntes, ejercicios y ejemplos correspondientes a la **Semana 16** del curso *Sistemas Operativos 2025*.

🔗 Objetivo de la Semana

Durante esta semana se aborda:

- Mecanismos **avanzados de seguridad y protección** dentro de los sistemas operativos. :contentReference[oaicite:1]{index=1}
- Conceptos de **seguridad integral**, incluyendo amenazas, contramedidas y herramientas. :contentReference[oaicite:2]{index=2}
- Políticas y técnicas para reforzar la protección de recursos críticos.
- Ejemplos prácticos y ejercicios enfocados en escenarios reales de administración segura del sistema.

📁 Contenido del Directorio

Archivo / Carpeta	Descripción
01-AmenazasYVulnerabilidades.md	Explicación de amenazas comunes y vulnerabilidades del sistema.
02-ProteccionYMecanismos.md	Técnicas y mecanismos de protección disponibles en el SO.
03-SeguridadEnProcesos.md	Seguridad aplicada a procesos y gestión de permisos seguros.
04-SeguridadEnMemoria.md	Técnicas de protección de memoria, ASLR, NX, y mitigaciones.
05-EjerciciosSemana16.md	Ejercicios de aplicación sobre seguridad del sistema.
06-HerramientasSeguridad/	Scripts y ejemplos con herramientas de seguridad.

| **07-Presentacion16.pdf / PPTX** | Presentación para repaso visual de conceptos.
|
| ` README.md` | **Este documento**, descripción general de la semana.
|

Temas abordados esta semana

Amenazas y Riesgos en Sistemas Operativos

- Tipos de amenazas: malware, acceso no autorizado, explotación de vulnerabilidades. :contentReference[oaicite:3]{index=3}
- Comprender vectores de ataque para proteger recursos sensibles. :contentReference[oaicite:4]{index=4}

Mecanismos de Protección del SO

- **Modelos de seguridad** y controles de acceso avanzados.
- **Barreras de protección**: mecanismos que limitan la capacidad de los procesos maliciosos.
- Técnica de asignación de roles y separaciones de privilegios.

Seguridad en Procesos

- Ejecución de procesos seguros:
- Minimización de privilegios
- Aislamiento de procesos
- Firmas de ejecución segura

Protección de Memoria

- Técnicas como **ASLR (Address Space Layout Randomization)** y **NX (No-Execute Bits)** para proteger contra ejecución no autorizada. :contentReference[oaicite:5]{index=5}
- Mitigaciones frente a exploit clásicos como desbordamientos de buffer.

Actividades y ejercicios sugeridos

1. **Revisar los documentos teóricos** para entender amenazas y mecanismos de protección.
2. **Aplicar políticas de seguridad** en ejemplos prácticos dentro de un entorno Linux o equivalente.
3. **Ejecutar los scripts y herramientas** presentes en la carpeta de ejemplos para experimentar con técnicas de protección.
4. **Resolver los ejercicios** de `05-EjerciciosSemana16.md` para reforzar lo aprendido.

Cómo usar este material

1. Abre los archivos ` .md` o ` .pdf` para estudiar cada tema paso a paso.
2. Usa las presentaciones como complemento visual.
3. Ejecuta los ejemplos y practica con las herramientas en un entorno seguro.
4. Documenta tus respuestas a los ejercicios y compáralas con soluciones sugeridas.

Licencia y Atribución

Este material forma parte del repositorio *Sistemas-Operativos-2025* y se distribuye bajo la licencia educativa del curso, **permitiendo su uso con la debida atribución al autor original**.