

PORAFOLIO DE EVIDENCIAS – SISTEMAS OPERATIVOS

GARCIA MACEDA
LUIS ANTONIO

Contenido

# Semana 1 — Introducción a los Sistemas Operativos	8
## 📦 Contenido	8
## 💡 Descripción ampliada de los recursos	8
### 📄 `00_intro.md`	8
### 📄 `01_conceptos_basicos.pdf`	9
### 📄 `02_historia_evolucion.pdf`	9
### 📁 `03_ejemplos_codigo/`	9
## 🛠️ Cómo usar este material	9
## 🚀 Recomendaciones	10
## 📚 Recursos sugeridos (no incluidos)	10
# Semana 2 — Sistemas Operativos 2025	11
## 📦 Contenido de la carpeta	11
## 💡 Descripción ampliada de los recursos	11
### 📄 `00_objetivos_semana2.md`	11
### 📄 `01_procesos_basicos.pdf`	12
### 📄 `02_creacion_y_finalizacion.pdf`	12
### 📁 `03_practica_procesos/`	12
### 📄 `ejercicios_semana2.ipynb` *(opcional)*	12
## 🎯 Recomendaciones para usar este material	13
## 🚀 Buenas prácticas	13
## 📚 (Opcional) Recursos externos sugeridos	13
# Semana 3 — Sistemas Operativos 2025	13
## 📦 Contenido	14
## 💡 Descripción ampliada de los recursos	14
### 📄 `00_objetivos_semana3.md`	14
### 📄 `01_hilos_y_concurrencia.pdf`	14
### 📄 `02_sincronizacion.pdf`	15

### 📁 `03_practica_hilos/`	15
### 📂 `04_ejercicios.ipynb`	15
## 🛠 Cómo usar este material	16
## 🎨 Buenas prácticas para esta semana	16
## 📚 Recursos adicionales sugeridos	16
# Semana 4 — Sistemas Operativos 2025	17
## 📦 Contenido (ejemplo)	17
## 🧠 Descripción ampliada de los recursos	17
### 📄 `00_objetivos_semana4.md`	17
### 📄 `01_planificacion_cpu.pdf`	17
### 📁 `02_algoritmos_planificacion/`	18
### 📄 `03_practica_planificacion.c`	18
### 📂 `04_ejercicios.ipynb`	18
## 🛠 Cómo usar este material	19
## 🎨 Buenas prácticas	19
## 📚 Recursos adicionales sugeridos	19
# Semana 5 — Sistemas Operativos 2025	20
## 📦 Contenido (ejemplo)	20
## 🧠 Descripción ampliada de los recursos	21
### 📄 `00_objetivos_semana5.md`	21
### 📄 `01_gestion_memoria.pdf`	21
### 📄 `02_paginacion.pdf`	21
### 📁 `03_practica_memoria/`	21
### 📂 `04_ejercicios.ipynb`	22
## 🛠 Cómo usar este material	22
## 🎨 Buenas prácticas	22
## 📚 Recursos adicionales sugeridos	22

# Semana 6 — Sistemas Operativos 2025	24
## Descripción ampliada de los recursos	24
### `00_objetivos_semana6.md`	24
### `01_sistemas_archivos.pdf`	25
### `02_manejo_espacio_datos.pdf`	25
### `03_practica_sistemas_archivos/`	25
### `04_ejercicios.ipynb`	26
## Cómo usar este material	26
## Buenas prácticas.....	26
## Recursos adicionales sugeridos	26
Semana 7 — Sistemas Operativos 2025	28
Contenido	28
Descripción ampliada de los recursos	28
00_objetivos_semana7.md.....	28
01_interbloqueo.pdf	28
02_comunicacion_procesos.pdf.....	29
03_practica_ipc/	29
04_ejercicios.ipynb	29
Cómo usar este material	29
Buenas prácticas.....	29
Recursos adicionales sugeridos.....	30
Semana 8 — Sistemas Operativos 2025	30
Contenido	30
Descripción ampliada de los recursos	30
00_objetivos_semana8.md.....	30
01_memoria_virtual.pdf.....	31
02_tecnicas_asignacion.pdf	31

📁 03_practica_memoria/	31
💻 04_ejercicios.ipynb	31
🔧 Cómo usar este material	32
📌 Buenas prácticas.....	32
📚 Recursos adicionales sugeridos.....	32
Semana 9 — Sistemas Operativos 2025	33
📦 Contenido	33
🧠 Descripción ampliada de los recursos	33
📘 00_objetivos_semana9.md.....	33
📄 01_sistemas_entrada_salida.pdf.....	33
📄 02_buffers_interrupciones.pdf	33
📁 03_practica_es/	34
💻 04_ejercicios.ipynb	34
🔧 Cómo usar este material	34
📌 Buenas prácticas.....	34
📚 Recursos adicionales sugeridos.....	34
Semana 10 — Sistemas Operativos 2025	35
📦 Contenido	35
🧠 Descripción ampliada de los recursos	35
📘 00_objetivos_semana10.md	35
📄 01_seguridad_procesos.pdf	35
📄 02_control_acceso_permisos.pdf	36
📁 03_practica_seguridad/	36
💻 04_ejercicios.ipynb	36
🔧 Cómo usar este material	36
📌 Buenas prácticas.....	36
📚 Recursos adicionales sugeridos.....	37

Semana 11 — Sistemas Operativos 2025	37
📦 Contenido	37
🧠 Descripción ampliada de los recursos	37
📘 00_objetivos_semana11.md	37
📄 01_redes_y_sockets.pdf	37
📄 02_protocolos_red.pdf	38
📁 03_practica_sockets/	38
📓 04_ejercicios.ipynb	38
🔧 Cómo usar este material	38
📌 Buenas prácticas.....	38
📚 Recursos adicionales sugeridos.....	39
Semana 12 — Sistemas Operativos 2025	39
📦 Contenido	39
🧠 Descripción ampliada de los recursos	39
📘 00_objetivos_semana12.md	39
📄 01_monitoreo_rendimiento.pdf.....	40
📄 02_herramientas_linux.pdf.....	40
📁 03_practica_rendimiento/	40
📓 04_ejercicios.ipynb	40
🔧 Cómo usar este material	41
📌 Buenas prácticas.....	41
📚 Recursos adicionales sugeridos.....	41
Semana 13 — Sistemas Operativos 2025	41
📦 Contenido	41
🧠 Descripción ampliada de los recursos	42
📘 00_objetivos_semana13.md	42
📄 01_backup_recuperacion.pdf	42

📄 02_restauracion_sistemas.pdf.....	42
📁 03_practica_backup/	42
💻 04_ejercicios.ipynb.....	43
🔧 Cómo usar este material	43
📌 Buenas prácticas.....	43
📚 Recursos adicionales sugeridos.....	43
 Semana 14 — Sistemas Operativos 2025	43
📦 Contenido	44
🧠 Descripción ampliada de los recursos	44
📘 00_objetivos_semana14.md	44
📄 01_finalizacion_curso.pdf.....	44
📄 02_proyecto_integrador.pdf	44
📁 03_practica_proyecto/	44
💻 04_ejercicios.ipynb	45
🔧 Cómo usar este material	45
📌 Buenas prácticas.....	45
📚 Recursos adicionales sugeridos.....	45
 Semana 15 — Sistemas Operativos 2025	45
📦 Contenido	46
🧠 Descripción ampliada de los recursos	46
📘 00_objetivos_semana15.md	46
📄 01_seguridad_sistemas_operativos.pdf	46
📄 02_privilegios_acceso.pdf.....	46
📁 03_practica_seguridad_sistemas/	47
💻 04_ejercicios.ipynb	47
🔧 Cómo usar este material	47
📌 Buenas prácticas.....	47

	Recursos adicionales sugeridos.....	47
	Semana 16 — Sistemas Operativos 2025	48
	Contenido	48
	Descripción ampliada de los recursos	48
	00_objetivos_semanal16.md	48
	01_tendencias_sistemas_operativos.pdf	48
	02_virtualizacion_containers.pdf	49
	03_practica_virtualizacion/	49
	04_ejercicios.ipynb	49
	Cómo usar este material	49
	Buenas prácticas.....	49
	Recursos adicionales sugeridos.....	50

Semana 1 — Introducción a los Sistemas Operativos

Este documento describe el contenido de la **Semana 1** del curso *Sistemas Operativos 2025*, perteneciente al repositorio **Sistemas-Operativos-2025**.

Aquí se presentan los archivos incluidos en esta carpeta, junto con una **descripción ampliada** de su propósito y cómo usar cada recurso.

📁 Contenido

> **AVISO:** Reemplaza los nombres y descripciones de cada archivo según corresponda una vez que confirmes el listado real.

Archivo Carpeta Tipo Descripción

`00_intro.md` Documento Introducción al curso, objetivos de la semana y guía de estudio.

`01_conceptos_basicos.pdf` PDF Contiene los conceptos fundamentales sobre sistemas operativos: definición, funciones y estructura general.

`02_historia_evolucion.pdf` PDF Explica la evolución de los sistemas operativos desde sus orígenes hasta los modernos.

`03_ejemplos_codigo` Carpeta Contiene ejemplos de código relacionados con llamadas al sistema o procesos básicos.

`README.md` Documento Este archivo de descripción general de la semana.

> **Nota:** Si en tu caso los nombres son distintos (por ejemplo `introduccion.md`, `practica1/` o `slides/`), reemplázalos en esta tabla.

🧠 Descripción ampliada de los recursos

📄 `00_intro.md`

Este documento contiene una presentación general de la **Semana 1**, los objetivos de aprendizaje, temas centrales y cómo se estructura el contenido de los demás archivos.

Ideal para comenzar a entender el enfoque del curso y qué se espera que domines al finalizar esta semana.

`01_conceptos_basicos.pdf`

Aquí se profundiza en los **conceptos fundamentales de los sistemas operativos**, tales como:

- ¿Qué es un sistema operativo?
- Funciones principales (gestión de procesos, memoria, E/S, etc.).
- Elementos que conforman un SO.

Este material es la base teórica para comprender todo lo que se abordará en semanas posteriores.

`02Historia_evolucion.pdf`

Presenta un **recorrido histórico y evolutivo** de los sistemas operativos, desde los primeros mainframes hasta los sistemas modernos como UNIX y Linux.

Incluye contextos tecnológicos y ejemplos de cómo han cambiado los modelos de cómputo y la gestión de recursos.

`03_ejemplos_codigo/`

Esta carpeta agrupa ejemplos de código y prácticas relacionadas con la teoría de la semana 1, por ejemplo:

- Código en C o scripts que muestren llamadas al sistema básicas.
- Secuencias de comandos para ilustrar conceptos de procesos, gestión o interacción con el sistema.

Si hay código fuente, explica también cómo compilar o ejecutar.

Cómo usar este material

1. **Lee primero los documentos introductorios** para comprender el contexto y los objetivos.

2. **Revisa los PDFs y notas teóricas** para asentar los conceptos básicos.
3. **Ejecuta y estudia los ejemplos de código** para ver aplicada la teoría.
4. **Haz tus anotaciones y preguntas** sobre cualquier punto que no quede claro.

Recomendaciones

- Mantén esta carpeta actualizada con títulos claros y descripciones completas.
- Si agregas vídeos, presentaciones o ejercicios prácticos, crea subcarpetas como `videos/`, `slides/` o `practicas/`.
- Añade referencias bibliográficas si usas material externo.

Recursos sugeridos (no incluidos)

Para complementar tu estudio de *Semana 1*, puedes consultar:

- Textos introductorios clásicos sobre sistemas operativos (por ejemplo, *Operating Systems: Three Easy Pieces*).
 - Documentos de la cátedra que cubren temas como llamadas al sistema, procesos y estructura de SO.
 - Tutoriales en Linux sobre comandos básicos y navegación por el sistema.
- > Una vez que me compartas el **listado real de archivos dentro de `Semana1`**, puedo actualizar este `README.md` automáticamente con **nombres de archivos reales y descripciones específicas** adaptadas al contenido.

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

Semana 2 — Sistemas Operativos 2025

Bienvenido a la **Semana 2** del curso *Sistemas Operativos 2025*.

Aquí se agrupa el material (teórico y práctico) correspondiente a esta segunda semana de clases.

📁 Contenido de la carpeta

> **IMPORTANTE:** Reemplaza estos nombres con los reales una vez confirmados.

Archivo Carpeta Tipo Descripción

`00_objetivos_semana2.md` Documento Objetivos de aprendizaje de la semana.

`01_procesos_basicos.pdf` PDF Explicación teórica sobre procesos y estados de proceso.

`02_creacion_y_finalizacion.pdf` PDF Documentación sobre creación y terminación de procesos.

`03_practica_procesos` Carpeta | Ejercicios y ejemplos prácticos sobre procesos en Linux/C.

`ejercicios_semana2.ipynb` Notebook Actividades interactivas para reforzar conceptos.

`README.md` Documento Este archivo (descripción general).

🍪 Descripción ampliada de los recursos

📄 `00_objetivos_semana2.md`

Este archivo detalla los **objetivos que debes alcanzar al concluir la segunda semana**, como comprender la gestión de procesos, los estados de ejecución, y la funcionalidad básica del planificador.

Incluye también vínculos con ejercicios y actividades sugeridas para tu autoestudio.

`01_procesos_basicos.pdf`

Contiene la **teoría central de la semana 2**, incluyendo conceptos como:

- Qué es un proceso y en qué se diferencia de un programa.
- Estados de proceso (nuevo, listo, ejecutando, bloqueado, terminado).
- Cómo un sistema operativo gestiona el ciclo de vida de un proceso.

Este material es ideal para leer antes de intentar los ejercicios prácticos.

`02_creacion_y_finalizacion.pdf`

Profundiza en:

- Cómo se **crea** un proceso (p. ej., llamadas `fork()` en Unix).
- Cómo se **termina** un proceso correctamente.
- Diferencias entre procesos padre e hijo.
- Ejemplos y diagramas de transición de estados.

`03_practica_procesos/`

Una colección de **ejemplos y prácticas** sobre:

- Código en C que crea y maneja procesos.
- Scripts de Bash para observar estados de procesos (`ps`, `top`, etc.).
- Explicaciones de cómo ejecutar y probar cada ejemplo.

Incluye instrucciones de compilación y ejecución.

`ejercicios_semana2.ipynb` *(opcional)*

Cuaderno interactivo (por ejemplo, en Jupyter Notebook) con ejercicios, reflexiones y quizzes que te ayudan a reforzar la teoría revisada en la semana.

Puede incluir:

- Modelos de planificación simples.
- Comparaciones entre estados de procesos.
- Bloques de código ejecutables para experimentar.

Recomendaciones para usar este material

1. **Primero**, lee los objetivos y la teoría principal.
2. **Luego**, revisa ejemplos de código y asegúrate de entender cada línea.
3. **Finalmente**, realiza los ejercicios y practica con herramientas reales como `ps`, `top`, `strace` o compiladores C en Linux.

Buenas prácticas

- Mantén los nombres de archivos claros y descriptivos.
- Si agregas más recursos (p. ej., vídeos o enlaces), crea carpetas como `videos/` o `links/`.
- Añade referencias bibliográficas si usas material externo.

(Opcional) Recursos externos sugeridos

Para profundizar en temas de procesos en sistemas operativos puedes consultar:

- Material docente de cursos de sistemas operativos (gestión de procesos, planificadores, context switching).
- Apuntes o libros reconocidos sobre sistemas operativos.

Semana 3 — Sistemas Operativos 2025

Este documento describe el contenido de la **Semana 3** del curso *Sistemas Operativos 2025*, dentro del repositorio. Aquí se enlistan los archivos y carpetas de esta semana junto con una descripción ampliada de cada recurso.

📁 Contenido

> **Nota:** Reemplaza los nombres de archivo de ejemplo con los reales una vez confirmados.

Archivo Carpeta Tipo Descripción

`00_objetivos_semana3.md` Documento Objetivos de aprendizaje de la semana 3.

`01_hilos_y_concurrencia.pdf` PDF Explicación teórica sobre hilos, concurrencia y paralelismo.

`02_sincronizacion.pdf` PDF Material sobre sincronización de hilos: mutexes, semáforos, monitores.

`03_practica_hilos` Carpeta Código y ejercicios prácticos sobre creación y sincronización de hilos.

`04_ejercicios.ipynb` Notebook Actividades interactivas para ejercitarse en los temas de la semana.

`README.md` | Documento | Este archivo de descripción general.

🎯 Descripción ampliada de los recursos

📄 `00_objetivos_semana3.md`

Documento que detalla los **objetivos de aprendizaje** para esta semana, por ejemplo:

- Comprender qué es un hilo de ejecución.
- Diferenciar entre procesos e hilos.
- Entender conceptos de **concurrencia y paralelismo**.
- Aprender mecanismos de sincronización básicos.

📄 `01_hilos_y_concurrencia.pdf`

Material teórico que cubre:

- Definición y ventajas de los hilos frente a procesos.

- Modelos de hilos (usuario vs kernel).
- Ejemplos de creación y manejo de hilos en sistemas operativos.

`02_sincronizacion.pdf`

Profundiza en cómo **coordinar la ejecución de múltiples hilos** sin generar condiciones de carrera o inconsistencias, con temas tales como:

- **Mutexes (exclusión mutua)**
- **Semáforos**
- **Monitores y variables de condición**
- Ejemplos y diagramas de secuencias de ejecución.

`03_practica_hilos/`

Carpeta de práctica con:

- **Código en C o Python** que ilustra la creación de hilos.
- Ejercicios para aplicar mecanismos de sincronización.
- Instrucciones para compilar o ejecutar los ejemplos en tu entorno.

Incluye pasos concretos de compilación, ejecución y pruebas.

`04_ejercicios.ipynb`

Cuaderno (**Jupyter Notebook**) con ejercicios interactivos para reforzar los temas principales:

- Crear hilos y observar su comportamiento.
- Comparar rendimiento con y sin sincronización.
- Resolver mini-retos de concurrencia.

Cómo usar este material

1. **Revisa primero el apartado de objetivos** para entender qué se espera que aprendas.
2. **Lee los PDFs teóricos** para asentarte en los conceptos centrales.
3. **Ejecuta los ejemplos de código** dentro de `03_practica_hilos/` para ver la teoría aplicada.
4. **Realiza los ejercicios interactivos** del notebook si está disponible.

Buenas prácticas para esta semana

- Antes de modificar código, **entiende el problema de la condición de carrera**.
- Documenta cada ejercicio con resultados observados y comentarios.
- Haz commits frecuentes con descripciones claras de lo que cambia.

Recursos adicionales sugeridos

Para complementar el estudio, puedes consultar bibliografía estándar sobre sistemas operativos y concurrencia en programación, como ejemplos en línea o textos clásicos.

> _Este documento puede ser actualizado para adaptarse a los nombres reales de los archivos en el directorio `Semana3` dentro de este repositorio.

Semana 4 — Sistemas Operativos 2025

Este documento describe el contenido de la **Semana 4** del curso *Sistemas Operativos 2025*, dentro del repositorio. Aquí se enlistan los archivos y carpetas de esta semana junto con una descripción ampliada de cada recurso.

📁 Contenido (ejemplo)

> **IMPORTANTE:** Reemplaza los nombres de archivo de ejemplo con los reales una vez confirmados.

Archivo Carpeta Tipo Descripción

`00_objetivos_semana4.md` Documento Objetivos de aprendizaje detallados.

`01_planificacion_cpu.pdf` PDF Teoría sobre algoritmos de planificación de CPU.

`02_algoritmos_planificacion/` Carpeta Código y ejemplos de planificación (FCFS, SJF, Round Robin).

`03_practica_planificacion.c` Código fuente Ejercicio práctico de planificación de procesos.

`04_ejercicios.ipynb` Notebook Ejercicios interactivos y análisis.

`README.md` Documento Este archivo de descripción general.

📖 Descripción ampliada de los recursos

📄 `00_objetivos_semana4.md`

Documento que presenta los **objetivos específicos de la semana**, como entender los algoritmos de planificación de CPU, sus ventajas y desventajas, así como métricas de evaluación.

📄 `01_planificacion_cpu.pdf`

Material teórico extenso que cubre:

- Qué es la planificación de CPU.
- Principales algoritmos (FCFS, SJF, Priority, Round Robin).
- Comparativas de rendimiento de cada algoritmo.
- Ejemplos y diagramas de ejecución.

Ideal para leer antes de hacer ejercicios prácticos.

📁 `02_algoritmos_planificacion/`

Carpeta que contiene:

- Implementaciones de diferentes **algoritmos de planificación** en C (o el lenguaje del curso).
- Casos de prueba para observar comportamiento con distintas cargas de procesos.
- Scripts y notas que explican cómo compilar y ejecutar cada ejemplo.

💾 `03_practica_planificacion.c`

Código fuente de una práctica guiada que incluye:

- Simulación de trabajos/procesos.
- Ejecución de diversos algoritmos.
- Instrucciones de compilación y ejecución paso a paso.

Asegúrate de compilar con instrucciones claras en *Linux* o el entorno usado por la materia.

📈 `04_ejercicios.ipynb`

Notebook interactivo (por ejemplo, Jupyter) que incluye:

- Tablas de comparación entre distintos algoritmos de planificación.
- Gráficas y métricas de rendimiento.
- Preguntas para reflexión y autoevaluación.

Cómo usar este material

1. **Empieza por objetivos** para enfocar tu estudio.
2. **Revisa la teoría de planificación** en el PDF antes de pasar a código.
3. **Ejecuta los ejemplos de la carpeta de algoritmos** para ver cómo funcionan.
4. **Completa los ejercicios del notebook** para afianzar conceptos.

Buenas prácticas

- Mantén nombres de archivos descriptivos y consistentes.
- Si agregas más recursos (p. ej., vídeos, presentaciones), crea subcarpetas `videos/` , `slides/` .
- Añade referencias bibliográficas si usas material extra.

Recursos adicionales sugeridos

Para profundizar sobre planificación de CPU puedes consultar:

- Material docente estándar sobre sistemas operativos: planificación de procesos y métricas de evaluación.
 - Texto “Operating Systems: Three Easy Pieces” o Silberschatz/Galvin.
 - Recursos en línea con visualizaciones de algoritmos de planificación.
- > _Este documento puede ser actualizado para adaptarse a los nombres reales de los archivos una vez que se agreguen aquí.
- 📌 **Nota:** si puedes abrir el enlace en tu navegador y copiar aquí **el listado exacto de archivos/carpetas de `Semana4`**, actualizaré este README automáticamente con descripciones precisas de cada recurso.

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

Semana 5 — Sistemas Operativos 2025

Este documento describe el contenido de la **Semana 5** del curso *Sistemas Operativos 2025*. Aquí se enlistan los archivos y carpetas típicos de esta semana junto con una descripción ampliada de cada recurso.

📁 Contenido (ejemplo)

> **IMPORTANTE:** Sustituye los nombres de archivo de ejemplo por los reales una vez confirmados.

Archivo Carpeta Tipo Descripción

`00_objetivos_semana5.md` Documento Objetivos de aprendizaje de la semana 5.

`01_gestion_memoria.pdf` PDF Teoría sobre administración de memoria en sistemas operativos.

`02_paginacion.pdf` PDF Material sobre paginación y manejo de memoria virtual.

`03_practica_memoria` Carpeta Practicas y ejemplos de paginación, segmentos, etc.

`04_ejercicios.ipynb` Notebook Ejercicios interactivos relacionados con memoria y paginación.

`README.md` Documento Este archivo de descripción general.

📄 Descripción ampliada de los recursos

📄 `00_objetivos_semana5.md`

Este archivo contiene los **objetivos específicos de la semana 5**, por ejemplo:

- Aprender los fundamentos de la **gestión de memoria**.
- Comprender **paginación y segmentación**.
- Conocer los mecanismos de acceso y optimización de memoria virtual.

Se recomienda leerlo al inicio para orientar el estudio.

📄 `01_gestion_memoria.pdf`

Documento con la **teoría principal** de administración de memoria:

- Conceptos de memoria física y virtual.
- Políticas de asignación de memoria.
- Fragmentación interna y externa.
- Ejercicios conceptuales y ejemplos ilustrativos.

Es el material base antes de pasar a prácticas.

📄 `02_paginacion.pdf`

Profundiza en **paginación y técnicas de memoria virtual**:

- Qué es la paginación y cómo permite la abstracción de memoria.
- Tablas de páginas, niveles de paginación.
- Ejemplos y diagramas que muestran la traducción de direcciones.

Incluye comparativas con otros mecanismos (por ejemplo, segmentación).

📂 `03_practica_memoria/`

Carpeta que contiene **ejercicios de práctica** con:

- Código en C (o lenguaje del curso) para simular paginación.
- Scripts y casos de prueba para explorar diferentes cargas de memoria.
- Instrucciones claras para compilar y ejecutar los ejemplos.

Ideal para reforzar la teoría con ejemplos prácticos.

`04_ejercicios.ipynb`

Notebook (por ejemplo, *Jupyter Notebook*) con **ejercicios interactivos**:

- Explora modelos de paginación y tablas de páginas.
- Preguntas guiadas y cálculos de direcciones virtuales.
- Evaluaciones cortas de auto-aprendizaje.

Permite ejecutar bloques de código y experimentar.

Cómo usar este material

1. **Revisa primero los objetivos** de la semana.
2. **Estudia la teoría principal** en los PDFs incluidos.
3. **Ejecuta los ejemplos prácticos** para ver la teoría aplicada.
4. **Realiza los ejercicios interactivos** para consolidar tu comprensión.

Buenas prácticas

- Usa nombres claros y consistentes para los archivos.
- Si agregas nuevos recursos (videos, presentaciones, más código), crea carpetas específicas como `videos/`, `slides/`, `practices/`.
- Incluye referencias bibliográficas si usas material externo adicional.

Recursos adicionales sugeridos

- Bibliografía estándar de *Sistemas Operativos*, capítulos sobre **gestión de memoria y paginación**.

- Documentación de sistemas reales (por ejemplo, Linux) sobre **manejo de memoria virtual**.
 - Simuladores o herramientas en línea para visualizar traducciones de memoria.
- > _Este documento sirve como plantilla; una vez que obtengas los nombres reales de los archivos en `Semana5`, reemplaza los ejemplos para reflejar con precisión el contenido del repositorio.

Semana 6 — Sistemas Operativos 2025

Este documento describe el contenido de la **Semana 6** del curso *Sistemas Operativos 2025*.

Incluye una lista de los materiales (teoría, prácticas, ejercicios) y una descripción ampliada de cada recurso.

📁 Contenido (ejemplo)

> **IMPORTANTE:** Reemplaza estos nombres de archivo de ejemplo con los reales una vez confirmados.

Archivo Carpeta Tipo Descripción

`00_objetivos_semana6.md` Documento Objetivos de aprendizaje de la semana 6.

`01_sistemas_archivos.pdf` PDF Teoría sobre sistemas de archivos y estructuras de datos relacionadas.

`02_manejo_espacio_datos.pdf` PDF Material sobre cómo el SO administra espacio en disco y bloques.

`03_practica_sistemas_archivos/` Carpeta Código y ejercicios prácticos sobre creación y manipulación de sistemas de archivos.

`04_ejercicios.ipynb` Notebook Ejercicios interactivos y reflexiones guiadas sobre esta semana.

`README.md` Documento Este archivo de descripción general.

📖 Descripción ampliada de los recursos

📄 `00_objetivos_semana6.md`

Documento introductorio que presenta los **objetivos específicos de la semana**.

Por ejemplo:

- Entender qué es un sistema de archivos.

- Comprender las estructuras de datos que utiliza un sistema operativo para organizar archivos.
- Conocer mecanismos de acceso, permisos y sincronización de E/S.

Lee este archivo primero para orientar tu estudio.

`01_sistemas_archivos.pdf`

Material teórico que cubre:

- Qué es un **sistema de archivos**.
- Cómo se organizan directorios y archivos en disco.
- Estructuras como **inode**, **bloques**, **tablas de archivos**, etc.
- Ejemplos en sistemas reales como ext4 o NTFS.

Este documento sirve de base para entender las prácticas posteriores.

`02_manejo_espacio_datos.pdf`

Explica cómo el sistema operativo **gestiona el espacio de almacenamiento**, incluyendo:

- Localización y asignación de bloques.
- Fragmentación interna y externa.
- Mecanismos de cache y buffer de E/S.
- Ejemplos de algoritmos de asignación.

Incluye diagramas y casos ilustrativos.

`03_practica_sistemas_archivos/`

Carpeta que contiene **ejercicios prácticos** como:

- Código en C (o el lenguaje del curso) para manipular estructuras de archivos.
- Scripts que muestran cómo montar, desmontar, listar y modificar sistemas de archivos.
- Casos de prueba y retos para reforzar la teoría.

Incluye instrucciones de compilación y ejecución paso a paso.

`04_ejercicios.ipynb`

Notebook (*Jupyter Notebook* u otro formato interactivo) con ejercicios:

- Actividades de reflexión guiada.
- Cálculos y simulaciones de estructuras de archivos.
- Preguntas de autoevaluación con ejemplos ejecutables.

Cómo usar este material

1. **Empieza con los objetivos** para saber qué esperas aprender.
2. **Lee la teoría** para asentar las bases conceptuales.
3. **Ejecuta las prácticas** para ver la aplicación real de lo aprendido.
4. **Completa los ejercicios interactivos** para consolidar tu comprensión.

Buenas prácticas

- Mantén nombres claros y descriptivos.
- Si hay muchos recursos adicionales (como vídeos o gráficas), considera subcarpetas como `videos/` o `slides/`.
- Documenta cada ejemplo con comentarios y resultados observados.

Recursos adicionales sugeridos

Puedes consultar libros estándar de sistemas operativos para profundizar en:

- Estructura y diseño de sistemas de archivos.

- Algoritmos de asignación y búsqueda de bloques.
- Mecanismos de protección, permisos y seguridad de acceso.

> _Este documento puede ser actualizado para adaptarse a los nombres reales de los archivos una vez que me compartas el listado de `Semana6`._

👉 **Por favor comparte los nombres de los archivos y carpetas de `Semana6`** para que pueda personalizar este README automáticamente con descripciones precisas de cada recurso.

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

Semana 7 — Sistemas Operativos 2025

Este documento describe el contenido de la **Semana 7** del curso *Sistemas Operativos 2025*. Incluye una lista de los materiales (teoría, prácticas, ejercicios) y una descripción ampliada de cada recurso.

Contenido

00_objetivos_semana7.md Documento Objetivos de aprendizaje de la semana 7.
01_interbloqueo.pdf PDF Teoría sobre interbloqueos y condiciones necesarias para que ocurran.
02_comunicacion_procesos.pdf PDF Material sobre mecanismos de comunicación entre procesos (IPC).
03_practica_ipc/ Carpeta Código y ejercicios prácticos sobre IPC, semáforos y sincronización.
04_ejercicios.ipynb Notebook Ejercicios interactivos y reflexiones guiadas de la semana.
README.md Documento Este archivo de descripción general.

Descripción ampliada de los recursos

00_objetivos_semana7.md

Documento introductorio que presenta los **objetivos específicos de la semana**. Incluye metas como:

- Comprender qué es un interbloqueo.
 - Identificar las condiciones que lo provocan.
 - Entender mecanismos de comunicación entre procesos (IPC).
- Lee este archivo primero para orientar tu estudio.

01_interbloqueo.pdf

Material teórico que cubre:

- Definición de interbloqueo (deadlock).
 - Condiciones de Coffman: exclusión mutua, espera circular, retención y no expulsión, y espera bloqueante.
 - Métodos de prevención, evasión y detección de interbloqueos.
- Incluye ejemplos y casos prácticos.



Explica de forma clara los mecanismos de **comunicación entre procesos**, incluyendo:

- Tuberías (pipes) y tuberías con nombre.
- Colas de mensajes.
- Memoria compartida.
- Semáforos y sincronización básica.

Incluye diagramas, ejemplos y estructura general de IPC en sistemas UNIX/Linux.



Carpeta que contiene **ejercicios prácticos**, como:

- Código en C para semáforos y memoria compartida.
- Ejemplos de cómo usar pipes y colas de mensajes.
- Scripts para compilar y ejecutar los programas.

Incluye instrucciones paso a paso para ejecutar cada práctica.



Notebook interactivo con ejercicios como:

- Problemas guiados sobre IPC.
- Prácticas de sincronización entre procesos.
- Preguntas de autoevaluación.

Permite ejecutar código directamente y ver resultados.

Cómo usar este material

1. **Empieza con los objetivos** para saber qué aprenderás.
2. **Revisa la teoría** para construir bases sólidas.
3. **Realiza las prácticas de código** para aplicar los conceptos.
4. **Completa los ejercicios** del notebook para reforzar tu aprendizaje.

Buenas prácticas

- Usa nombres claros y consistentes en los archivos.
- Si agregas materiales extra, organízalos en subcarpetas.
- Documenta tus resultados y observaciones.

Recursos adicionales sugeridos

Puede complementar esta información con bibliografía estándar como:

- *Operating Systems Concepts* (Silberschatz).
- *Operating Systems: Three Easy Pieces*.
- Documentación de Linux sobre IPC, semáforos y memoria compartida.

Semana 8 — Sistemas Operativos 2025

Este documento describe el contenido de la **Semana 8** del curso *Sistemas Operativos 2025*. Incluye una lista de los materiales (teoría, prácticas, ejercicios) y una descripción ampliada de cada recurso.

Contenido

00_objetivos_semana8.md Documento Objetivos de aprendizaje de la semana 8.

01_memoria_virtual.pdf PDF Teoría sobre memoria virtual, paginación y administración de memoria.

02_tecnicas_asignacion.pdf PDF Material sobre técnicas de asignación de memoria y reemplazo de páginas.

03_practica_memoria/ Carpeta Código y ejercicios prácticos sobre paginación, marcos, fallos de página y simuladores.

04_ejercicios.ipynb Notebook Ejercicios interactivos y simulaciones de memoria virtual.

README.md Documento Este archivo de descripción general.

Descripción ampliada de los recursos

00_objetivos_semana8.md

Documento introductorio que presenta los **objetivos específicos de la semana**. Incluye metas como:

- Comprender el concepto de memoria virtual.
 - Saber cómo funciona la paginación y segmentación.
 - Analizar técnicas de reemplazo de páginas.
 - Identificar ventajas y desventajas de usar memoria virtual.
- Es recomendable leerlo primero para tener claridad del enfoque de la semana.



Material teórico que cubre:

- Qué es la **memoria virtual** y por qué se usa.
- Espacio lógico vs. espacio físico.
- Tablas de páginas, TLB y acceso a memoria.
- Ventajas como aislamiento, protección y uso eficiente de RAM.
Incluye diagramas y ejemplos prácticos de accesos a páginas.



Explica técnicas clásicas de administración y reemplazo de memoria:

- Paginación simple y multinivel.
- Segmentación y segmentación paginada.
- Algoritmos de reemplazo: FIFO, LRU, Optimal, Clock, NRU.
- Casos donde ocurren fallos de página y cómo se manejan.
El documento compara cada técnica y explica cuándo es más eficiente cada una.



Carpeta que contiene **actividades prácticas**, como:

- Programas en C para simular tablas de páginas.
- Simuladores de fallos de página y marcos de memoria.
- Ejercicios para aplicar algoritmos de reemplazo.
- Scripts para compilar y ejecutar los programas en Linux.
Incluye instrucciones claras y resultados esperados.



Notebook interactivo enfocado en:

- Simulaciones de accesos a memoria.
- Ejercicios con secuencias de referencia de páginas.
- Comparación entre algoritmos de reemplazo.
- Pruebas y evaluaciones automáticas.
Permite observar visualmente cómo cambia el rendimiento según el algoritmo utilizado.

Cómo usar este material

1. **Empieza con los objetivos** para entender el propósito de la semana.
2. **Lee la teoría** sobre memoria virtual y administración de memoria.
3. **Realiza las prácticas**, probando algoritmos y simuladores.
4. **Resuelve los ejercicios** del notebook para evaluar tu comprensión.

Buenas prácticas

- Documenta tus pruebas, especialmente los algoritmos utilizados.
- Mantén un registro de resultados para comparar desempeño.
- Usa nombres claros para tus scripts y archivos de pruebas.

Recursos adicionales sugeridos

Puedes complementar la información consultando:

- *Operating Systems Concepts* (Silberschatz).
- *Modern Operating Systems* (Tanenbaum).
- *Operating Systems: Three Easy Pieces* (capítulos de memoria virtual).
- Documentación de Linux sobre paginación y administración de memoria.

Semana 9 — Sistemas Operativos 2025

Este documento describe el contenido de la **Semana 9** del curso *Sistemas Operativos 2025*. Incluye una lista de los materiales (teoría, prácticas y ejercicios) y una descripción ampliada de cada recurso.

Contenido

00_objetivos_semana9.md Documento Objetivos de aprendizaje de la semana 9.
01_sistemas_entrada_salida.pdf PDF Teoría sobre administración de dispositivos, controladores y manejo de E/S.
02_buffers_interrupciones.pdf PDF Material sobre buffers, interrupciones, DMA y controladores.
03_practica_es/ Carpeta Prácticas y ejercicios sobre sistemas de entrada/salida.
04_ejercicios.ipynb Notebook Actividades interactivas y simulaciones de la semana.
README.md Documento Este archivo de descripción general.

Descripción ampliada de los recursos

00_objetivos_semana9.md

Documento introductorio que presenta los **objetivos específicos de la semana**, tales como:

- Comprender cómo funciona la **entrada/salida en un sistema operativo**.
- Conocer el rol de los controladores y módulos de hardware.
- Entender conceptos como interrupciones, buffering y DMA.
Recomendado leerlo primero.

01_sistemas_entrada_salida.pdf

Material teórico que explica:

- Qué es un **sistema de E/S** dentro de un sistema operativo.
- Tipos de dispositivos y su administración.
- Arquitectura de controladores y módulos de comunicación.
- Métodos de manejo de dispositivos: programado, interrumpido y DMA.
Incluye diagramas y ejemplos ilustrativos.

02_buffers_interrupciones.pdf

Documento que profundiza en:

- Buffers: simples, dobles y circulares.
- Cómo funcionan las **interrupciones** y los manejadores de interrupción.
- Transferencias directas a memoria (**DMA**).
- Comparativa entre métodos de transferencia.
Perfecto para relacionar hardware con funcionamiento del núcleo del sistema.

03_practica_es/

Carpeta que contiene prácticas centradas en:

- Simulación de operaciones de entrada/salida.
- Ejercicios sobre interrupciones y priorización.
- Códigos de ejemplo relacionados con buffers o controladores.
Incluye instrucciones de compilación y de ejecución.

04_ejercicios.ipynb

Notebook interactivo que incluye:

- Problemas guiados sobre E/S.
- Actividades para analizar interrupciones y DMA.
- Ejercicios automáticos para practicar los conceptos.
Permite visualizar procesos y medir tiempos simulados.

Cómo usar este material

1. **Comienza con los objetivos** para saber qué dominar esta semana.
2. **Revisa la teoría** para comprender E/S y controladores.
3. **Realiza las prácticas** para aplicar los conceptos aprendidos.
4. **Completa los ejercicios** del notebook para reforzar y evaluar tu nivel.

Buenas prácticas

- Documenta tus resultados en las prácticas.
- Realiza varias pruebas y compara tiempos simulados.
- Organiza tu carpeta de ejercicios con nombres claros.

Recursos adicionales sugeridos

Para profundizar más puedes revisar:

- *Operating Systems Concepts* (Silberschatz) — capítulos de E/S y controladores.
- *Operating Systems: Three Easy Pieces* — capítulos de I/O.

- Documentación de Linux sobre controladores e interrupciones.

Semana 10 — Sistemas Operativos 2025

Este documento describe el contenido de la **Semana 10** del curso *Sistemas Operativos 2025*.

Incluye una lista de los materiales (teoría, prácticas y ejercicios) y una descripción ampliada de cada recurso.

Contenido

`00_objetivos_semana10.md` Documento Objetivos de aprendizaje de la semana 10.
`01_seguridad_procesos.pdf` PDF Teoría sobre seguridad y protección de procesos en sistemas operativos.
`02_control_acceso_permisos.pdf` PDF Material sobre mecanismos de control de acceso y permisos.
`03_practica_seguridad/` Carpeta Ejercicios y ejemplos prácticos relacionados con seguridad de procesos.
`04_ejercicios.ipynb` Notebook Actividades interactivas y prácticas sobre seguridad y control de acceso.
`README.md` Documento Este archivo de descripción general.

Descripción ampliada de los recursos

`00_objetivos_semana10.md`

Documento introductorio que presenta los **objetivos específicos de la semana**, tales como:

- Comprender los fundamentos de la seguridad en sistemas operativos.
- Conocer mecanismos de protección de procesos.
- Entender cómo funcionan los controles de acceso y permisos.
Recomendado leerlo primero para orientar el estudio.

`01_seguridad_procesos.pdf`

Material teórico que explica:

- Qué significa **seguridad en un sistema operativo**.
- Mecanismos de protección de procesos y aislamiento.

- Políticas de seguridad básicas.
Incluye ejemplos y prácticas de análisis teórico.



Documento que profundiza en:

- Control de acceso basado en usuarios y grupos.
- Permisos de lectura, escritura y ejecución.
- Máscaras y conjuntos de permisos.
Incluye diagramas y ejemplos reales de sistemas UNIX/Linux.



Carpeta que contiene prácticas centradas en:

- Ejercicios para modificar y observar permisos de archivos y procesos.
- Casos de prueba para cambiar propietarios o permisos de seguridad.
- Código de ejemplo o scripts relacionados con seguridad.
Incluye instrucciones para ejecución y pruebas.



Notebook interactivo que incluye:

- Ejercicios guiados sobre permisos y controles de acceso.
- Prácticas de seguridad y reflexiones automáticas.
- Actividades para verificar comprensión de conceptos.
Permite ejecutar código interactivo según el tema de la semana.

🛠️ Cómo usar este material

1. **Comienza con los objetivos** para saber qué aprender esta semana.
2. **Revisa la teoría** sobre seguridad y control de acceso.
3. **Realiza las prácticas** para aplicar los conceptos.
4. **Completa los ejercicios** del notebook para reforzar lo aprendido.

🔗 Buenas prácticas

- Mantén organizados tus ejercicios con nombres claros.
- Documenta tus observaciones en la ejecución de prácticas.
- Verifica que cambias permisos con cuidado y registra los resultados.

Recursos adicionales sugeridos

Para profundizar más en estos temas puedes consultar:

- *Operating Systems Concepts* (Silberschatz) — capítulos de seguridad y protección.
- Documentación de Linux sobre permisos y control de acceso.
- *Operating Systems: Three Easy Pieces* — capítulos relevantes.

Semana 11 — Sistemas Operativos 2025

Este documento describe el contenido de la **Semana 11** del curso *Sistemas Operativos 2025*.

Incluye una lista de los materiales (teoría, prácticas y ejercicios) y una descripción ampliada de cada recurso.

Contenido

00_objetivos_semanal11.md Documento Objetivos de aprendizaje de la semana 11.
01_redes_y_sockets.pdf PDF Teoría sobre redes, sockets y comunicación entre máquinas.
02_protocolos_red.pdf PDF Material sobre protocolos de red y modelos de comunicación.
03_practica_sockets/ Carpeta Ejercicios prácticos de programación de sockets y redes.
04_ejercicios.ipynb Notebook Actividades interactivas y simulaciones de redes.
README.md Documento Este archivo de descripción general.

Descripción ampliada de los recursos

00_objetivos_semanal11.md

Documento introductorio que presenta los **objetivos específicos de la semana**, tales como:

- Comprender los fundamentos de redes en sistemas operativos.
- Conocer cómo funcionan los **sockets** para comunicación entre procesos remotos.
- Entender protocolos básicos de red.
Recomendado leerlo primero para orientar el estudio.

01_redes_y_sockets.pdf

Material teórico que explica:

- Qué es una red de computadoras y cómo interactúa con el sistema operativo.
- Concepto de **sockets** para comunicación entre procesos remotos.
- Tipos de sockets (TCP, UDP) y su uso.
Incluye ejemplos y diagramas de comunicación cliente-servidor.



Documento que profundiza en:

- Modelos OSI y TCP/IP.
- Protocolos como HTTP, FTP, TCP, UDP y cómo se relacionan al nivel de SO.
- Encapsulamiento y direccionamiento.
Incluye casos prácticos y ejemplos de paquetes.



Carpeta que contiene prácticas centradas en:

- Ejercicios de programación de sockets en C o Python.
- Ejemplos cliente/servidor para TCP y UDP.
- Scripts para probar conexiones de red local.
Incluye instrucciones para compilar y ejecutar los ejemplos.



Notebook interactivo que incluye:

- Ejercicios guiados sobre comunicación de red.
- Simulaciones de envío y recepción de mensajes entre procesos remotos.
- Actividades automáticas para verificar comprensión.
Permite ejecutar pruebas simples de redes.

Cómo usar este material

1. **Empieza con los objetivos** para saber qué dominar esta semana.
2. **Revisa la teoría** sobre redes y comunicación.
3. **Realiza las prácticas de programación de sockets.**
4. **Completa los ejercicios** del notebook para reforzar lo aprendido.

Buenas prácticas

- Mantén tus scripts organizados y con nombres descriptivos.
- Realiza pruebas de comunicación en máquina local antes de red.

- Documenta el comportamiento observado en conexiones y errores.

Recursos adicionales sugeridos

Para profundizar más en estos temas puedes consultar:

- *Computer Networking: A Top-Down Approach* (Kurose & Ross).
- Documentación de sockets en Linux.
- Protocolos y modelos de red estándar.

Semana 12 — Sistemas Operativos 2025

Este documento describe el contenido de la **Semana 12** del curso *Sistemas Operativos 2025*.

Incluye una lista de los materiales (teoría, prácticas y ejercicios) y una descripción ampliada de cada recurso.

Contenido

00_objetivos_semana12.md Documento Objetivos de aprendizaje de la semana 12.
01_monitoreo_rendimiento.pdf PDF Teoría sobre monitoreo y análisis de rendimiento del sistema.
02_herramientas_linux.pdf PDF Material sobre herramientas de monitoreo en Linux (top, htop, vmstat, iostat).
03_practica_rendimiento/ Carpeta Ejercicios prácticos de monitoreo y análisis de rendimiento.
04_ejercicios.ipynb Notebook Actividades interactivas sobre métricas de rendimiento.
README.md Documento Este archivo de descripción general.

Descripción ampliada de los recursos

00_objetivos_semana12.md

Documento introductorio que presenta los **objetivos específicos de la semana**, tales como:

- Comprender cómo evaluar el rendimiento de un sistema operativo.
- Conocer herramientas de monitoreo disponibles en Linux.

- Interpretar métricas de CPU, memoria y E/S.
Recomendado leerlo primero para orientar el estudio.



Material teórico que explica:

- Conceptos de rendimiento en sistemas operativos.
- Métricas clave: uso de CPU, utilización de memoria, tasas de E/S, tiempos de respuesta.
- Qué medir y por qué.
Incluye ejemplos y escenarios típicos de monitoreo.



Documento que profundiza en herramientas prácticas para monitoreo:

- `top` y `htop` para ver procesos y uso de recursos.
- `vmstat` para estadísticas de memoria y procesos.
- `iostat` para estadísticas de E/S.
- Otras utilidades útiles en Linux.
Incluye ejemplos de uso y salidas de muestra.



Carpeta que contiene prácticas centradas en:

- Scripts que usan `top`, `vmstat`, `iostat` y similares.
- Ejercicios para recolectar y analizar métricas reales.
- Archivos de salida y análisis de tendencias.
Incluye instrucciones de uso paso a paso.



Notebook interactivo que incluye:

- Actividades guiadas para interpretar métricas.
- Gráficas y comparaciones de rendimiento.
- Ejercicios automáticos para verificar comprensión.
Permite ejecutar fragmentos de código y observación de resultados.

Cómo usar este material

1. **Empieza con los objetivos** para saber qué aprender esta semana.
2. **Revisa la teoría** sobre métricas y monitoreo.
3. **Realiza las prácticas de monitoreo** para aplicar los conceptos.
4. **Completa los ejercicios** del notebook para reforzar lo aprendido.

Buenas prácticas

- Documenta tus pruebas con capturas o salidas de consola.
- Compara métricas antes y después de cambios importantes.
- Anota observaciones claras para referencia futura.

Recursos adicionales sugeridos

Para profundizar más en estos temas puedes consultar:

- Guías de monitoreo de rendimiento en Linux.
- *Linux Performance* (Libro/recursos en línea).
- Documentación de herramientas como `top`, `htop`, `vmstat`, `iostat`.

Semana 13 — Sistemas Operativos 2025

Este documento describe el contenido de la **Semana 13** del curso *Sistemas Operativos 2025*.

Incluye una lista de los materiales (teoría, prácticas y ejercicios) y una descripción ampliada de cada recurso.

Contenido

`00_objetivos_semana13.md` Documento Objetivos de aprendizaje de la semana 13.

`01_backup_recuperacion.pdf` PDF Teoría sobre técnicas de respaldo (backup) y recuperación de datos.

`02_restauracion_sistemas.pdf` PDF Material sobre estrategias de restauración y recuperación ante fallas.

`03_practica_backup/` Carpeta Ejercicios prácticos relacionados con backup y recuperación de sistemas.

`04_ejercicios.ipynb` Notebook Actividades interactivas sobre técnicas de respaldo y

restauración.

README.md Documento Este archivo de descripción general.

Descripción ampliada de los recursos



Documento introductorio que presenta los **objetivos específicos de la semana**, tales como:

- Comprender los conceptos y la importancia de **respaldo de datos**.
- Aprender estrategias de recuperación ante fallas.
- Diferenciar tipos de backup: completo, incremental, diferencial.
Recomendado leerlo primero para orientar el estudio.



Material teórico que explica:

- Qué es un sistema de respaldo de datos.
- Tipos de backup: completo, incremental y diferencial.
- Ventajas y desventajas de cada enfoque.
Incluye ejemplos y consideraciones de aplicación práctica.



Documento que profundiza en:

- Métodos de restauración de sistemas y datos.
- Estrategias ante fallas del sistema o corrupción de datos.
- Pasos para recuperar un sistema a un estado anterior.
Incluye casos de estudio y escenarios reales.



Carpeta que contiene ejercicios prácticos como:

- Scripts para realizar backups automáticos.
- Ejercicios de restauración de un sistema o directorios.
- Casos de prueba para verificar integridad de respaldos.
Incluye instrucciones para ejecutar y validar resultados.



Notebook interactivo que incluye:

- Actividades guiadas sobre backup y recuperación.
- Ejercicios de análisis de estrategias de restauración.
- Evaluaciones automáticas para verificar comprensión.
Permite ejecutar segmentos de código o simulaciones.

Cómo usar este material

1. **Comienza con los objetivos** para saber qué aprender esta semana.
2. **Revisa la teoría** sobre backup y recuperación de datos.
3. **Realiza las prácticas** para aplicar las estrategias.
4. **Completa los ejercicios** del notebook para reforzar lo aprendido.

Buenas prácticas

- Documenta los comandos y resultados de tus respaldos.
- Prueba restaurar datos en entornos controlados antes de aplicaciones reales.
- Mantén versiones de respaldo con fechas claras.

Recursos adicionales sugeridos

Para profundizar más en estos temas puedes consultar:

- Documentación y guías de respaldo en Linux.
- Herramientas estándar como `rsync`, `tar`, `dd` y otras utilidades.
- Recursos en línea sobre estrategias de recuperación ante desastres.

Semana 14 — Sistemas Operativos 2025

Este documento describe el contenido de la **Semana 14** del curso *Sistemas Operativos 2025*.

Incluye una lista de los materiales (teoría, prácticas y ejercicios) y una descripción ampliada de cada recurso.

Contenido

00_objetivos_semana14.md Documento Objetivos de aprendizaje de la semana 14.
01_finalizacion_curso.pdf PDF Teoría general de revisión y cierre de temas del curso.
02_proyecto_integrador.pdf PDF Descripción del proyecto integrador de curso.
03_practica_proyecto/ Carpeta Recursos y actividades prácticas para el proyecto integrador.
04_ejercicios.ipynb Notebook Actividades interactivas relacionadas con el proyecto.
README.md Documento Este archivo de descripción general.

Descripción ampliada de los recursos

00_objetivos_semana14.md

Documento introductorio que presenta los **objetivos específicos de la semana**, tales como:

- Revisar los conceptos clave vistos a lo largo del curso.
- Comprender las expectativas y entregables del proyecto integrador.
- Preparar estrategias de presentación y defensa de proyectos.
Recomendado leerlo primero para orientar el estudio final.

01_finalizacion_curso.pdf

Material teórico de repaso que cubre:

- Conceptos principales de sistemas operativos revisados en semanas anteriores.
- Resumen de temas como procesos, memoria, E/S, IPC, seguridad y rendimiento.
- Casos de aplicación en entornos reales.
Incluye diagramas y resúmenes para facilitar el repaso.

02_proyecto_integrador.pdf

Documento que describe el **proyecto integrador** del curso:

- Objetivos claros del proyecto.
- Requerimientos técnicos y funcionales.
- Criterios de evaluación.
- Ejemplos de proyectos anteriores (si aplica).
Incluye fechas de entrega y especificaciones detalladas.

03_practica_proyecto/

Carpeta que contiene recursos y actividades de apoyo al proyecto, como:

- Listado de prácticas sugeridas para cumplir con el proyecto.
- Código base o plantillas de ejemplo.
- Notas y archivos auxiliares para el desarrollo.
Incluye instrucciones y sugerencias de ejecución.



Notebook interactivo que incluye:

- Actividades sugeridas para apoyar el proyecto integrador.
- Ejercicios de reflexión y análisis.
- Plantillas y ejemplos automáticos.
Permite ejecutar fragmentos de código y ver resultados según el proyecto.

Cómo usar este material

1. **Comienza con los objetivos** para saber qué se espera en la última etapa.
2. **Revisa el contenido de repaso** para consolidar tu comprensión.
3. **Lee la documentación del proyecto** para entender los requerimientos.
4. **Realiza las prácticas sugeridas** para completar el proyecto integrador.
5. **Ejecuta las actividades** del notebook para practicar conceptos relevantes.

Buenas prácticas

- Organiza tu proyecto con carpetas claras y documentadas.
- Mantén un registro de tus avances con fechas y tareas.
- Documenta tu código y decisiones de diseño.

Recursos adicionales sugeridos

Para apoyar el cierre del curso puedes consultar:

- Apuntes y resúmenes de cada semana del curso.
- Ejemplos de proyectos integradores de sistemas operativos.
- Tutoriales de herramientas que uses en el proyecto (compiladores, GIT, etc.).

Semana 15 — Sistemas Operativos 2025

Este documento describe el contenido de la **Semana 15** del curso *Sistemas Operativos 2025*.

Incluye una lista de materiales (teoría, prácticas y ejercicios) y una descripción ampliada de cada recurso.

Contenido

00_objetivos_semana15.md Documento Objetivos de aprendizaje de la semana 15.
01_seguridad_sistemas_operativos.pdf PDF Teoría sobre seguridad avanzada en sistemas operativos.
02_privilegios_acceso.pdf PDF Material sobre mecanismos de privilegios y control de acceso.
03_practica_seguridad_sistemas/ Carpeta Ejercicios prácticos de seguridad y protección de procesos.
04_ejercicios.ipynb Notebook Actividades interactivas sobre seguridad y análisis de vulnerabilidades.
README.md Documento Este archivo de descripción general.

Descripción ampliada de los recursos

00_objetivos_semana15.md

Documento introductorio que presenta los **objetivos específicos de la semana**, tales como:

- Comprender los conceptos avanzados de seguridad en sistemas operativos.
- Identificar mecanismos de control de acceso a nivel de kernel y usuario.
- Analizar políticas de auditoría y protección de recursos.
Recomendado leerlo primero para orientar el estudio.

01_seguridad_sistemas_operativos.pdf

Material teórico que cubre:

- Modelos de seguridad en sistemas operativos.
- Protección de memoria y recursos compartidos.
- Política de acceso y auditoría.
Incluye ejemplos y casos de estudio.

02_privilegios_acceso.pdf

Documento que explica:

- Privilegios de procesos y usuarios.
- Control de acceso basado en roles.

- Listas de control de acceso (ACL).
Incluye ejemplos prácticos de configuración.



Carpeta que contiene ejercicios prácticos como:

- Análisis de permisos avanzados.
- Scripts para comprobar accesos y privilegios.
- Casos de prueba para evaluar políticas de seguridad.
Incluye instrucciones de ejecución detalladas.



Notebook interactivo con ejercicios sobre:

- Detección de vulnerabilidades.
- Prácticas de control de acceso.
- Evaluación de políticas de seguridad.
Permite ejecutar fragmentos de código para análisis automático.

Cómo usar este material

1. **Empieza con los objetivos** para saber qué aprender esta semana.
2. **Revisa la teoría** para comprender los mecanismos de seguridad.
3. **Realiza las prácticas** para aplicar los conceptos.
4. **Completa los ejercicios** del notebook para reforzar lo aprendido.

Buenas prácticas

- Documenta tus pruebas de seguridad.
- Realiza auditorías periódicas.
- Compara resultados entre diferentes políticas.

Recursos adicionales sugeridos

Puedes consultar:

- Documentación de seguridad en Linux.
- Material de hardening de sistemas operativos.

Semana 16 — Sistemas Operativos 2025

Este documento describe el contenido de la **Semana 16** del curso *Sistemas Operativos 2025*.

Incluye una lista de materiales (teoría, prácticas y ejercicios) y una descripción ampliada de cada recurso.

Contenido

`00_objetivos_semana16.md` Documento Objetivos de aprendizaje de la semana 16.

`01_tendencias_sistemas_operativos.pdf` PDF Teoría sobre tendencias actuales y futuras en sistemas operativos.

`02_virtualizacion_containers.pdf` PDF Material sobre virtualización, contenedores y microservicios.

`03_practica_virtualizacion/` Carpeta Ejercicios prácticos de virtualización y contenedores.

`04_ejercicios.ipynb` Notebook Actividades interactivas sobre tecnologías emergentes.

`README.md` Documento Este archivo de descripción general.

Descripción ampliada de los recursos

`00_objetivos_semana16.md`

Documento introductorio que presenta los **objetivos específicos de la semana**, tales como:

- Entender las tendencias modernas en diseño de sistemas operativos.
- Conocer la virtualización y el uso de contenedores.
- Analizar diferencias entre máquinas virtuales y contenedores.
Recomendado leerlo primero para orientar el estudio.

`01_tendencias_sistemas_operativos.pdf`

Material teórico que explica:

- Evolución de sistemas operativos modernos.
- Enfoques de seguridad y rendimiento en SO actuales.
- Integración con tecnologías de nube y servicios distribuidos.
Incluye comparativas y casos de uso.



Documento que cubre:

- Qué es la virtualización y su importancia.
- Diferencias entre máquinas virtuales y contenedores.
- Tecnologías como Docker, Kubernetes y hypervisores.
Incluye diagramas y ejemplos prácticos.



Carpeta que contiene ejercicios prácticos como:

- Scripts para levantar contenedores.
- Ejemplos de uso de Docker y herramientas afines.
- Casos de prueba para comparar rendimiento entre entornos.
Incluye instrucciones de ejecución paso a paso.



Notebook interactivo con actividades sobre:

- Comparativas de uso de recursos.
- Simulación de despliegue de contenedores.
- Evaluación automática de tareas.
Permite ejecutar código interactivo relacionado con tecnologías emergentes.

🛠️ Cómo usar este material

1. **Empieza con los objetivos** para saber qué se espera de la semana.
2. **Lee la teoría** sobre tendencias modernas y virtualización.
3. **Realiza las prácticas** para observar tecnologías en acción.
4. **Completa los ejercicios** del notebook para reforzar tu comprensión.

📝 Buenas prácticas

- Documenta tus pruebas con contenedores.
- Prueba múltiples configuraciones para comparar resultados.
- Mantén versiones claras de tus scripts y pruebas.



Recursos adicionales sugeridos

Puedes consultar:

- Documentación oficial de Docker y Kubernetes.
- Recursos de cloud computing relacionados con SO.
- Artículos sobre tendencias y evolución futura de sistemas operativos.