



UNIVERSIDAD DE LAS PALMAS  
DE GRAN CANARIA

GUÍA DOCENTE

CURSO: 2023/24

## 40970 - FUNDAMENTOS DE LOS SISTEMAS OPERATIVOS

**CENTRO:** 180 - Escuela de Ingeniería Informática

**TITULACIÓN:** 4008 - Grado en Ingeniería Informática

**ASIGNATURA:** 40970 - FUNDAMENTOS DE LOS SISTEMAS OPERATIVOS

**CÓDIGO UNESCO:**                      **TIPO:** Obligatoria                      **CURSO:** 2                      **SEMESTRE:** 2º semestre

**CRÉDITOS ECTS:** 6                      **Especificar créditos de cada lengua:**                      **ESPAÑOL:** 6                      **INGLÉS:**

### SUMMARY

The operating system is an essential component in every computer system. It provides a basic environment for programming and user interaction which enables people to effectively use the computer. Moreover, the operating system manages the system resources so that they can be accessed with a reasonable balance of efficiency, security and fairness.

It is crucial that a graduate in Computer Science or Computer Engineering knows the fundamental concepts related to the operating system functions, structure and implementation.

By acquiring knowledge of the operating system's internals, the student will have an opportunity to learn how software technologies are integrated with the computer's physical architecture, and what kinds of challenges and tradeoffs have to be addressed when designing resource management strategies.

### REQUISITOS PREVIOS

Habilidades profesionales para ingenieros (40955)

Estructura de Computadores (40959)

Periféricos e Interfaces (40964)

Tecnologías de Programación (40963)

Aclaraciones sobre las competencias específicas que son requisitos:

- Nivel intermedio (B1) de comprensión lectora en inglés.
- Conocimiento de la estructura de un computador: arquitectura de von Neumann, jerarquía de memorias, interrupciones.
- Dominio básico de las técnicas de programación estructurada.
- Nociones de programación concurrente.

### Plan de Enseñanza (Plan de trabajo del profesorado)

### Contribución de la asignatura al perfil profesional:

El sistema operativo es una parte esencial dentro de un sistema informático. Administra los recursos del sistema y ofrece un entorno de programación y operación a los usuarios, permitiendo trabajar con la máquina de forma cómoda y eficaz.

Es imprescindible que un titulado universitario en Informática conozca los conceptos fundamentales vinculados a los sistemas operativos, sus funciones, su estructura y su implementación.

Para un ingeniero en Informática, conocer el funcionamiento interno del sistema operativo es una

oportunidad de entender cómo se integran las tecnologías software con la arquitectura física del computador. También le ayudará a tomar mejores decisiones en múltiples aspectos de su vida profesional, relativas al tipo o cantidad de recursos que debe dimensionar en un sistema de información; o a diagnosticar y corregir problemas en el funcionamiento de los sistemas.

Por último, las políticas y mecanismos de gestión de recursos como el procesador o la memoria tienen una utilidad más allá del sistema operativo, pues sirven para entender la administración de un sistema complejo, que exige un difícil equilibrio entre el rendimiento, la seguridad, la usabilidad y la equidad en el reparto de los recursos.

## Competencias que tiene asignadas:

CB2, CB5, G4, G8, N2, CI5, CI10, CI14

## Objetivos:

Ob1. Adquirir conocimiento cualificado sobre la estructura general, funciones y principios de diseño de los sistemas operativos.

Ob2. Entender y aplicar las políticas de gestión de recursos propias del sistema operativo.

Ob3. Adiestrarse en el manejo del entorno Unix/Linux con su interfaz de consola (CLI).

Ob4. Adquirir competencia en programación de sistemas, mediante el lenguaje C y la API del sistema operativo Unix.

## Contenidos:

La asignatura tiene cinco unidades teóricas y tres unidades prácticas.

Competencias: todos los contenidos contribuyen a las competencias CB2, CB5, G4, G8, N2, CI5, CI10.

»»

»» CONTENIDOS TEÓRICOS

»»

Bibliografía básica: [Silberschatz2019]

Bibliografía complementaria: [ULPGC2007]

### 1. CONCEPTOS SOBRE SISTEMAS OPERATIVOS

- El sistema operativo como interfaz: API, llamadas al sistema, CLI, GUI, recursos virtuales.
- El sistema operativo como administrador de recursos: procesos y recursos; políticas de gestión; objetivos de eficiencia, seguridad y equidad; interbloqueo e inanición.
- Tipos de sistemas informáticos: procesamiento por lotes, tiempo compartido, sistemas empujados, sistemas distribuidos.
- Arquitectura del hardware para el SO: sistema de interrupciones, excepciones, modo dual de operación, jerarquía de memorias, mecanismos de protección.
- Componentes del sistema operativo: programas del sistema, núcleo, shell; arquitecturas de núcleos; máquinas virtuales.

### 2. PROCESOS

- Sistemas multitarea: estados de un proceso; colas; cambio de contexto; el planificador.
- Servicios: procesos pesados e hilos; llamadas al sistema usuales.
- Planificación del procesador: políticas básicas (FCFS, SJF, Round Robin, prioridades); evaluación de las políticas; métodos multicolos; multiprocesadores.

### 3. CONCURRENCIA

- Programación concurrente: especificación algorítmica; beneficios para el programador.

- Propiedades de los sistemas concurrentes: no determinismo; necesidad de sincronización; acciones atómicas.
  - El problema de la sección crítica.
  - Sincronización: problemas clásicos; mutex y variables condición; técnicas algorítmicas de solución de problemas de sincronización. El problema del interbloqueo.
- Competencia específica: CI14

#### 4. MEMORIA

- Elementos comunes: fragmentación; gestión del espacio libre; protección; espacios de direcciones lógicas y físicos; traducción de direcciones y MMU.
- Gestión de memoria contigua: enlace dinámico; políticas de gestión del espacio libre; intercambio (swapping).
- Gestión de memoria no contigua: memoria paginada; TLB; compartición y protección de zonas de memoria.
- Memoria virtual: memoria virtual paginada; paginación bajo demanda; hardware de apoyo a la memoria virtual; algoritmos de reemplazo (FIFO, óptimo, LRU, LFU, aproximaciones mediante el bit de referencia); teoría del conjunto de trabajo (working set).

#### 5. ARCHIVOS

- Interfaz del sistema: estructura y atributos de los archivos; operaciones sobre archivos; métodos de acceso (secuencial, directo, indexado); organización de directorios; montaje de sistemas de archivos; protección.
- Implementación del sistema: organización del disco para un sistema de archivos; políticas de asignación de espacio a archivos (contigua, enlazada, FAT, indexada); implementación de directorios.

»»

#### »» CONTENIDOS PRÁCTICOS

»»

Bibliografía básica: [Schildt2003], [Love2013]

##### 1. Operación en entorno Unix/Linux

- Estructura y servicios de los sistemas Unix
- Uso del shell
- Programas de manejo de archivos y procesos
- Permisos de acceso a archivos

##### 2. Programación en lenguaje C

- Sentencias, variables, funciones
- Punteros, cadenas de texto, memoria dinámica
- Funciones de la biblioteca estándar

##### 3. Programación de sistemas en Unix/Linux

- Llamadas al sistema para gestión de archivos
- Llamadas al sistema para gestión de procesos
- Programación con hilos (pthreads)

Competencia específica: CI14

Al tratarse de una asignatura introductoria, se hará énfasis en la adquisición de conceptos claves de la materia (sistemas operativos), de manera que el estudiante:

- a) Se familiarice con la terminología;
- b) Entienda cuáles han sido las motivaciones y objetivos que han llevado a la industria informática a desarrollar los componentes habituales de un sistema operativo y
- c) Pueda utilizar los conceptos en situaciones realistas de su vida profesional.

Por otro lado, se pretende que el estudiante adquiera un nivel de competencia básico en la programación de sistemas. La mejor forma de lograrlo es a través de la práctica directa. Por tanto el estudiante debe recibir un adiestramiento en esas herramientas y embarcarse en proyectos de programación. A su vez, estos proyectos guardarán una relación estrecha con los contenidos teóricos de la asignatura, de manera que se reforzará el aprendizaje de esos contenidos mediante su aplicación práctica en el desarrollo de software de sistemas.

Las actividades formativas que desarrollan este plan son las habituales en la enseñanza universitaria: sesiones teóricas, sesiones en laboratorio y proyectos prácticos. El profesor actúa como expositor del contenido, como guía y como supervisor. La adquisición de competencias se certifica mediante pruebas formales (exámenes y revisión de entregas de proyectos).

Estas son las actividades formativas:

### »»» AF1. CLASES DE TEORÍA.

(según la guía docente, «sesiones académicas de fundamentación»)

Se desarrollarán en el aula. En ellas el profesor explicará los fundamentos sobre sistemas operativos descritos en el temario. En estas sesiones los profesores emplearán diferentes técnicas para suscitar el aprendizaje de los estudiantes, tales como:

- Exposición de tópicos y conceptos claves.
- Presentación de problemas, retos o casos de estudio.
- Presentación y resolución de ejercicios.
- Microevaluaciones de repaso.

Las sesiones tratarán de contribuir al aprendizaje personal del estudiante, quien deberá utilizar la bibliografía básica para estudiar, analizar y reflexionar sobre los contenidos teóricos. El profesor actuará con un rol de orientador y facilitador de este proceso.

### »»» AF2. CLASES DE PRÁCTICAS.

(según la guía docente, «sesiones académicas de aplicación»)

Estas sesiones estarán centradas en la programación de sistemas en entornos UNIX/Linux. En estas sesiones se desarrollarán actividades de alguno de estos tres tipos:

- Entrenamiento sobre herramientas de programación de sistemas.
- Demostraciones sobre la aplicación práctica de conceptos de la asignatura.
- Presentación y seguimiento de las actividades prácticas evaluables.

Es muy importante que el estudiante asista regularmente a estas sesiones, sobre todo si tiene poca experiencia como programador.

### »»» AF3. TRABAJOS PRÁCTICOS DE PROGRAMACIÓN.

(según la guía básica, «trabajos»)

Consistirán en pequeños proyectos de programación en lenguaje C sobre Linux, utilizando la API de este sistema operativo. También se propondrán actividades para adiestrarse con la CLI de Unix/Linux y otras herramientas de programación de sistemas.

Los proyectos de programación tratarán sobre los tópicos claves en la asignatura. Algunos de estos trabajos contribuirán a la calificación final.

El profesor presentará los proyectos en las sesiones de laboratorio y los estudiantes los desarrollarán de forma parcialmente supervisada, con parte del trabajo en el laboratorio y otra gran parte en el horario que el estudiante considere oportuno.

Para el desarrollo de estas prácticas, el estudiante contará con un material didáctico diseñado para que ella/él pueda realizar el trabajo por su cuenta, mediante una guía bien clara de los tópicos que debe aprender y una lista de tareas previas de autoaprendizaje.

#### »»» AF4. ACTIVIDADES EN LÍNEA.

(según la guía básica, «trabajos»)

Como refuerzo de las unidades teóricas, se establecerán actividades en línea dentro del Campus Virtual, con el fin de que los estudiantes trabajen y refuercen contenidos y competencias. Estas actividades tendrán normalmente alguno de estos formatos:

- Resolver un problema basado en un caso de estudio o supuesto.
- Reflexionar o profundizar sobre algún tópico de la asignatura.
- Ejercicios sobre algoritmos de gestión de recursos.
- Reto avanzado sobre algún tópico de la asignatura.

#### »»» AF5. SESIONES DE TUTORIZACIÓN.

Esta actividad se realiza normalmente de forma presencial junto con el profesor y a demanda del estudiante. Se mantiene una relación más directa profesor/estudiante y la tutorización está individualizada y adaptada a las características del estudiante. También existe la opción de la tutoría telemática, a través del Campus Virtual o de otros medios de mensajería.

#### »»» AF6. EXÁMENES.

Esta actividad se realiza en varias modalidades:

- Exámenes de teoría sumativos: el estudiante demuestra la adquisición de las competencias sobre los fundamentos de los sistemas operativos y recibe una calificación.
- Exámenes prácticos sumativos: el estudiante demuestra su aptitud en las competencias de programación de sistemas y recibe una calificación.
- Exámenes de adiestramiento: pruebas voluntarias y formativas para que los estudiantes puedan evaluar su nivel de competencia en algún tópico de la asignatura, o para adiestrarse en el estilo y formato de los exámenes sumativos.

Los exámenes teóricos podrán contener preguntas sobre cualquier ámbito de la asignatura, incluyendo conceptos sobre las herramientas utilizadas en la parte práctica.

### **Evaluación:**

Criterios de evaluación

-----

»» Fuentes de evaluación

-----

Las fuentes de evaluación usadas en la calificación sumativa son estas cuatro:

FE1. Exámenes teóricos (actividad AF6)

FE2. Trabajos prácticos (actividad AF3)

FE3. Exámenes prácticos (actividad AF6)

FE4. Actividades de evaluación formativa (actividades AF1, AF4, AF6)

#### »» FE1. EXÁMENES TEÓRICOS (actividad AF6)

Para la calificación de la teoría de la asignatura se realizarán estos exámenes sumativos:

- Examen parcial. Se realizará a mitad del semestre y abarcará aproximadamente los tres primeros temas de la teoría.
- Examen final (ordinaria). El examen de la convocatoria ORDINARIA tendrá dos partes: la primera parte tendrá el mismo temario del examen parcial y la segunda parte con el resto del temario (aunque podrán evaluarse contenidos de la totalidad de la asignatura).
- Otras convocatorias. Los exámenes de las convocatorias EXTRAORDINARIA y ESPECIAL consistirán en una prueba única que abarcará todo el temario de la asignatura.

En el examen final (ordinaria), el estudiante puede presentarse por separado a cualquiera de las dos partes del examen, según mejor le convenga.

#### »» FE2. TRABAJOS PRÁCTICOS (actividad AF3)

- Estos trabajos se realizarán como parte del sistema de evaluación continua durante el periodo lectivo.
- Habrá al menos tres trabajos prácticos que deberán entregarse a los profesores para su evaluación.
- Algunos trabajos serán obligatorios y otros serán optativos.
- Cada trabajo tendrá un plazo de entrega establecido para la evaluación continua.
- El trabajo se realizará de forma individual o en equipo, según lo indique el profesor.
- Para cada trabajo, el profesor recabará evidencias del resultado y también del proceso de desarrollo. Podrá requerirse una entrevista oral si el profesor lo estima conveniente.
- Para favorecer la evaluación continua, siempre que la entrega supere un cierto umbral de calidad, el profesor informará de los posibles puntos de mejora y el/la estudiante tendrá la oportunidad de enviar una nueva versión de su trabajo para una segunda evaluación.

#### »» FE3. EXAMENES PRÁCTICOS (actividad AF6)

Se realizarán pruebas individuales para verificar que el estudiante ha adquirido las competencias en programación de sistemas que se han trabajado en las prácticas de la asignatura.

- Estos exámenes prácticos se realizarán junto con los exámenes teóricos (FE1), es decir, en el examen parcial y en los exámenes de convocatoria.
- El examen práctico parcial evaluará las competencias prácticas trabajadas hasta esa fecha, mientras que el examen práctico final de la convocatoria ordinaria hará una evaluación general de todas las competencias prácticas de la asignatura.
- En las convocatorias EXTRAORDINARIA y ESPECIAL se realizará un examen práctico general, que será más profundo y amplio que los exámenes del periodo ordinario y que podrá incluir un examen oral individualizado. La extensión y el formato de la prueba podrán variar para cada estudiante, en función de las evidencias previas sobre los contenidos prácticos que esa persona haya acreditado en convocatorias anteriores.

#### »» FE4. ACTIVIDADES DE EVALUACIÓN FORMATIVA (actividades AF1, AF4, AF6)

- A lo largo del periodo lectivo se irán proponiendo ejercicios y tareas que desarrollarán los tópicos de cada unidad teórica (actividad tipo AF4). También se podrán plantear ensayos de examen o tests de autoevaluación (AF6), así como pequeñas evaluaciones de repaso durante las clases de teoría (AF1).
- Algunas de estas tareas serán supervisadas por el profesor, para valorar el progreso del estudiante y tomar acciones de mejora del aprendizaje. De esta manera se desarrolla un proceso de evaluación continua.
- Cada actividad AF4 o AF6 tendrá un plazo de realización. Las actividades entregadas fuera de su plazo no computarán en la calificación.
- Para que estas actividades puedan computar en la calificación, el estudiante deberá realizar una cantidad mínima, no inferior al 50% del total de actividades planteado en el semestre.

## »» Criterios de evaluación

---

Para diseñar los exámenes y trabajos, así como para elaborar las rúbricas de los trabajos prácticos y para valorar el progreso del estudiante, se utilizarán varios criterios de evaluación.

NOTAS:

- Entre (paréntesis), las fuentes de evaluación vinculadas a cada criterio.
- Entre [corchetes], las competencias oficiales con las que está vinculado cada criterio.

LISTA DE CRITERIOS:

- Dominio de los tópicos de los contenidos teóricos (FE1,FE4) [CB2,CI5,CI10,CI14]
- Capacidad de aplicar los contenidos teóricos de la asignatura a escenarios profesionales y sociales (FE1,FE4) [CB5,G4,G8]
- Capacidad para especificar y construir programas de sistemas en lenguaje C (FE2,FE3) [CB5,CI5]
- Capacidad para programar siguiendo un conjunto de buenas prácticas de codificación (FE3) [CB2,G8]
- Capacidad para resolver algoritmos concurrentes que contengan problemas de sincronización (FE1, FE2, FE3, FE4) [CI14]
- Expresión oral (FE3) [CB2,N2]
- Expresión escrita (FE1, FE3, FE4) [CB2]
- Capacidad de trabajo en equipo (FE3) [N2]

Sistemas de evaluación

---

La asignatura utiliza principalmente un sistema de evaluación sumativo. Las actividades permiten una cierta evaluación formativa durante el curso. La componente sumativa viene sobre todo de los exámenes y los trabajos prácticos, mientras que la parte formativa se trabaja con las actividades de evaluación continua y las sesiones presenciales.

Criterios de calificación

---

Este apartado define cómo se puntúan las fuentes de evaluación para dar lugar a la calificación final de la asignatura.

### » CALIFICACIÓN GENERAL

La calificación se obtendrá a partir de estos dos componentes:

- Teoría y participación activa (TEO, puntuado de 0 a 12).
- Prácticas (PRA, puntuado de 0 a 4).

La calificación bruta será la suma ponderada de TEO y PRA:

$$0,75 \times (\text{TEO} + \text{PRA})$$

Sobre esta calificación bruta se aplicarán estas restricciones:

- Hay que obtener al menos 6 puntos en TEO.
- Hay que obtener al menos 2 puntos en PRA.

Si no se cumple alguna de estas restricciones, la asignatura estará suspendida, con una calificación máxima de 4 puntos.

Si el resultado bruto es mayor que 10, la calificación definitiva será de 10 puntos.

## »»» TEO. TEORÍA

### a) Convocatoria ORDINARIA:

La nota de teoría (TEO) se obtendrá con esta fórmula:

$$\text{TEO} = \text{EXA1} + \text{EXA2} + \text{PAC}$$

Donde:

- EXA1 es la calificación del examen de la primera parte (de 0 a 5 puntos, ver nota a.2).
- EXA2 es la calificación del examen de la segunda parte (de 0 a 5 puntos).
- PAC es la calificación por participación activa (de 0 a 2 puntos, ver notas a.3 y a.4).

Para superar la asignatura, TEO debe valer al menos 6 puntos.

EXA1 y EXA2.

- a.1. Cada examen de la convocatoria ordinaria (parcial, primera y segunda parte de la ordinaria) se puntuará entre 0 y 5.
- a.2. La nota EXA1 es la mayor de las notas obtenidas en el examen parcial y en la primera parte de la convocatoria ORDINARIA.

PAC. PARTICIPACIÓN ACTIVA.

- a.3. El estudiante obtendrá una puntuación PAC según su registro personal de participación en las actividades formativas realizadas durante el periodo de evaluación continua (actividades tipo AF4), de acuerdo con esta fórmula:
  - Si el estudiante ha realizado menos del 50% del total de actividades propuestas, obtendrá cero puntos.
  - Si el estudiante ha realizado entre el 50% y el 75% del total de actividades, obtendrá un punto.
  - Si el estudiante ha realizado más del 75% del total de actividades, obtendrá dos puntos.
- a.4. La calificación definitiva del apartado PAC quedará fijada al final del periodo de evaluación continua y se mantendrá intacta en todas las convocatorias.

### b) Convocatoria EXTRAORDINARIA:

- b.1. La convocatoria tendrá un único examen de teoría, que puntuará entre 0 y 10 (EXA\_EXTR).
- b.2. Si el estudiante se presenta al examen y  $\text{EXA\_EXTR} > \text{EXA1} + \text{EXA2}$ , entonces se recalculará TEO como  $\text{EXA\_EXTR} + \text{PAC}$ . En cualquier otro caso, se mantendrá la nota TEO de la convocatoria ordinaria.

### c) Convocatoria ESPECIAL:

- c.1. La convocatoria tendrá un único examen de teoría, que puntuará entre 0 y 10 (EXA\_ESP).
- c.2. Si el estudiante se presenta al examen y  $\text{EXA\_ESP} > \text{MAX}\{\text{EXA\_EXTR}, \text{EXA1} + \text{EXA2}\}$ , entonces se recalculará TEO como  $\text{EXA\_ESP} + \text{PAC}$ . En cualquier otro caso, se mantendrá la nota TEO de la anterior convocatoria calificada.

## »»» PRA. PRÁCTICAS

### a) Convocatoria ORDINARIA:

- p.1. Cada trabajo práctico obligatorio será calificado como «superado» o «no superado».
- p.2. Para aprobar el apartado PRA hay que superar todos los trabajos prácticos obligatorios.
- p.3. El examen práctico parcial otorgará hasta 1 punto al apartado PRA. El examen práctico final otorgará hasta 2 puntos.

p.4. Los trabajos prácticos (obligatorios u optativos) tendrán asociadas tareas opcionales y avanzadas, llamadas «retos», que contribuirán en conjunto con hasta 1 punto al apartado PRA.

p.5. Los puntos de PRA (p.3, p.4) solo se aplicarán si se cumple el requisito p.2. En caso contrario se recibirán cero puntos en PRA.

b) Convocatorias EXTRAORDINARIA y ESPECIAL:

p.6. Las personas que aprueban el apartado PRA en una convocatoria conservan su calificación en las siguientes.

p.7. En las convocatorias EXTRAORDINARIA y ESPECIAL, el estudiante que no tenga ya superada la parte práctica deberá realizar un examen práctico. Este examen será puntuado de 0 a 3 puntos y será la fuente única para la calificación PRA en estas convocatorias.

## **Plan de Aprendizaje (Plan de trabajo de cada estudiante)**

### **Tareas y actividades que realizará según distintos contextos profesionales (científico, profesional, institucional, social)**

El contexto dominante en esta asignatura será el denominado «profesional». Todas las actividades descritas tienen que ver con este contexto.

#### »»» TEORÍA »»»

Cada unidad teórica (cinco en total) tendrá asociado un bloque de actividades, según lo descrito en el apartado de «metodología»: sesiones teóricas en aula, ejercicios asociados al tema y actividades en línea en el Moodle. El estudiante tendrá que emplear tiempo propio para desarrollar su aprendizaje, con actividades tales como:

- Estudio del material didáctico (libros y esquemas)
- Resolución de ejercicios, controles o ensayos de examen
- Reflexión y resolución de los retos planteados en Moodle

El tiempo recomendable para estas actividades autónomas es más o menos proporcional a las horas dedicadas a cada tema en las sesiones teóricas.

#### »»» PRÁCTICA »»»

El plan de trabajo para las actividades prácticas es el siguiente:

TP1. Uso de la CLI de UNIX/Linux.

TP2. Uso de herramientas de desarrollo en lenguaje C y UNIX/Linux.

TP3. Trabajar con procesos pesados.

TP4. Trabajar con la API de ficheros de UNIX/Linux.

TP5. Trabajar con procesos ligeros (hilos) y programación concurrente.

- Se ha procurado que los requisitos técnicos del puesto de trabajo y del software permitan la ejecución de los trabajos con un equipo informático convencional sin depender de las instalaciones de la ULPGC, por ejemplo en el ordenador personal del estudiante. En este sentido, se utiliza software libre y contenidos abiertos que no incurren en costes económicos adicionales.

**GRUPOS DE PRÁCTICAS.** Los estudiantes tendrán que respetar escrupulosamente la asignación oficial a grupos de prácticas y asistir a las sesiones que le corresponden a su grupo.

## Temporalización semanal de tareas y actividades (distribución de tiempos en distintas actividades y en presencialidad - no presencialidad)

El tiempo de trabajo del estudiante dependerá de su perfil personal y no puede generalizarse, aunque se intentará favorecer que la distribución de esfuerzo sea uniforme a lo largo del semestre, según esta semana tipo:

- 2 horas de sesiones teóricas
- 2 horas de sesiones de laboratorio
- 6 horas de trabajo personal

La temporalización prevista para la asignatura es la siguiente:

- Semana 1. Tema 1 (conceptos generales) | Práctica 1 (operatoria Linux)
- Semana 2. Tema 1 (conceptos generales) | Práctica 1 (operatoria Linux)
- Semana 3. Tema 1 (conceptos generales) | Práctica 2 (lenguaje C) \*
- Semana 4. Tema 1 (conceptos generales) | Práctica 2 (lenguaje C)
- Semana 5. Tema 2 (procesos) | Práctica 3 (procesos) \*
- Semana 6. Tema 2 (procesos) | Práctica 3 (procesos)
- Semana 7. Tema 3 (conurrencia) | Práctica 4 (archivos) \*
- Semana 8. Tema 3 (conurrencia) | Práctica 4 (archivos)
- Semana 9. Tema 3 (conurrencia) | Práctica 4 (archivos)
- Semana 10. EXAMEN PARCIAL + repaso de temas anteriores
- Semana 11. Tema 4 (memoria) | Práctica 5 (conurrencia con hilos)\*
- Semana 12. Tema 4 (memoria) | Práctica 5 (conurrencia con hilos)
- Semana 13. Tema 4 (memoria) | Práctica 5 (conurrencia con hilos)
- Semana 14. Tema 5 (archivos) | Práctica 5 (conurrencia con hilos)
- Semana 15. Tema 5 (archivos) | Revisiones finales de prácticas, seminarios o ensayos de examen

Las prácticas marcadas con un asterisco (\*) tienen entregas evaluables.

Al comienzo del semestre se distribuirá a los estudiantes un calendario de trabajo, con las fechas propuestas para las entregas de trabajos, cuáles de ellos son obligatorios y los exámenes teóricos y prácticos.

## Recursos que tendrá que utilizar adecuadamente en cada uno de los contextos profesionales.

Directorio de recursos:

Re1: Aula para la teoría (en el Edificio de Informática y Matemáticas).

Re2: Laboratorio para las prácticas.

Re3: Máquina virtual con Linux.

Re4: Web pública de la asignatura (<http://sopa.dis.ulpgc.es/fso>).

Re5: Infraestructura TIC del DIS y la ULPGC: Campus Virtual y Microsoft Teams.

Material didáctico para la teoría (todo disponible en [Re4]):

Re6: Guías didácticas teóricas.

Re7: Recopilaciones de ejercicios sobre teoría.

Re8: Exámenes de años anteriores (enunciados y soluciones).

Material didáctico para las prácticas (todo disponible en [Re5]):

Re9: Guías didácticas sobre las herramientas prácticas.

Re10: Fichas de las actividades prácticas.

Observaciones importantes sobre el uso de los recursos:

- Los comunicados oficiales de la asignatura se enviarán siempre a través del Moodle [Re5]. El estudiante deberá visitar el Moodle con frecuencia y consultar habitualmente la cuenta de correo de la ULPGC.
- Para realizar los trabajos prácticos, se dispone de varios laboratorios en el Edificio de Informática y Matemáticas, así como de una máquina virtual Linux que el estudiante podrá descargar y utilizar en sus propios equipos.
- La naturaleza de los trabajos prácticos permite que éstos se realicen en cualquier equipo con Linux, así que quien lo desee podrá implementarlos en su casa.

## **Resultados de aprendizaje que tendrá que alcanzar al finalizar las distintas tareas.**

Al completar con éxito la asignatura, el estudiante deberá ser capaz de:

RA1: Describir la estructura general, funciones y principios de diseño de los sistemas operativos.

RA2: Describir las políticas de gestión de recursos propias del sistema operativo.

RA3: Operar con el intérprete de línea de órdenes para interactuar con el sistema.

RA4: Identificar los principios metodológicos de la programación de sistemas.

RA5: Desarrollar componentes de software de sistemas, incluidos dentro del sistema operativo o del middleware.

--- Notas sobre el alcance de los resultados de aprendizaje

- El resultado RA3 se alcanzará en el contexto de la CLI de Linux, para ejecutar órdenes simples.

- El resultado RA4 se alcanzará en un nivel de principiante, centrado en los procesos básicos de escritura de código, compilación y pruebas.

- El resultado RA5 se alcanzará en el contexto de los sistemas Linux, desarrollando en lenguaje C con la biblioteca estándar (ficheros, procesos, hilos).

--- Relación con las actividades formativas

RA1, RA2: se vinculan con todas las actividades formativas.

RA3, RA4, RA5: se vinculan específicamente con las actividades formativas AF2, AF3, AF4, AF5, AF6.

## **Plan Tutorial**

### **Atención presencial individualizada (incluir las acciones dirigidas a estudiantes en 5ª, 6ª y 7ª convocatoria)**

La atención presencial individualizada se registrará por las directrices que establezca la universidad.

Las horas de atención al alumnado del equipo docente se encuentran publicadas y actualizadas en la página web del Departamento de Informática y Sistemas ([www.dis.ulpgc.es](http://www.dis.ulpgc.es)). Dichos horarios de atención se encuentran igualmente enlazados desde el sistema de información on-line de la asignatura ([sopa.dis.ulpgc.es/fso](http://sopa.dis.ulpgc.es/fso)).

Para garantizar una correcta atención al alumnado se recomienda reservar cita previa con el profesor. La reserva se podrá realizar contactando en persona con el profesor, mediante correo electrónico, Moodle o MS Teams. Tendrán preferencia en la atención aquellos alumnos que hayan realizado una reserva previa.

## Atención presencial a grupos de trabajo

La atención presencial a los grupos de trabajo se realizará en el horario de tutoría del profesor encargado del seguimiento del grupo, previa reserva de cita. Dicha reserva se podrá pactar en el despacho del profesor de forma presencial, mediante correo electrónico o mediante las herramientas disponibles en el Campus Virtual.

## Atención telefónica

Se podrán realizar consultas telefónicas al equipo docente en los horarios de atención presencial al alumnado.

## Atención virtual (on-line)

Por correo electrónico o mediante Microsoft Teams.

Se recomienda que las consultas virtuales se realicen EN LOS FOROS de Moodle, dado que es probable que una consulta y su respuesta tengan interés para toda la comunidad.

Las tutorías privadas sólo deberían usarse en casos en los que se requiera una atención individualizada.

## Datos identificativos del profesorado que la imparte.

### Datos identificativos del profesorado que la imparte

**Dr./Dra. José Miguel Santos Espino** (COORDINADOR)

**Departamento:** 260 - *INFORMÁTICA Y SISTEMAS*

**Ámbito:** 075 - *Ciencia De La Comp. E Intel. Artificial*

**Área:** 075 - *Ciencia De La Comp. E Intel. Artificial*

**Despacho:** *INFORMÁTICA Y SISTEMAS*

**Teléfono:** 928458783 **Correo Electrónico:** *josemiguel.santos@ulpgc.es*

**Dr./Dra. José Antonio Muñoz Blanco**

**Departamento:** 260 - *INFORMÁTICA Y SISTEMAS*

**Ámbito:** 075 - *Ciencia De La Comp. E Intel. Artificial*

**Área:** 075 - *Ciencia De La Comp. E Intel. Artificial*

**Despacho:** *INFORMÁTICA Y SISTEMAS*

**Teléfono:** 928458754 **Correo Electrónico:** *joseantonio.munoz@ulpgc.es*

**Dr./Dra. Francisco Alexis Quesada Arencibia**

**Departamento:** 260 - *INFORMÁTICA Y SISTEMAS*

**Ámbito:** 075 - *Ciencia De La Comp. E Intel. Artificial*

**Área:** 075 - *Ciencia De La Comp. E Intel. Artificial*

**Despacho:** *INFORMÁTICA Y SISTEMAS*

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

## Dr./Dra. Juan Carlos Quevedo Losada

**Departamento:** 260 - INFORMÁTICA Y SISTEMAS

**Ámbito:** 075 - Ciencia De La Comp. E Intel. Artificial

**Área:** 075 - Ciencia De La Comp. E Intel. Artificial

**Despacho:** INFORMÁTICA Y SISTEMAS

**Teléfono:** 928458757 **Correo Electrónico:** juancarlos.quevedo@ulpgc.es

## Bibliografía

---

### [1 Básico] Operating system concepts /

*Abraham Silberschatz, Peter Baer Galvin, Greg Gagne.*

*John Wiley & Sons,, Hoboken : (2019) - (Ed. global.)*

*9781119454083 (Observaciones: Silberschatz2019)*

---

### [2 Básico] C: manual de referencia /

*Herbert Schildt ; traducción, Luis Hernández Yáñez; revisión técnica, Antonio Vaquero Sánchez.*

*Osborne/McGraw-Hill,, Madrid [etc.] : (2003) - (4ª ed.)*

*970-10-4770-2 (Observaciones: Schildt2003)*

---

### [3 Básico] Linux system programming /

*Robert Love.*

*O'Reilly,, Beijing ... [etc.] : (2013) - (2nd ed.)*

*978-1-4493-3953-1 (Observaciones: Love2013)*

---

### [4 Recomendado] Fundamentos de sistemas operativos :teoría y ejercicios resueltos /

*Santiago Candela Solá...[et al.].*

*Thomson,, Madrid : (2007)*

*9788497325479 (Observaciones: ULPGC2007)*