



Universidad
de Alcalá

GUÍA DOCENTE

Sistemas Operativos

Grado en
Ingeniería Informática (GII)
Ingeniería de Computadores (GIC)

Universidad de Alcalá

Curso Académico 2022/2023

2º Curso - 1^{er} Cuatrimestre (GII+GIC)

GUÍA DOCENTE

Nombre de la asignatura:	Sistemas Operativos
Código:	780007 (GII+GIC)
Titulación en la que se imparte:	Grado en Ingeniería Informática (GII) Ingeniería de Computadores (GIC)
Departamento y Área de Conocimiento:	Automática Arquitectura y Tecnología de Computadores
Carácter:	Básica (GII+GIC)
Créditos ECTS:	6.0
Curso y cuatrimestre:	2º Curso - 1º Cuatrimestre (GII+GIC)
Profesorado:	Se indica en la página web de la asignatura Sistemas Operativos
Horario de Tutoría:	Consultar al comienzo de la asignatura
Idioma en el que se imparte:	Español

1a. PRESENTACIÓN

Esta guía es una herramienta que permitirá al alumno conocer los contenidos que componen la materia, las competencias que se adquirirán con su estudio, la distribución en el tiempo de las diferentes actividades, y los requisitos para superar la asignatura, así como otros datos de interés. Podrá también descargarse de la plataforma docente disponible en Internet para la asignatura.

Sistemas Operativos es una asignatura básica que se imparte en el primer cuatrimestre del segundo curso del Grado en Ingeniería de Computadores y del Grado en Ingeniería Informática. Es la primera de las asignaturas de la materia de Sistemas Operativos, que se completará durante el Grado con Sistemas Operativos Avanzados y, finalmente, con Sistemas Empotrados y Sistemas de Tiempo Real, siendo estas dos últimas de carácter obligatorio en la titulación de Grado en Ingeniería de Computadores.

El objetivo de esta asignatura es introducir al alumno en la necesidad de emplear sistemas software que ayuden a proporcionar niveles de abstracción suficientemente altos como para acometer el desarrollo de otros sistemas aún más complejos.

Los Sistemas Operativos son los encargados de poner los recursos hardware de nuestra plataforma, de forma sencilla y segura, a disposición de los usuarios. Su evolución ha estado frecuentemente ligada a la de las Arquitecturas de Computadores, tomando de esta disciplina gran número de conceptos y técnicas. A su vez, las Arquitecturas de Computadores han evolucionado para dar soporte a los requisitos que, a través de los Sistemas Operativos, han ido imponiendo los usuarios a lo largo del tiempo. Esta realimentación mutua es vital para la comprensión del estado actual de esta disciplina, así como también para entender sus tendencias futuras.

El primer tema se dedicará al estudio de esta relación, haciendo principal hincapié en aquellos elementos de la Arquitectura de Computadores que sean necesarios para abordar el estudio de los Sistemas Operativos.

El segundo tema plantea el estudio de los diferentes enfoques de diseño que se pueden emplear para construir un Sistema Operativo. Se introducirá al estudiante en una serie de conceptos generales sobre el diseño de grandes sistemas particularizados para esta disciplina concreta. Entre ellos, cabe resaltar el diseño por capas, la separación entre mecanismos y políticas, etc. A continuación, se hará un estudio de las interfaces que generalmente comunican al usuario y a las aplicaciones con el Sistema Operativo, para concluir con un estudio sobre el núcleo y los mecanismos de llamadas al sistema.

El tercer tema permitirá al alumno establecer las diferencias entre programas y procesos, así como la estructura de ambos en cada uno de los contextos en los que se desenvuelven. Al finalizar el tema el estudiante será capaz de justificar la introducción de hilos en los Sistemas Operativos modernos, establecer las características de los mismos, y realizar pequeños programas que hagan uso de ellos. Este tema concluirá con una serie de casos de estudio de Sistemas Operativos reales. Estos casos permitirán encuadrar todos los conceptos teóricos aprendidos anteriormente, así como detalles particulares propios de cada implementación.

El cuarto tema presenta los diferentes escenarios en los que es necesario que procesos e hilos se sincronicen. Se estudiarán conceptos claves como exclusión mutua, atomicidad, sección crítica, etc., en el contexto de algunos de los problemas típicos. Se hará especial hincapié en los mecanismos proporcionados por el Sistema Operativo para resolver los problemas que pueden surgir al acceder varios procesos o hilos de forma no controlada a recursos del sistema compartidos.

El último tema está dedicado a la planificación de procesos e hilos. Con este tema se planteará al alumno la necesidad de llevar a cabo una selección de qué proceso debe ejecutarse en cada momento para conseguir mejorar una serie de parámetros de rendimiento. Asimismo, se estudiarán las políticas de planificación clásicas y de sincronización con semáforos, finalizando con el estudio de las técnicas utilizadas por algunos Sistemas Operativos comerciales.

Prerrequisitos y Recomendaciones

Esta asignatura se apoya en los conocimientos adquiridos por los estudiantes en las asignaturas de *Fundamentos de Tecnología de Computadores*, *Estructura y Organización de Computadores* y *Fundamentos de Programación*, impartidas las tres durante los dos cuatrimestres precedentes.

1b. COURSE SUMMARY

Operating Systems is a basic subject imparted in the second term, of the first course, of the Graduate on Computer Engineering (second term of the whole degree). It is the first subject of the matter of Operating Systems, which will continue with Advanced Operating Systems.

The goal of this subject is to introduce the student to the need of using software systems that help providing abstraction levels high enough to accomplish the development of even more complex systems. Operating systems are in charge of making hardware resources available to the users in a simple and safe manner. Their evolution has often been linked to that of Computer Architectures, being conditioned by this discipline in a large number of concepts and techniques. In turn, Computer Architectures evolved to support the requisites imposed by the users through Operating Systems. This mutual feedback is vital to explain the current state of this discipline, as well as to understand its future trends.

2. COMPETENCIAS

Competencias básicas, generales y transversales.

Esta asignatura contribuye a adquirir las siguientes competencias básicas, generales y transversales:

CG4 - Capacidad para definir, evaluar y seleccionar plataformas hardware y software para el desarrollo y la ejecución de sistemas, servicios y aplicaciones informáticas, de acuerdo con los conocimientos adquiridos según lo establecido en el apartado 5, anexo 2, de la resolución BOE-A-2009-12977.

CG6 - Capacidad para concebir y desarrollar sistemas o arquitecturas informáticas centralizadas o distribuidas integrando hardware, software y redes de acuerdo con los conocimientos adquiridos según lo establecido en el apartado 5, anexo 2, de la resolución BOE-A-2009-12977.

CG8 - Conocimiento de las materias básicas y tecnologías, que capaciten para el aprendizaje y desarrollo de nuevos métodos y tecnologías, así como las que les doten de una gran versatilidad para adaptarse a nuevas situaciones.

CG9 - Capacidad para resolver problemas con iniciativa, toma de decisiones, autonomía y creatividad. Capacidad para saber comunicar y transmitir los conocimientos, habilidades y destrezas de la profesión de Ingeniero Técnico en Informática.

CB1 - Que los estudiantes hayan demostrado poseer y comprender conocimientos en un área de estudio que parte de la base de la educación secundaria general, y se suele encontrar a un nivel que, si bien se apoya en libros de texto avanzados, incluye también algunos aspectos que implican conocimientos procedentes de la vanguardia de su campo de estudio.

CB2 - Que los estudiantes sepan aplicar sus conocimientos a su trabajo o vocación de una forma profesional y posean las competencias que suelen demostrarse por medio de la elaboración y

defensa de argumentos y la resolución de problemas dentro de su área de estudio.

CB3 - Que los estudiantes tengan la capacidad de reunir e interpretar datos relevantes (normalmente dentro de su área de estudio) para emitir juicios que incluyan una reflexión sobre temas relevantes de índole social, científica o ética.

CB4 - Que los estudiantes puedan transmitir información, ideas, problemas y soluciones a un público tanto especializado como no especializado.

CB5 - Que los estudiantes hayan desarrollado aquellas habilidades de aprendizaje necesarias para emprender estudios posteriores con un alto grado de autonomía.

TRU1 - Capacidad de análisis y síntesis.

TRU2 - Comunicación oral y escrita.

TRU3 - Capacidad de gestión de la información.

TRU4 - Capacidad de aprendizaje autónomo.

TRU5 - Capacidad para trabajar en equipo.

Competencias Específicas

Esta asignatura proporciona la(s) siguiente(s) competencia(s) de carácter específicas:

CIB4 - Conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería.

Resultados de aprendizaje

Al terminar con éxito esta asignatura/enseñanza, los estudiantes serán capaces de:

RA1. Identificar los fundamentos de un sistema operativo, sus componentes y los conceptos esenciales para la comprensión de los mismos.

RA2. Justificar la necesidad de los sistemas operativos en los entornos de computación actuales y su papel como interfaz entre el hardware y los programas de usuario.

RA3. Distinguir diversos sistemas operativos y entornos de operación (tradicionales, GUI, multimedia, etc.), sus diferencias y requisitos en términos de recursos.

RA4. Instalar, configurar y operar un sistema operativo multiusuario.

RA5. Desarrollar aplicaciones mediante el uso de la API de un sistema operativo a nivel de procesos e hilos.

RA6. Distinguir las técnicas de planificación de tareas más relevantes, para sistemas batch, interactivos y de tiempo real .

RA7. Identificar la necesidad de realizar actividades concurrentes, los problemas que estas provocan y las soluciones a estos problemas.

3. CONTENIDOS

Bloques de contenido partes Teóricas	Total de horas presenciales
<p>PARTE 1: Introducción a los Sistemas Operativos y conceptos previos sobre Arquitectura de Computadores.</p> <ul style="list-style-type: none"> • Definición de Computador • Definición de Sistema Operativo • Modelo de máquina desnuda <ul style="list-style-type: none"> ◦ Arquitectura de Von Neumann. ◦ CPU, memoria, buses y Entrada y Salida • Monitor simple residente <ul style="list-style-type: none"> ◦ Monoprocesamiento ◦ Protección del monitor ◦ Llamadas a servicios del monitor ◦ Concepto de interrupción • Procesamiento por lotes (batch) <ul style="list-style-type: none"> ◦ Alternancia en el uso de CPU y E/S ◦ Arquitectura de E/S y técnicas de buffering • Sistemas multiprogramados <ul style="list-style-type: none"> ◦ E/S por interrupciones ◦ Protección de memoria ◦ Modo dual de ejecución: trap's e Interrupciones. ◦ Separación de espacios de direccionamiento: concepto de MMU • Tiempo compartido <ul style="list-style-type: none"> ◦ Protección contra monopolio de CPU ◦ Técnica de desalojo ◦ Necesidad de planificación • Técnicas de diseño: <ul style="list-style-type: none"> ◦ Vista funcional y estructural ◦ Diferencia entre mecanismos y políticas ◦ Representación jerárquica de un computador 	8 horas
<p>PARTE 2: Estructura del Sistema Operativo</p> <ul style="list-style-type: none"> • Vistas restringidas y amplia de un Sistema Operativo. • Funciones del Sistema Operativo • Interfaces del Sistema Operativo <ul style="list-style-type: none"> ◦ Con el usuario: <ul style="list-style-type: none"> ▪ La shell ▪ Entornos gráficos ◦ Con las aplicaciones: llamadas sistema. • Descomposición en capas del Sistema Operativo • El núcleo del Sistema Operativo <ul style="list-style-type: none"> ◦ Descripción y funciones básicas ◦ Casos de estudio de diseños: Linux, Windows, Mach • Mecanismos de llamadas al sistema <ul style="list-style-type: none"> ◦ Descripción ◦ Tipos de llamadas al sistema 	12 horas

<p>PARTE 3: Procesos e Hilos</p> <ul style="list-style-type: none"> • Programas vs. procesos • Estructura de un programa • Concepto de proceso • Proyección de un programa en memoria en el contexto de usuario • Creación de un proceso <ul style="list-style-type: none"> ◦ Ciclo de vida básico ◦ Llamadas al sistema relacionadas con procesos ◦ Contexto del núcleo: estructuras de datos internas • Hilos <ul style="list-style-type: none"> ◦ Justificación ◦ Características ◦ Procesos vs. hilos ◦ Ejemplo de uso de hilos POSIX 	<p>14 horas</p>
<p>PARTE 4: Sincronización de procesos</p> <ul style="list-style-type: none"> • Introducción <ul style="list-style-type: none"> ◦ Necesidad de comunicación y sincronización ◦ Tipos de escenarios concurrentes • Conceptos asociados a la sincronización <ul style="list-style-type: none"> ◦ Condiciones de carrera ◦ Atomicidad ◦ Problema de la sección crítica: exclusión mutua, progreso y espera limitada • Mecanismos de comunicación y sincronización proporcionados por el Sistema Operativo <ul style="list-style-type: none"> ◦ Tuberías <ul style="list-style-type: none"> ▪ Tipos y operaciones sobre tuberías ▪ Llamadas al sistema para tuberías ▪ Problemas clásicos de sincronización con tuberías ◦ Semáforos <ul style="list-style-type: none"> ▪ Operaciones sobre semáforos ▪ Llamadas al sistema para semáforos ▪ Problemas clásicos de sincronización con semáforos ◦ Mutex y variables condicionales <ul style="list-style-type: none"> ▪ Operaciones sobre mutex y variables condicionales ▪ Llamadas al sistema para mutex y variables condicionales ▪ Problemas clásicos de sincronización con mutex y variables condicionales 	<p>12 horas</p>

PARTE 5: Planificación del uso de la CPU <ul style="list-style-type: none"> • Mecanismo de conmutación de CPU • Concepto de planificación <ul style="list-style-type: none"> ◦ Justificación ◦ Objetivos ◦ Parámetros de evaluación • Tipos de planificadores • Políticas de planificación básicas <ul style="list-style-type: none"> ◦ FCFS ◦ SJF • Conceptos de prioridad y requisa • Políticas de planificación avanzadas <ul style="list-style-type: none"> ◦ Planificación por prioridad pura ◦ Turno rotatorio (<i>round robin</i>) ◦ Colas multinivel con y sin realimentación • Casos de uso: Linux y Windows 	10 horas
--	----------

Semana	Contenido
1 ^a -2 ^a	PARTE 1: Teoría (4h) + Práctica (4h)
3 ^a -5 ^a	PARTE 2: Teoría (6h) + Práctica (6h)
6 ^a -9 ^a	PARTE 3: Teoría (6h) + Práctica (8h) + Evaluación (1h)
9 ^a -12 ^a	PARTE 4: Teoría (6h) + Práctica (6h)
12 ^a -15	PARTE 5: Teoría (6h) + Práctica (4h) + Evaluación (1h)
	Examen final (2h)

4. METODOLOGÍAS DE ENSEÑANZA APRENDIZAJE. ACTIVIDADES FORMATIVAS

4.1. Distribución de créditos (especificar en horas)

Número de horas presenciales:	58 horas (56 horas de clase presencial +2 horas de evaluación)
Número de horas del trabajo propio del estudiante:	92 (Incluye horas de estudio, elaboración de actividades, preparación de exámenes, actividades <i>online</i>)
Total horas	150

4.2. Estrategias metodológicas, materiales y recursos didácticos

Clases presenciales	<ul style="list-style-type: none"> • Clases teóricas: estas clases se impartirán en grupos grandes y en ellas, mediante docencia magistral, el profesor desarrollará los conceptos más importantes para la comprensión de los contenidos de la asignatura. • Resolución de casos prácticos: se harán en grupos reducidos. Durante las sesiones se plantearán problemas susceptibles de resolución mediante técnicas expuestas en clase. De forma guiada se procederá a la aplicación de dichas técnicas para la resolución del problema. • Presentación de informes y trabajos: el alumno deberá presentar a sus compañeros y al profesor informes y proyectos que haya realizado de forma individual o en grupos reducidos. Las presentaciones harán uso de las técnicas multimedia apropiadas. • Pruebas parciales: durante el desarrollo del curso el profesor propondrá diversas pruebas parciales para revisar la adquisición de conocimientos y su aplicación.
Trabajo autónomo	<ul style="list-style-type: none"> • Lecturas • Realización de actividades: ejercicios, mapas conceptuales, ejemplificaciones o búsqueda de información. • Participación en foros y actividades: generalmente, a través de la plataforma docente de la asignatura.
Tutorías	<p>Las tutorías podrán ser tanto en grupos como individuales. Durante las mismas, el profesor podrá evaluar la adquisición de las competencias y revisará los informes aportados por los estudiantes sobre los trabajos encomendados.</p>

Materiales y recursos

Los materiales para la preparación de las sesiones presenciales, así como las actividades a realizar por el estudiante de forma individual, se podrán encontrar en el Aula Virtual de la Universidad de Alcalá. El funcionamiento de esta herramienta docente se detallará en la clase de presentación de la asignatura; se explicará, entre otros aspectos, la forma en la que los estudiantes se inscribirán en el foro general de mensajes, mecanismo habitual de comunicación con los estudiantes.

Para cada actividad, el profesor proporcionará una serie de referencias bibliográficas que pueden consultarse en la biblioteca de la Escuela Politécnica.

Para aquellas actividades que así lo requieran, el profesor indicará la forma de planificarlas, así como los trabajos a entregar que deban resultar de la realización de las mismas.

5. EVALUACIÓN: Procedimientos, criterios de evaluación y calificación

Preferentemente se ofrecerá a los alumnos un sistema de evaluación continua que tenga características de evaluación formativa de manera que sirva de realimentación en el proceso de enseñanza-aprendizaje por parte del alumno.

5.1. PROCEDIMIENTOS

La evaluación debe estar inspirada en los criterios de evaluación continua (Normativa de Evaluación de los Aprendizajes, NEA, art 3). No obstante, respetando la normativa de la Universidad de Alcalá se pone a disposición del alumno un proceso alternativo de evaluación final de acuerdo a la Normativa de Evaluación de los Aprendizaje (aprobada en Consejo de Gobierno de 24 de marzo de 2011 y modificada en Consejo de Gobierno de 5 de mayo de 2016) según lo indicado en su Artículo 10, los alumnos tendrán un plazo de quince días desde el inicio del curso para solicitar por escrito al Director de la Escuela Politécnica Superior su intención de acogerse al modelo de evaluación no continua aduciendo las razones que estimen convenientes. La evaluación del proceso de aprendizaje de todos los alumnos que no cursen solicitud al respecto o vean denegada la misma se realizará, por defecto, de acuerdo al modelo de evaluación continua. El estudiante dispone de dos convocatorias para superar la asignatura, una ordinaria y otra extraordinaria.

Evaluación continua

La evaluación continua valora el desarrollo de las competencias durante todo el proceso de aprendizaje de la asignatura mediante una serie de pruebas de carácter sumatorio y formativo distribuidas a lo largo del curso. De este modo, el estudiante puede abordar la asignatura de forma progresiva.

Esta forma de evaluación garantiza la retroalimentación temprana en el proceso de aprendizaje del alumno y permite a los profesores, coordinadores y demás elementos del Sistema de Garantía de Calidad hacer un seguimiento global, con la posibilidad de actuar en caso de que lo aconsejen indicadores o situaciones determinadas.

La evaluación de la parte relacionada con las prácticas se realizará al finalizar su bloque correspondiente.

5.2. EVALUACIÓN

Criterios de evaluación

Para determinar el grado de adquisición de las competencias por parte del alumno, se tendrán en cuenta las habilidades, actitudes y valores demostrados por el estudiante de acuerdo con los siguientes criterios de evaluación:

CE1: El alumno ha adquirido los conocimientos sobre los fundamentos de los sistemas operativos y su estructura, sobre los procesos e hilos, así como sobre los mecanismos básicos de planificación y sincronización.

CE2: El alumno muestra capacidad de aplicación e integración de los contenidos a problemas, escenarios o casos de estudio fines a la asignatura.

CE3: El alumno muestra capacidad e iniciativa para operar la interfaz de usuario de un sistema operativo.

CE4: El alumno muestra capacidad e iniciativa a la hora de desarrollar software utilizando la API (*Application Programming Interface*) del sistema operativo a nivel de gestión de procesos.

CE5: El alumno muestra capacidad de argumentación y de emisión de juicios sobre escenarios y casos de estudios planteados en la asignatura.

CE6: El alumno cumple con las tareas encomendadas.

CE7: El alumno muestra interés por los contenidos y la materia trabajada.

CE8: El alumno muestra cuidado formal, claridad y rigor en la exposición de ideas y razonamientos.

Instrumentos de calificación

Sistema de evaluación continua

La evaluación de los alumnos se realizará progresivamente durante el desarrollo del curso. Su rendimiento será evaluado por su trabajo, conocimientos y destrezas adquiridas, así como la mejora de su proceso de aprendizaje.

Los instrumentos de **evaluación continua** a emplear consistirán en la realización de actividades planteadas por el profesor para cada uno de los temas. En global, estas actividades supondrán el 65% de la calificación del alumno.

El total de las actividades planteadas, su tipo, contenido y temporización, se comunicará al estudiante durante la clase de presentación y a la finalización del bloque de contenido correspondiente. Dichas actividades comprenden:

- **PEI: Pruebas de Evaluación Intermedia** de tipo test realizadas durante el periodo docente sobre los contenidos de las partes teóricas de la asignatura. No se permitirán el uso de libros (*25% de la nota final*).
- **PL, E:** Pruebas realizadas sobre la resolución de ejercicios prácticos en el laboratorio (*40% de la nota final*). La evaluación se realizará considerando el desarrollo de un conjunto de prácticas realizadas en equipos de trabajo. La calificación se obtendrá mediante pruebas de laboratorio individuales (**PL**) de tipo test y/o ensayo (respuesta corta, defensa oral) y/o mediante la entrega (**E**) de dichas prácticas. Como herramienta de apoyo en la evaluación se utilizarán rúbricas.

Ambos tipos de pruebas permiten mostrar la capacidad del alumno de aplicar herramientas para la comprensión del núcleo del sistema operativo y las interfaces del sistema operativo con el usuario (comandos) y con las aplicaciones (API), las herramientas de desarrollo de programas y/o aplicación de algoritmos de planificación de la CPU.

- **PEF: Prueba de Evaluación Final** consistente en la resolución de problemas al terminar el periodo docente (*35% de la nota final*).

Sistema de evaluación única

A aquellos alumnos a los que se les haya concedido la evaluación final, se les evaluará mediante un único examen global acerca de todos los contenidos de la asignatura.

La **evaluación final** considera los siguientes instrumentos de evaluación:

- **PL, E:** Pruebas individuales y en grupo sobre la resolución de ejercicios prácticos en laboratorio (*40% de la nota final*) realizadas durante el periodo docente (véase su descripción en **Sistemas de evaluación continua**).
- **PEFT: Prueba de Evaluación Final** de tipo Test sobre los contenidos de todas las partes teóricas de la asignatura (*25% de la nota final*).
- **PEF:** Prueba consistente en la resolución de problemas (*35% de la nota final*).

Si el alumno no supera la asignatura en la convocatoria ordinaria, en evaluación continua o única, tendrá la posibilidad de presentarse a un examen extraordinario con los mismos instrumentos de evaluación descritos para evaluación final.

El estudiante encontrará en el curso virtual de la asignatura todos los detalles sobre las pruebas de evaluación: fechas, contenidos, formatos, etc. Asimismo, es muy importante que tenga en cuenta el

Artículo 34.3 de la Normativa de Evaluación de los Aprendizajes, respecto a la originalidad de los trabajos y pruebas:

"Se entiende por plagio la copia de textos sin citar su procedencia y dándolos como de elaboración propia y conllevará automáticamente la calificación de suspenso (0) en los trabajos o pruebas en los que se hubiera detectado. El profesor que advierta indicios de plagio en los trabajos o pruebas de evaluación que les sean presentados dará cuenta de este hecho al responsable del título en un plazo máximo de dos días, para que proceda, en su caso, a ponerlo en conocimiento del Rector por si pudiera ser constitutivo de infracción disciplinaria o de delito."

Criterios de calificación

Esta sección muestra la relación entre resultados de aprendizaje, criterios, instrumentos y calificación.

En las siguientes tablas **PLi** referencia la Prueba de Laboratorio asociada a la **práctica i** y **Ei** la Entrega asociada a la **práctica i**.

Convocatoria ordinaria, evaluación continua

Competencias	Resultado de Aprendizaje	Criterio de Evaluación	Instrumento de evaluación	Peso en la calificación
CG8, CIB4	RA1-RA3, RA6, RA7	CE1	PEI1, PEI2	25%
CG8, CG4, CG9, CIB4	RA3-RA4	CE2, CE3	PL1	15%
CG8, CG6, CG9, CIB4	RA3-RA5	CE2, CE4-CE8	E2+PL2, E3	15%, 10%
CG8, CG9, CIB4	RA1-RA7	CE1-CE8	PEF1	35%

Convocatoria ordinaria, evaluación final, y convocatoria extraordinaria

Competencias	Resultado de Aprendizaje	Criterio de Evaluación	Instrumento de evaluación	Peso en la calificación
CG8, CIB4	RA1-RA3, RA6, RA7	CE1	PEFT1	25%
CG8, CG4, CG9, CIB4	RA3-RA4	CE2, CE3	PL1	15%
CG8, CG6, CG9, CIB4	RA3-RA5	CE2, CE4-CE8	E2+PL2, E3	15%, 10%
CG8, CG9, CIB4	RA1-RA7	CE1-CE8	PEF2	35%

Es requisito para aprobar la asignatura, la superación de las prácticas realizadas durante el curso ya que constituyen la parte práctica, según el artículo 6.4 de la Normativa de Evaluación de los Aprendizajes.

6. BIBLIOGRAFÍA

6.1. Bibliografía básica

- Sistemas Operativos. S. Sánchez Prieto. Segunda edición. Servicio de Publicaciones de la Universidad de Alcalá. 2005.

6.2. Bibliografía complementaria

Teoría

- Sistemas Operativos Modernos. A. S. Tanenbaum. 3ª edición. Prentice Hall, 2009.
- Operating Systems. Internals and Design Principles. W. Stallings. 8ª edición. Pearson Prentice Hall, 2014.
- Operating Systems Design and Implementation. A. S. Tanenbaum y A. S. Woodhull. 3th edition. Pearson Prentice Hall, 2009.
- Operating System Concepts. A. Silberschatz, P. B. Galvin y G. Gagne. 9th edition. Wiley, 2015.
- Unix. Programación avanzada. Francisco M. Márquez García. Tercera edición. Ra-Ma, 2004.
- Computer organization and architecture. William Stallings. 10th edition. Pearson Prentice Hall, 2015.

Práctica

- Unix y Linux, guía práctica. S. Sánchez Prieto y O. García Población. 3ª edición. Ra-Ma, 2004.
- El entorno de programación UNIX. B. W. Kernighan y R. Pike. 1ª edición. Prentice Hall Hispanoamericana S. A., 1989.
- Unix, programación avanzada. F. Márquez García. 3ª edición. Ra-Ma, 2004.
- El lenguaje de programación C. B. W. Kernighan y D. Ritchie. 2ª edición. Prentice Hall Hispanoamericana S. A., 1991.
- C/C++. Curso de programación. F. J. Ceballos Sierra. 5ª edición. Ra-Ma, 2019.

NOTA INFORMATIVA

La Universidad de Alcalá garantiza a sus estudiantes que, si por exigencias sanitarias las autoridades competentes impidieran la presencialidad total o parcial de la actividad docente, los planes docentes alcanzarían sus objetivos a través de una metodología de enseñanza-aprendizaje y evaluación en formato online, que retornaría a la modalidad presencial en cuanto cesaran dichos impedimentos.