25-26

GRADO EN INGENIERÍA EN ELECTRÓNICA INDUSTRIAL Y AUTOMÁTICA
TERCER CURSO

GUÍA DE ESTUDIO PÚBLICA

SISTEMAS EN TIEMPO REAL

CÓDIGO 6802306-

UNED

25-26

SISTEMAS EN TIEMPO REAL

CÓDIGO 6802306-

ÍNDICE

PRESENTACIÓN Y CONTEXTUALIZACIÓN
REQUISITOS Y/O RECOMENDACIONES PARA CURSAR LA ASIGNATURA
EQUIPO DOCENTE
HORARIO DE ATENCIÓN AL ESTUDIANTE
TUTORIZACIÓN EN CENTROS ASOCIADOS
COMPETENCIAS QUE ADQUIERE EL ESTUDIANTE
RESULTADOS DE APRENDIZAJE
CONTENIDOS
METODOLOGÍA
SISTEMA DE EVALUACIÓN
BIBLIOGRAFÍA BÁSICA
BIBLIOGRAFÍA COMPLEMENTARIA
RECURSOS DE APOYO Y WEBGRAFÍA
PRÁCTICAS DE LABORATORIO
IGUALDAD DE GÉNERO

NOMBRE DE LA ASIGNATURA	SISTEMAS EN TIEMPO REAL
CÓDIGO	6802306-
CURSO ACADÉMICO	2025/2026
DEPARTAMENTO	SISTEMAS DE COMUNICACIÓN Y CONTROL, INGENIERÍA ELÉCTRICA, ELECTRÓNICA, CONTROL, TELEMÁTICA Y QUÍMICA APLICADA A LA INGENIERÍA
TÍTULO EN QUE SE IMPARTE	GRADO EN INGENIERÍA EN ELECTRÓNICA INDUSTRIAL Y AUTOMÁTICA
CURSO - PERIODO - TIPO	GRADUADO EN ING. EN ELECTRÓNICA INDUSTRIAL Y AUTOMÁTICA (PLAN 2024) - TERCERCURSO - SEMESTRE 2 - OBLIGATORIAS
CURSO - PERIODO - TIPO	GRADUADO EN ING. EN ELECTRÓNICA INDUSTRIAL Y AUTOMÁTICA (PLAN 2009) - TERCERCURSO - SEMESTRE 2 - OBLIGATORIAS
Nº ETCS	5
HORAS	125.0
IDIOMAS EN QUE SE IMPARTE	CASTELLANO

PRESENTACIÓN Y CONTEXTUALIZACIÓN

La programación de sistemas en tiempo real es una disciplina compleja ya que el desarrollo de código para este tipo de sistemas requiere una especificación de los “límites” de los sistemas en cuanto a temporización de las tareas y las facilidades de comunicación y sincronización entre procesos del sistema. Adicionalmente es muy relevante la consideración sobre la restricción de ejecución de las tareas y sus prioridades, es decir, la capacidad de decidir que tareas deben ejecutarse, ya que la falta de tiempo para estas puede provocar estados de emergencia en el propio sistema (imagine que falla el software de navegación de un avión). Para resolver estos problemas se introducen conceptos como fiabilidad y tolerancia a fallos, que permiten diseñas sistemas en tiempo real fiables y no propensos a fallos en el propio software. La asignatura pretende mostrar dichos mecanismos y las diferentes estructuras software que permiten la implementación de estos objetivos básicos, en cuanto a: temporización estricta de la ejecución de las tareas de proceso, facilidad de comunicación sin bloqueo de procesos y estabilidad del software.
La programación de sistemas en tiempo real es una disciplina específica de cierto tipo de sistemas que tienen una serie de restricciones temporales en cuanto a la ejecución planificada de procesos. Estas restricciones de desarrollan en base a la “importancia” del proceso que se debe ejecutar en unos instantes predeterminados y predecibles, de forma que se sepa de manera fiable que dichos procesos realizan su función dentro de los límites temporales establecidos. Para realizar esta función, lo primero es conocer que es lo que se entiende pro sistema en tiempo real y las diferentes categorías en las que se pueden dividir,

de forma que se disponga de una separación clara entre procesos prioritarios y no prioritarios, en cuanto a ejecución temporal se refieren.
Como consecuencia de la prioridad de las tareas, es muy importante definir el concepto de fiabilidad, que nos permita asegurar que las tareas se ejecutan en tiempo y forma, haciendo este sistema en tiempo real seguro y solvente a fallos. La tolerancia a fallos permite describir la forma de manejar posibles fallos y las posibles acciones que se deben realizar bajo distintos tipos de fallos, con el objetivo de que el sistema en tiempo real puede seguir realizando su función.
Sin embargo, no solo se debe asegurar que las tareas se ejecuten de un manera específica y predecible, sino que tampoco existan interferencias entre los procesos de comunicación de información entre dichas tareas. Esto es, las tareas deben intercambiar información entre ellas y por tanto se crea un escenario de concurrencia que debe ser resuelto con los mecanismos conocidos (variables compartidas, semáforos, mensajes, monitores, etc.) y teniendo en cuenta las propias restricciones de la planificación de sistemas de tiempo real.
Una vez que se tienen en cuenta los aspectos anteriores se pueden usar diversos tipos de estrategias de planificación, de acuerdo a los requerimientos del sistema en tiempo real que se desea desarrollar. Todas estos tipos de planificación requieren de la definición de un mecanismo de temporización, denominado reloj, que se usa como base para el desarrollo de los distintos tipos de planificación y que el sistema operativo que ejecuta el software debe proporcionar.
Teniendo en cuenta lo anterior, se pretende que esta asignatura sirva como curso introductorio a la programación de sistemas en tiempo real y tiene como objetivo principal que el alumno pueda adquirir los conocimientos y habilidades necesarias para poder desarrollar ejemplos sencillos de planificaciones temporales basadas en restricciones de tiempo real y que sea capaz de disponer de información suficiente para analizar las diferentes alternativas de programación de sistemas de tiempo real.

REQUISITOS Y/O RECOMENDACIONES PARA CURSAR LA ASIGNATURA

Se considera imprescindible para la realización y seguimiento del curso, que el alumno posea unos sólidos fundamentos en el área de la programación con Java. Durante la práctica totalidad del temario de la asignatura, los ejemplos prácticas y actividades planificadas presuponen un conocimiento medio del lenguaje de programación Java, es decir, la capacidad de desarrollar clases Java que implementen diferentes comportamientos y usen diferentes librerías del propio lenguaje. En particular, la asignatura enseña a emplear ciertas librerías específicas adecuadas para entornos de sistemas de tiempo real. Se presupone que el alumno ha cursado la asignatura FUNDAMENTOS DE INFORMÁTICA, ubicada en cursos anteriores del plan de estudio.

Es recomendable, aunque no necesario, el conocimiento sobre sistemas operativos. Los propios sistemas operativos están pensados como piezas de software que pueden estar distribuidas en diferentes recursos físicos y que se encargan de la ejecución de las

tareas/procesos involucrados en los sistemas de tiempo real. De hecho, la propia programación de los sistemas operativos en tiempo real es un buen ejemplo de desarrollo de servicios orientados a tiempo real.

EQUIPO DOCENTE

Nombre y Apellidos	RAFAEL PASTOR VARGAS (Coordinador/a de asignatura)
Correo Electrónico	rpastor@dia.uned.es
Teléfono	91398-8383
Facultad	ESCUELA TÉCN.SUP INGENIERÍA INFORMÁTICA
Departamento	SISTEMAS DE COMUNICACIÓN Y CONTROL

Nombre y Apellidos	RAFAEL PASTOR VARGAS (Coordinador/a de asignatura)
Correo Electrónico	rpastor@scc.uned.es
Teléfono	91398-8383
Facultad	ESCUELA TÉCN.SUP INGENIERÍA INFORMÁTICA
Departamento	SISTEMAS DE COMUNICACIÓN Y CONTROL

Nombre y Apellidos	ROBERTO HERNANDEZ BERLINCHES
Correo Electrónico	roberto@scc.uned.es
Teléfono	91398-7196
Facultad	ESCUELA TÉCN.SUP INGENIERÍA INFORMÁTICA
Departamento	SISTEMAS DE COMUNICACIÓN Y CONTROL

Nombre y Apellidos	PABLO RUIPEREZ GARCIA
Correo Electrónico	pablo@scc.uned.es
Teléfono	91398-7159
Facultad	ESCUELA TÉCN.SUP INGENIERÍA INFORMÁTICA
Departamento	SISTEMAS DE COMUNICACIÓN Y CONTROL

HORARIO DE ATENCIÓN AL ESTUDIANTE

La tutorización de los estudiantes tendrá lugar esencialmente a través de los foros de la plataforma, aunque también podrán utilizarse ocasionalmente otros medios, tales como chats interactivos, servicios de mensajería instantánea y el correo electrónico. Adicionalmente, está también previsto, para temas personales que no afecten al resto de los estudiantes, atender consultas en persona o por teléfono.

El seguimiento del aprendizaje se realizará revisando la participación de los alumnos en los distintos foros de debate y las aportaciones de material nuevo además de la entrega en fecha de los diferentes trabajos prácticos que se han planificado durante la evolución del curso.

Adicionalmente a la participación en el curso virtual, se puede contactar con el equipo docente en horario de guardia (por teléfono) o a través del correo electrónico:

Rafael Pastor Vargas, Lunes de 16:00 a 18:00 horas (rpastor@scc.uned.es). Teléfono: 91 398 8383

Roberto Hernández Berlinches, Lunes de 16:00 a 18:00 horas (roberto@scc.uned.es). Teléfono: 91 398 7196

Pablo Ruipérez García, Miercoles de 16:00 a 18:00 horas (pablo@scc.uned.es). Teléfono:

91 398 7159
Para el contacto por correo electrónico se debe comenzar el Asunto del mensaje mediante el prefijo [STR], es decir, un mensaje de ejemplo como consulta al equipo docente tendría como asunto del correo:
[STR] Duda sobre el temario
También se puede contactar por correo ordinario, en la siguiente dirección postal:
Sistemas de Comunicación y Control, ETSI Informática
Calle Juan del Rosal, 16
Ciudad Universitaria
28040 Madrid

TUTORIZACIÓN EN CENTROS ASOCIADOS

Las sesiones de tutoría se imparten presencialmente y en línea desde el Centro Asociado de referencia o, en su defecto, desde otro Centro Asociado. A continuación se muestran los centros asociados y extensiones en las que se imparten tutorías de la asignatura.

COMPETENCIAS QUE ADQUIERE EL ESTUDIANTE

Ver sección de Resultados de Aprendizaje.

RESULTADOS DE APRENDIZAJE

HABILIDADES O DESTREZAS:
CTE-El.10 - Conocimiento aplicado de informática industrial y comunicaciones.
CTE-El.6 - Capacidad para diseñar sistemas electrónicos analógicos, digitales y de potencia.
CTE-El.11 - Capacidad para diseñar sistemas de control y automatización industrial.
CTE-El.8 - Conocimientos de regulación automática y técnicas de control y su aplicación a la automatización industrial.
CTE-El.9 - Conocimientos de principios y aplicaciones de los sistemas robotizados.

COMPETENCIAS:
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.
CB5 - Que los estudiantes hayan desarrollado aquellas habilidades de aprendizaje necesarias para emprender estudios posteriores con un alto grado de autonomía.
CG.10 - Capacidad de trabajar en un entorno multilingüe y multidisciplinar.
CG.11 - Conocimiento, comprensión y capacidad para aplicar la legislación necesaria en el ejercicio de la profesión de Ingeniero Técnico Industrial.
CG.3 - Conocimiento en materias básicas y tecnológicas, que les capacite para el aprendizaje de nuevos métodos y teorías, y les dote de versatilidad para adaptarse a nuevas situaciones.
CG.4 - Capacidad de resolver problemas con iniciativa, toma de decisiones, creatividad, razonamiento crítico y de comunicar y transmitir conocimientos, habilidades y destrezas en el campo de la Ingeniería Industrial.
CG.5 - Conocimientos para la realización de mediciones, cálculos, valoraciones, tasaciones, peritaciones, estudios, informes, planes de labores y otros trabajos análogos.
CG.6 - Capacidad para el manejo de especificaciones, reglamentos y normas de obligado cumplimiento.
CG.7 - Capacidad de analizar y valorar el impacto social y medioambiental de las soluciones técnicas.

CONTENIDOS

Módulo I: CONCEPTOS BÁSICOS DE LOS SISTEMAS DE TIEMPO REAL

Introducción

El primer módulo se estructura en cuatro temas/unidades. Se explican los fundamentos de los sistemas de tiempo real y las características a tener en cuenta en su desarrollo. Se define el modelo de fallo y el concepto de tolerancia a fallos, indicando algunas técnicas para su resolución. Se termina el módulo explicando cómo se gestionan las excepciones (posibles errores) desde el punto de vista del lenguaje de programación y del propio sistema operativo (en tiempo real o no)

Contextualización

Estos contenidos deberán ser los primeros que el alumno tendrá que estudiar, y en este orden, antes de abordar contenidos más específicos sobre el diseño y programación de sistemas de tiempo real.

Los objetivos generales de este módulo son:

1. Introducir los conceptos de sistemas de tiempo real, su definición y ejemplos concretos de este tipo de sistemas
2. Revisar el modelo de desarrollo de un sistema de tiempo real en cuanto a análisis de requisitos, diseño y pruebas

3. Introducir el concepto de fallo y de disponibilidad, para abordar la fiabilidad de los sistemas en tiempo real usando los conceptos anteriores.
4. Revisar de manera detallada el concepto de fallo, incidiendo en los fallos provocados por el software (denominados excepciones) y la manera de gestionar dichos fallos.

Resultados de aprendizaje asociados a los contenidos

Los resultados de aprendizaje que se pretenden conseguir son los siguientes:

- Comprender el concepto de sistema de tiempo real (Tema/Unidad 1)
- Ser capaz de analizar el modelo de interacción/desarrollo de un sistema de tiempo real (Tema/Unidad 1)
- Entender y asimilar el concepto de fallo y disponibilidad, así como revisar las diferentes alternativas que existen para generara un sistema de tiempo real disponible y fiable (Tema/Unidad 2)
- Comprender los modelos y mecanismos de gestión de errores (excepciones) producidos en entidades software de tiempo real (Tema/Unidad 3).
- Reconocer las situaciones específicas que necesitan servicios redundantes en la implementación de aplicaciones informáticas de tiempo real y aplicar las técnicas y algoritmos adecuados (Tema/Unidad 3).

Bibliografía básica

Los contenidos correspondientes al primer módulo se encuentran definidos en su totalidad en el libro base de la asignatura, disponible en español y en inglés (versión actualizada). Se recomienda el uso del libro en inglés, por estar actualizado:

- Real Time Systems and Programming Languages. Alan Burns y Andy Wellings. Editorial: Addison-Wesley, 4ª Edición (2009). ISBN: 978-0-321-41745-9
- Sistemas de tiempo real y lenguajes de programación. Alan Burns y Andy Wellings. Editorial: Addison-Wesley, 3ª Edición (2003). ISBN: 84-7829-058-3

Dado que existen dos ediciones del libro, en la tabla mostrada a continuación se muestra la correlación de los elementos de estudio de cada unidad con cada versión del libro.

Unidad	Edición Inglés (2009)	Edición Español (2003)
Unidad 1	Capítulo 1 completo	Capítulos 1 y 2 completos
Unidad 2	Capítulo 2 completo	Capítulo 5 completo
Unidad 3	Capítulo 3 completo	Capítulo 6, excepto el punto 6.4

Materiales complementarios

Como ejercicios de autoevaluación se proponen cuestiones y soluciones a dichas cuestiones, accesibles a través del Curso virtual. Estos ejercicios de autoevaluación se usarán como entrenamiento caro a la parte correspondiente del examen presencial (que se empleará para evaluar lo conocimientos teóricos adquiridos en este módulo).

Módulo II: ASPECTOS DE LA CONCURRENCIA EN SISTEMAS DE TIEMPO REAL

Introducción

Este módulo explica los conceptos de ejecución concurrente, que se basa en el modelo de procesos/tareas/hebras. Se indica cómo se implementa la concurrencia en los tres modelos de programación (Java RT, Ada y C/Real Time POFIX) para pasar a describir como se implementan los mecanismos de comunicación y sincronización mediante variables compartidas y mensajes. A continuación, se introduce el concepto de acción atómica y la forma de implementarlas en los tres modelos de programación propuestos. Se finaliza el módulo explicando la gestión de acceso a los recursos y el modelo de interbloqueo.

Contextualización

En el contexto de la asignatura, los objetivos de este módulo son:

1. Entender el modelo de ejecución concurrente y los elementos subyacentes: hebras/procesos y tareas.
2. Profundizar en el soporte existente para distintos lenguajes de programación de los elementos de la programación concurrente.
3. Comprender los mecanismos de compartición de información básicos, soportados por los sistemas operativos que ejecutan el software de tiempo real.
4. Conocer como se definen acciones atómicas, como conjunto de acciones agrupadas que se ejecutan de manera única y atómica (no divisible).
5. Entender el modelo de interbloqueo de procesos.

Resultados de aprendizaje asociados a los contenidos

Los resultados de aprendizaje para cada este módulo son los siguientes:

- Asimilar el modelo de ejecución concurrente y los elementos asociados (Unidad/Tema 4)
- Ser capaz de emplear estructuras como monitores, regiones críticas y semáforos para el control de acceso a información compartida (Unidad/Tema 5)
- Manejar la estructura de mensajes de los sistemas operativos como medio específico para compartir información entre tareas/procesos (Unidad/Tema 6)
- Analizar la estructura de los mensajes POSIX y generarlos mediante programación de los mismos (Unidad/Tema 6)
- Entender el concepto de acción atómica, siendo capaz de generar modelos de transferencia asíncronas en varios lenguajes (Unidad/Tema 7)
- Asimilar el modelo de interbloqueo y sus implicaciones en el desarrollo de modelos de ejecución concurrente (Unidad/Tema 8)

Bibliografía básica

Los contenidos correspondientes al segundo módulo se encuentran definidos en su totalidad en el libro base de la asignatura, disponible en español y en inglés (versión actualizada). Se recomienda el uso del libro en inglés, por estar actualizado:

- Real Time Systems and Programming Languages. Alan Burns y Andy Wellings. Editorial: Addison-Wesley, 4ª Edición (2009). ISBN: 978-0-321-41745-9
- Sistemas de tiempo real y lenguajes de programación. Alan Burns y Andy Wellings. Editorial: Addison-Wesley, 3ª Edición (2003). ISBN: 84-7829-058-3

Dado que existen dos ediciones del libro, en la tabla mostrada a continuación se muestra la correlación de los elementos de estudio de cada unidad con cada versión del libro.

Unidad	Edición Inglés (2009)	Edición Español (2003)
Unidad 4	Capítulo 4, excepto puntos 4.7 y 4.9	Capítulo 7 completo.
Unidad 5	Capítulo 5, excepto puntos 5.7, 5.10 y 5.11	Capítulo 8 completo.
Unidad 6	Capítulo 6,excepto punto 6.9	Capítulo 9, excepto el punto 9.6 y en el punto 9.3 no se debe tener en cuenta el apartado sobre OCCAM2.
Unidad 7	Capítulo 7 completo	Capítulo 10 completo
Unidad 8	Capítulo 8 completo	Capítulo 11 completo

Material complementario

Como ejercicios de autoevaluación se proponen cuestiones y soluciones a dichas cuestiones, accesibles a través del Curso virtual. Estos ejercicios de autoevaluación se usarán como entrenamiento cara a la parte correspondiente del examen presencial (que se empleará para evaluar lo conocimientos teóricos adquiridos en este módulo).
Adicionalmente, las PED1 y PED2 (instalación/configuración de un sistema operativo en tiempo real y una implementación de RTSJ) necesitan de material complementario que se ubicará en el curso virtual:

- Entorno virtual para la distribución/instalación del sistema operativo en tiempo real (RTLinux basado en Debian).
- Extensiones necesarias para el desarrollo de la práctica (librerías Java), correspondientes a la implementación de referencia de RTSJ.
- Cualquier otro material necesario (documentación, software adicional, etc.).

Módulo III: TEMPORIZACIÓN DE LAS TAREAS EN UN SISTEMA DE TIEMPO REAL

Introducción

Este módulo explica los mecanismos de temporización que se pueden aplicar en un sistema de tiempo real, tomando como base el reloj del sistema en tiempo real. Una vez definidos los conceptos necesarios (timeouts, ámbitos temporales, etc.) se explica la forma de categorizar las tareas para su inclusión y priorización en la programación temporal del sistema. Una vez

hecho esto, se pueden aplicar varias estrategias de planificación basadas en prioridades o tiempos de ejecución. Se termina el módulo mostrando los detalles específicos de programación sobre entornos de ejecución asociados a los mecanismos de entrada/salida de dichos entornos.

Contextualización

En el contexto de la asignatura, los objetivos de este módulo son:

1. Entender la noción de tiempo en un sistema de tiempo real y los ámbitos temporales asociados.
2. Comprender el modelo de planificación de tareas de un sistema de tiempo real, y las diferentes alternativas existentes en los análisis de los tiempos de respuesta.
3. Profundizar en los mecanismos de programación de bajo nivel, como son el acceso al hardware de E/S y tareas del núcleo del sistema operativo.

Resultados de aprendizaje asociados a los contenidos

Los resultados de aprendizaje para cada este módulo son los siguientes:

- Analizar la definición de reloj de un sistema de tiempo real y ser capaz de definir los elementos asociados al tratamiento de un reloj en un sistema de tiempo real: retardos, timeouts, etc. (Unidad/Tema 9)
- Ser capaz de implementar pruebas de planificabilidad de conjuntos de procesos, mejorando la eficiencia de la ejecución de los procesos en cuanto al aseguramiento del cumplimiento de los tiempos de ejecución (Unidad/Tema 10)
- Analizar los problemas derivados del modelo de interbloqueo en la planificación de un conjunto de procesos con prioridades de ejecución especificadas como requisito (Unidad/Tema 10)
- Manejar los mecanismos de E/S proporcionados por los mecanismos de bajo nivel y su implicación en la planificabilidad de tareas (Unidad/Tema 11)

Bibliografía básica

Los contenidos correspondientes al tercer módulo se encuentran definidos en su totalidad en el libro base de la asignatura, disponible en español y en inglés (versión actualizada). Se recomienda el uso del libro en inglés, por estar actualizado:

- Real Time Systems and Programming Languages. Alan Burns y Andy Wellings. Editorial: Addison-Wesley, 4ª Edición (2009). ISBN: 978-0-321-41745-9
- Sistemas de tiempo real y lenguajes de programación. Alan Burns y Andy Wellings. Editorial: Addison-Wesley, 3ª Edición (2003). ISBN: 84-7829-058-3

Dado que existen dos ediciones del libro, en la tabla mostrada a continuación se muestra la correlación de los elementos de estudio de cada unidad con cada versión del libro.

Unidad	Edición Inglés (2009)	Edición Español (2003)
Unidad 9	Capítulo 9 completo	Capítulo 12, excepto el punto 12.8.

Unidad 10	Capítulo 11, excepto puntos 11.14, 11.15 y 11.16	Capítulo 13, excepto el punto 13.14.
Unidad 11	Capítulo 14	Capítulo 15, excepto los puntos 15.3 y 15.6.

Material complementario

Como ejercicios de autoevaluación se proponen cuestiones y soluciones a dichas cuestiones, accesibles a través del Curso virtual. Estos ejercicios de autoevaluación se usarán como entrenamiento caro a la parte correspondiente del examen presencial (que se empleará para evaluar lo conocimientos teóricos adquiridos en este módulo).
Adicionalmente, para la PED3 (desarrollo de una aplicación en tiempo real con RTSJ) se ubicará todo el material complementario en el curso virtual que sea necesario (documentación, software adicional, etc.). En cuanto a la PED4 (que se corresponde con el desarrollo de la integración de código en tiempo real con código no en tiempo real) necesita material complementario que se ubicará en el curso virtual:

- Extensiones necesarias para el desarrollo de la práctica (librerías Java), correspondientes al entorno de ejecución de tiempo no real.
- Cualquier otro material necesario (documentación, software adicional, etc.).

ORIENTACIONES PARA EL ESTUDIO DE LOS CONTENIDOS

El temario se estructura en tres módulos que agrupan los contenidos básicos a desarrollar en la asignatura: Introducción a los sistemas de tiempo real, Concurrencia y aspectos relativos (acceso a la información en los STR) y Temporización de tareas (ejecución y planificación e tareas en STR). Se han planteado en este orden para conseguir que el alumno, de forma gradual, pueda ir adquiriendo los conocimientos básicos necesarios para superar la asignatura. Dichos conocimientos básicos se demostrarán en el desarrollo de la práctica final, donde se aplicarán dichos conceptos en un sistema de tiempo real basado en Linux (RTLinux) y el desarrollo de ejemplos prácticos de programación en sistemas de tiempo real usando la especificación Real Time Specification for Java (RTSJ) y una de las implementaciones de referencia (IBM Real Time).
Para el estudio de la asignatura, se debe tener en cuenta que la disponibilidad del libro en español puede ser limitada y, por tanto, se recomienda usar la edición actualizada en inglés (además de ser una versión más reciente y corregida). En esta guía de estudio, para cada módulo, se muestra la correlación entre el temario y el material de estudio para ambas ediciones del libro. Se debe tener en cuenta que el uso del libro en inglés, le ayudará también a adquirir una de las competencias básicas en la titulación: Competencia general CG12 de “Comprensión de textos técnicos en lengua inglesa”.

METODOLOGÍA

De forma resumida la metodología docente se concreta en:
•Adaptada a las directrices del EEES.

•La asignatura no tiene clases presenciales. Los contenidos teóricos se impartirán a distancia, de acuerdo con las normas y estructuras de soporte telemático de la enseñanza en la UNED. Esta asignatura se impartirá a distancia, utilizando una plataforma de educación a través de Internet. Se organizarán foros de discusión para dudas y debates.

•El material docente incluye cuestionarios de autoevaluación sobre los contenidos de cada módulo y distintos tipos de actividades relacionadas con la asignatura: consulta bibliográfica, consulta de información en Internet, trabajos de análisis y resumen, y uso avanzado de herramientas software.

•Las actividades de aprendizaje se estructuran en torno al estado del arte en cada una de las materias del curso y a los problemas en los que se va a focalizar en los trabajos de evaluación a distancia.

La metodología docente se desarrolla de acuerdo con los siguientes principios:
•Además de adoptar la metodología docente general del grado, y en coherencia con el propósito de utilizar los sistemas interactivos de educación con fines pedagógicos y/o formativos, la asignatura diseñada se apoya en gran medida en los recursos educativos de este medio.

•La metodología del trabajo de la asignatura se basa en una planificación temporal de las actividades. Existirán diferentes módulos o unidades didácticas. Cada uno de éstos tendrá asociado unas unidades de aprendizaje y un material asignado (capítulos del libro base, artículos relacionados, direcciones adicionales de Internet, o cualquier otro material que se proporcione). Se asignará un período para cada módulo, en el que deberán realizar las actividades relacionadas con el mismo.

SISTEMA DE EVALUACIÓN

TIPO DE PRUEBA PRESENCIAL

Tipo de examen	Examen tipo test
Preguntas test	20
Duración del examen	60 (minutos)
Material permitido en el examen	
Calculadora no programable	
Criterios de evaluación	

La prueba consta de un test de 20 preguntas. Para superar la prueba se deberá obtener una puntuación mínima de 5 puntos. En cada pregunta del test se proponen cuatro respuestas de las cuales sólo una es correcta. Únicamente puntuarán las preguntas contestadas. Si la respuesta es correcta la puntuación será de 0.5 puntos y si es incorrecta restará 0.1 puntos

% del examen sobre la nota final	60
Nota del examen para aprobar sin PEC	0
Nota máxima que aporta el examen a la calificación final sin PEC	0
Nota mínima en el examen para sumar la PEC	0

Comentarios y observaciones

En el curso virtual se dispone de un conjunto de autoevaluaciones (una por cada unidad/tema asociado a su correspondiente módulo) que el estudiante puede utilizar/responder como test preparatorio del examen (por cada unidad).

PRUEBAS DE EVALUACIÓN CONTINUA (PEC)

¿Hay PEC? Si

Descripción

Existen cuatro, correspondientes a cuatro prácticas de laboratorio virtuales, a realizar tal y como se indica en el cronograma de trabajo de la asignatura:

1. Las dos primeras prácticas tienen una valoración total de un 5% sobre la nota final de la asignatura. Es obligatorio realizar estas dos prácticas para poder aprobar la asignatura, debiendo obtener una nota superior o igual a cinco en cada una de ellas.

2. La tercera práctica tiene una ponderación del 20% y como las pruebas anteriores, es obligatorio realizarla y obtener una nota superior o igual a cinco en la valoración de la prueba.

3. La cuarta práctica, PED4, se corresponde con el 10% de la nota y se debe desarrollar siguiendo las directrices que se especificaran en el curso virtual de la asignatura (junto con todos los recursos asociados a la misma). Es obligatorio hacerla para poder aprobar la asignatura (de manera independiente al examen), debiendo obtenerse una nota superior o igual a cinco en la valoración de la prueba.

Criterios de evaluación

Todas las prácticas son obligatorias, y se debe obtener al menos una nota superior o igual a cinco en cada una de ellas para superarlas.

Ponderación de la PEC en la nota final	40
Fecha aproximada de entrega	Disponibles en el curso virtual, de acuerdo a la fechas del calendario del segundo cuatrimestre

Comentarios y observaciones

- Plazo 1 (convocatoria ordinaria): Prácticas de laboratorio recibidas antes del 30 de Mayo.
- Plazo 2 (convocatoria extraordinaria): Prácticas de laboratorio recibidas con posterioridad al 30 de Mayo y antes del 20 de Septiembre.

OTRAS ACTIVIDADES EVALUABLES

¿Hay otra/s actividad/es evaluable/s?	No
Descripción	
Criterios de evaluación	
Ponderación en la nota final	0
Fecha aproximada de entrega	
Comentarios y observaciones	

¿CÓMO SE OBTIENE LA NOTA FINAL?

Para que un alumno pueda aprobar la asignatura deberá haber superado el examen teórico (un cinco sobre diez, como mínimo) y haber aprobado las prácticas de laboratorio virtuales (un cinco sobre diez, para cada una de ellas). Si se cumplen estas condiciones, la nota final se calcula de la siguiente manera

Nota final = 0.6 x Nota_Examen + 0.4 x Nota_Prácticas_Laboratorio (PECs)

Solo se tendrán en cuenta aquellas calificaciones que tengan aprobadas tanto el examen como todas las prácticas.

BIBLIOGRAFÍA BÁSICA

ISBN(13):9780321417459
Título:REAL-TIME SYSTEMS AND PROGRAMMING LANGUAGES4
Autor/es:Andrew J Wellings ; Burns, Alan ;
Editorial:ADDISON-WESLEY

ISBN(13):9788478290581
Título:SISTEMAS DE TIEMPO REAL Y LENGUAJES DE PROGRAMACIÓN3ª
Autor/es:Wellings, Andy ; Burns, Alan ;
Editorial:PEARSON ADDISON-WESLEY

El libro base de Burns recoge todo del temario de la asignatura y presenta de una forma clara y concisa todos los conceptos clave. En cada capítulo se introducen los conceptos de una manera sencilla, progresiva y acompañados de ejemplos aclaratorios finalizando con un resumen que recoge los conceptos fundamentales presentados al alumno. Adicionalmente se le entregarán al estudiante una colección de enlaces y recursos relevantes que completaran la bibliografía básica con contenidos de actualidad sobre la programación de sistemas de tiempo real.

La disponibilidad del libro en español puede ser limitada y por tanto, se recomienda usar la edición actualizada en inglés (además de ser una versión más reciente y corregida). En la guía de estudio (segunda parte) se muestra la correlación entre el temario y el material de estudio para ambas ediciones del libro. Se debe tener en cuenta que el uso del libro en inglés, le ayudará también a aquirir una de las competencias básicas en la titulación: Competencia general CG12 de “Comprensión de textos técnicos en lengua inglesa”

BIBLIOGRAFÍA COMPLEMENTARIA

ISBN(13):9780137142989
Título:REAL-TIME JAVA PROGRAMMING: WITH JAVA RTSPrimera edición
Autor/es:Greg Bollella ; Eric J. Bruno ;
Editorial:Ed. Prentice-Hall

ISBN(13):9780201633924
Título:PROGRAMMING WITH POSIX THREADSPrimera edición
Autor/es:D. Butenhof ;
Editorial:ADDISON - WESLEY

ISBN(13):9780470844373
Título:CONCURRENT AND REAL-TIME PROGRAMMING IN JAVAPrimera edición
Autor/es:Andrew J Wellings ;
Editorial:JOHN WILLEY & SONS

ISBN(13):9780521866972
Título:CONCURRENT AND REAL-TIME PROGRAMMING IN ADATercera Edición
Autor/es:Andrew J Wellings ; Burns, Alan ;
Editorial:CAMBRIDGE UNIVERSITY PRESS..

ISBN(13):9781119482758
Título:CONCURRENT, REAL-TIME AND DISTRIBUTED PROGRAMMING IN JAVA: THREADS, RTSJ AND RMI2017
Autor/es:Badr Benmammar ;
Editorial:Wiley-ISTE

ISBN(13):9781419656491
Título:REAL-TIME JAVA PLATFORM PROGRAMMING Segunda edición
Autor/es:Peter C Dibble ;
Editorial:BookSurge Publishing

El estudiante puede consultar la siguiente bibliografía con el fin de aclarar o extender los conocimientos que debe adquirir a lo largo del curso, y más en concreto en lo concerniente a Java RTS, Ada y POSIX para la realización de las actividades prácticas:

•Real-Time Java Programming: With Java RTS. Eric J. Bruno y Greg Bollella. Editorial: Prentice Hall (2009). ISBN: 978-0137142989

El libro se centra en la implementación de Java RTS, como motor de desarrollo para sistemas de tiempo real, incluyendo capítulos específicos sobre el recolector de basura, eficiencia de la implementación y detalles sobre la planificación y gestión temporal de las tareas, además de los modelos de memoria para RTSJ

•Real-Time Java Platform Programming (Segunda Edición). Dr. Peter C Dibble. Editorial: BookSurge Publishing (2008). ISBN: 978-1419656491

El libro describe los aspectos relevantes de la especificación RTJS, focalizando en temas como la integración con tareas de tipo Non-RT (no real time), accesos a los diferentes tipos de heap de memoria, uso de la memoria “sucia” para acceder a distintos dispositivos hardware o detallar los errores más habituales en el uso de Java RT para el desarrollo de aplicaciones.

•Concurrent and Real-Time Programming in Java. Andrew Wellings. Editorial: Willey (2004). ISBN: 978-0470844373

El libro se centra en los aspectos de programación concurrente que proporciona el modelo RTJS, y en particular en el modelo de gestión de memoria y control de recursos compartidos. Dispone de un ejemplo práctico muy significativo, en el área de la automoción, presentando un caso de desarrollo de un sistema de control de crucero.

•Concurrent and Real-Time Programming in Ada. Alan Burns y Andy Wellings. Editorial: Cambridge University Press (Tercera Edición) (2007). ISBN: 978-0521866972

Se presenta el lenguaje de programación ADA,y en concreto las características que proporciona para la programación de sistemas en tiempo real y computación distribuida. ADA es un lenguaje pensado para ser escalable y robusto, con una fuerte estructura de planificación adecuada para los sistemas de tiempo resl. Se muestran los mecanismos proporcionados para la gestión de memoria compartida y la planificación de tiempos de las tareas.

•Programming with POSIX threads. D. Butenhof. Editorial: Addison-Wesley (1997). ISBN: 978-0201633924

En este libro se describe el modelo de tareas (threads) usados en los sistemas Unix (POSIX) y que formalmente se denomina como Pthreads (de POSIX threads). Se describen los mecanismos de definición e intercomunicación de dichas tareas y su aplicabilidad al desarrollo de sistemas de computación distribuidos y de tiempo real.

RECURSOS DE APOYO Y WEBGRAFÍA

Curso virtual

Para alcanzar todos los objetivos propuestos, el curso se va a articular, como ya se ha comentado, a través de una plataforma especialmente diseñada para facilitar el trabajo colaborativo en Internet (basada en comunidades virtuales), desarrollada por la Sección de Innovación del Centro de Innovación y Desarrollo Tecnológico de la UNED: aLF, ubicada en https://www.innova.uned.es.

La plataforma de e-Learning aLF, proporcionará el soporte requerido para gestionar los procesos de enseñanza y aprendizaje, compartir documentos y enlaces de interés, crear y participar en comunidades temáticas y grupos de trabajo específicos, realizar proyectos de diversa naturaleza, organizar el trabajo mediante agendas compartidas e individuales, acceder y publicar noticias de interés, etc.

La plataforma de aprendizaje en Internet permitirá realizar el seguimiento de las actividades del curso, así como estar al tanto de cualquier información o documentación de interés relacionada con el mismo. Para poder utilizar esta plataforma y para mantener un contacto personal con el alumnado se necesitará una dirección de correo electrónico suministrada por el Centro de Servicios Informáticos de la Uned. La filosofía de uso es bien sencilla. Todas las interacciones se hacen a través de enlaces. Por lo tanto, con sólo seguir dichos enlaces se podrá acceder a foros de discusión, documentos de compañeros, etc.
Una vez familiarizados con su uso, es importante tener en cuenta que todas las novedades, instrucciones, actividades se van a publicar utilizando este medio, por tanto, el alumno debe entrar en el grupo frecuentemente para ver si hay alguna novedad en el curso. Si, además, tiene activados ciertos avisos, podrá recibir notificaciones en el correo electrónico utilizado para acceder a la plataforma de los mensajes publicados en los foros, los documentos subidos, las citas puestas en el calendario, por lo que tendrá una información instantánea de todo lo que acontece en la plataforma.
Se ofrecerán las herramientas necesarias para que, tanto el equipo docente como el alumnado, puedan compaginar el trabajo individual y el aprendizaje colaborativo.
Software para prácticas.
Se ubicará en la propia plataforma, en el área correspondiente, o bien se darán los enlaces correspondientes de las ubicaciones originales donde descargar tanto el software como los correspondientes manuales.

PRÁCTICAS DE LABORATORIO

¿Hay prácticas en esta asignatura de cualquier tipo (en el Centro Asociado de la Uned, en la Sede Central, Remotas, Online,..)?

Si

CARACTERÍSTICAS GENERALES

Presenciales: No

Obligatorias: Sí

Es necesario aprobar el examen para realizarlas: No

Fechas aproximadas de realización: Son cuatro prácticas online que se hacen en la primera semana de marzo, ultima semana de marzo, principio de abril (mes entero) y principio de mayo (mes entero). Las fechas concretas se muestran en el calendario del curso virtual.

Se guarda la nota en cursos posteriores si no se aprueba el examen:
(Si es así, durante cuántos cursos)
No, solo se mantiene la nota en el curso lectivo actual. Si se aprueban en la convocatoria ordinaria (Junio) se guarda la nota para la convocatoria extraordinaria (Septiembre)

Cómo se determina la nota de las prácticas: Cada práctica se evalúa de 0 a 10, y en el enunciado de la práctica se determinan los criterios/entregables de cada una. Al ser prácticas online y dinámicas, estos criterios varían dependiendo de las actividades a realizar en cada práctica.

REALIZACIÓN

Lugar de realización (Centro Asociado/ Sede central/ Remotas/ Online): Online

N.º de sesiones: No aplica

Actividades a realizar: Tareas y pasos descritos en cada uno de los enunciados de la práctica.

OTRAS INDICACIONES: Es muy importante leerse los enunciados de las prácticas, ya que se usan recursos en la nube en la mayoría de ellas, que constituyen el entorno online para la realización de las mismas.

IGUALDAD DE GÉNERO

En coherencia con el valor asumido de la igualdad de género, todas las denominaciones que en esta Guía hacen referencia a órganos de gobierno unipersonales, de representación, o miembros de la comunidad universitaria y se efectúan en género masculino, cuando no se hayan sustituido por términos genéricos, se entenderán hechas indistintamente en género femenino o masculino, según el sexo del titular que los desempeñe.