Sistemas de tiempo real

Curso Académico: (2020 / 2021) **Fecha de revisión: 08/07/2020 14:57:07**

Departamento asignado a la asignatura:

Coordinador/a: FERNANDEZ MUÑOZ, JAVIER

Tipo: Optativa Créditos ECTS : 6.0

Curso : 4 Cuatrimestre :

REQUISITOS (ASIGNATURAS O MATERIAS CUYO CONOCIMIENTO SE PRESUPONE)

Diseño de Sistemas Operativos
Arquitectura de Computadores

OBJETIVOS

El objetivo de este curso es que el estudiante conozca los principales conceptos relacionados con el diseño y construcción de sistemas con restricciones de tiempo asociadas, es decir sistemas de tiempo real.
Para lograr este objetivo el alumno debe adquirir una serie de competencias genéricas, conocimientos, capacidades y actitudes

Competencias básicas:
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.

Competencias específicas para mención de ingenieria de computadores:
CEIC2.- Capacidad de desarrollar procesadores específicos y sistemas empotrados, así como desarrollar y optimizar el software de dichos sistemas.

Asi como las siguientes competencias del programa: a, c, d, e, g, i, j, k

Competencias Transversales/Genéricas:
- Capacidad de análisis y síntesis (PO a)
- Capacidad de organizar y planificar (PO a)
- Resolución de problemas (PO c)
- Trabajo en equipo (PO d)
- Capacidad de aplicar los conocimientos en la práctica (PO a, c)

Competencias Específicas:
- Cognitivas (Saber) (PO a)

1. Comprender los conceptos y problemas específicos de los Sistemas de Tiempo Real y las características que los diferencian de otros sistemas informáticos.
2. Conocer los métodos más importantes que utilizan para desarrollar Sistemas de Tiempo Real, su organización software y comprender sus principios y formas de aplicación.
3. Conocer los aspectos de planificación de sistemas de tiempo real.
4. Conocer la funcionalidad de los Sistemas de Tiempo Real, así como su estructura y funcionamiento interno.
5. Conocer algunas herramientas (lenguajes de programación y sistemas operativos) adecuadas para desarrollar Sistemas de Tiempo Real.
6. Conocer las alternativas existentes para la construcción de sistemas basados en microprocesador para entornos empotrados
7. Conocer las técnicas de desarrollo sobre procesadores especificos y de sistemas empotrados.
8. Conocer los métodos para analizar, evaluar y seleccionar las plataformas hardware/software empotradas y de tiempo real.

- Procedimentales/Instrumentales (Saber hacer) (PO e, g, j, k)
1. Diseñar y evaluar un sistema de tiempo real.
2. Capacidad para calcular planificadores de tiempo real cíclicos y monótonos basados en prioridad y para comprobar la corrección de una aplicación de tiempo real.
3. Diseñar y desarrollar aplicaciones de tiempo real usando sistemas operativos de tiempo real.
4. Capacidad de diseñar y construir sistemas basados en microprocesador para entornos empotrados y sistemas de tiempo real.
5. Capacidad de diseñar e implementar sistemas que utilizan procesadores especificos y de sistemas empotrados.
6. Capacidad de analizar, evaluar y seleccionar las plataformas hardware/software empotradas y de tiempo real.
7. Usar herramientas (lenguajes de programación y sistemas operativos) adecuados para la realización de sistemas de tiempo real distribuidos, comprobando su corrección temporal
- Actitudinales (Ser) (PO c, i)
1. Capacidad para generar nuevas ideas (creatividad)
2. Actitud crítica respecto a los sistemas de tiempo real actuales
3. Preocupación por la calidad de los sistemas de tiempo real
4. Motivación por investigar soluciones a nuevos problemas de sistemas de tiempo real.
5. Capacidad de aprendizaje y motivación de logro.

DESCRIPCIÓN DE CONTENIDOS: PROGRAMA

Los descriptores asociados con la asignatura son:
Tiempo real, sistemas operativos en tiempo real, planificación cíclica y basada en prioridades, concurrencia de tareas, reparto de tiempo de recursos, aplicaciones de tiempo real, sincronización, tolerancia a fallos, comportamiento determinista, diseño con microprocesadores generales y especificos, integracion hardware/software, systemas empotados.

Temario:
1. Introducción a los sistemas de tiempo real
    - Definición de sistemas de control, sistemas de tiempo real y sistemas empotrados.
    - Clasificación de sistemas de tiempo real.
    - Formalización de los requisitos temporales.

2. Sistemas de tiempo real empotrados.
    - Arquitecturas hardware para empotrados.
    - Frameworks software para empotrados y tiempo real.
    - Sistemas operativos de tiempo real y empotrados.
    - Diseño con microprocesadores de arquitecturas empotradas
    - Diseño de sistemas de tiempo real y empotrados

3. Planificación cíclica de tareas
    - Sistemas cíclicos y sistemas multiprogramados
    - Modelo simple de tareas.
    - Diseño de planificadores cíclicos de tareas.

4. Planificación con prioridades de tareas
    - Diseño de planificadores con prioridades.
    - Protocolo de herencia de prioridades
    - Protocolo de techo de prioridad
    - Extensiones POSIX para planificación con prioridades

5. Planificación dinámica de tareas y calidad de servicio
    - Sistemas no críticos y sistemas multimedia
    - Gestión de recursos y calidad de servicio
    - Planificación dinamica de los recursos.

6.- Certificación y desarrollo basado en modelos
    - Estándares de certificación (Ej: ECSS, DO-178C)
    - Metodología de desarrollo basado en modelos.

7.- Panorámica de los sistemas de control

ACTIVIDADES FORMATIVAS, METODOLOGÍA A UTILIZAR Y RÉGIMEN DE TUTORÍAS

Clases Teóricas: 1.5 ECTS. Tienen por objetivo alcanzar las competencias específicas cognitivas de la asignatura. En ellas se presentarán los conocimientos que los alumnos deben adquirir. Para facilitar su desarrollo los alumnos recibirán las notas de clase y tendrán textos básicos de referencia que les permita completar y profundizar en aquellos temas en los cuales estén más interesados. (CB1, CEIC2 y POs a, j)
Clases Prácticas: 1.5 ECTS. Desarrollan las competencias específicas instrumentales y la mayor parte de las transversales, como son la de trabajo en equipo, capacidad de aplicar los conocimientos a la práctica, de planificar y organizar y de análisis y síntesis. También tienen por objetivo desarrollar las capacidades específicas actitudinales. Consisten en el diseño y desarrollo de una aplicación de tiempo real utilizando planificación cíclica y basadas en prioridades. Además se evaluan diversos entornos empotrados. Se desarrollan en grupos de trabajo y en clases informáticas con presencia de profesor. (CB1, CEIC2 y POs a, c, d, e , g, i, j, k)
Realización de Actividades Académicas Dirigidas Con presencia del profesor: 1 ECTS (CB1, CEIC2 y POs a, c, d, e, g, i, j, k)
- Resolución de ejercicios y casos prácticos de forma participativa en la clase. Puede incluir además el estudio de ejemplos de sistemas operativos, común de las respuestas a los ejercicios y corrección conjunta, presentación de trabajos, etc.
Estudio del alumno: 1.5 ETC. (CB1, CEIC2 y POs a, i, j)
- Ejercicios y lecturas complementarias propuestas por el profesor.
- Estudio personal.
Ejercicios y Examen: 0.5 ECTS. Tienen por objeto incidir y complementar en el desarrollo de las capacidades específicas cognitivas y procedimentales. (CB1, CEIC2 y POs a, c, e, g)

SISTEMA DE EVALUACIÓN

Peso porcentual del Examen/Prueba Final:	40
Peso porcentual del resto de la evaluación:	60

La evaluación tiene como misión conocer el grado de cumplimiento de los objetivos de aprendizaje, por ello se valorará todo el trabajo del alumno mediante la evaluación continua de sus actividades a través de los ejercicios y exámenes, trabajos prácticos y otras actividades académicas dirigidas.

CONVOCATORIA ORDINARIA
El 40% de la calificación final se obtendrá de un examen final global de todo el contenido de la asignatura. (CB1, CEIC2 y POs a, c, e)
El 60% restante se obtendrá en el proceso de evaluación continua.
El proceso de evaluación continua incluye:
+ Prácticas con un 40% del peso de la nota (CB1, CEIC2 y POs a, b, c, d, e, g, i, k)
NOTA: Las prácticas son obligatorias
+ Ejercicios teóricos puntuables con un 10% de la nota (CB1, CEIC2 y POs a, c, e)
NOTA: Los ejercicios teóricos puntuables son obligatorios
+ Actividades Académicas Dirigidas con un 10% de la nota (CB1, CEIC2 y POs j, k)

Se considera que se ha seguido la evaluación continua si se ha entregado las prácticas y se ha realizado las actividades académicas indicadas como obligatorias.

Alternativamente, el examen final representará el 60% de la calificación para todos aquellos alumnos que decidan no integrarse en el sistema anterior de evaluación continua.

CONVOCATORIA EXTRAORDINARIA
La calificación de los estudiantes en la convocatoria extraordinaria se ajustará a las siguientes reglas:
a) Si siguió el proceso de evaluación continua el examen final será el 40% de la nota y la nota de la evaluación continua será el 60% restante.
b) Si no se ha realizado el proceso de evaluación continua se podrá realizar un examen final que será el 100% de la calificación total de la asignatura.
c) Aunque el estudiante hubiera seguido el proceso de evaluación continua, tendrá derecho a ser calificado en la forma indicada en el apartado b) cuando le resulte más favorable.

Tanto la convocatoria ordinaria como la extraordinaria siguen la normativa oficial de la Universidad al respecto.

En caso de detectarse una copia entre dos prácticas (copiados y copiadores), los alumnos involucrados perderán las calificaciones obtenidas por evaluación continua. Según la gravedad del caso, además, supondrá la apertura de un procedimiento administrativo.

BIBLIOGRAFÍA BÁSICA

- Alan Burns & Andy Wellings Sistemas de Tiempo Real y lenguajes de Programación. Tercera edición (b), Pearson Educación, 2005

- Alan Burns and Andy Wellings Real-Time Systems and Programming Languages: Ada, Real-Time Java and C/Real-Time POSIX (4th Edition) , Pearson Education , 2009

BIBLIOGRAFÍA COMPLEMENTARIA

- Bill Gallmeister Posix, O'Reilly, 1995.

- Bradford Nichols, Dick Butlar, Jacqueline Farrell Pthreads programming, O'Reilly, 1996..

- Hermann Kopetz Real-Time Systems. Design Principles for Distributed Embedded Applications, Kluwer, 1997.

- J.P. Cohoon & J.W. Davidson The C Programming Language. 2nd. ed (ANSI-C), Prentice-Hall, 1991.

- John Barnes Programming in Ada 95, 2nd. ed. Addison-Wesley, 1998..