

Guía docente

330232 - PCTR - Programación Concurrente y en Tiempo Real

Última modificación: 25/04/2024

Unidad responsable: Escuela Politécnica Superior de Ingeniería de Manresa
Unidad que imparte: 750 - EMIT - Departamento de Ingeniería Minera, Industrial y TIC.
Titulación: GRADO EN INGENIERÍA DE SISTEMAS TIC (Plan 2010). (Asignatura obligatoria).
Curso: 2024 **Créditos ECTS:** 6.0 **Idiomas:** Catalán

PROFESORADO

Profesorado responsable: ANTONI ESCOBET CANAL

Otros: SEBASTIAN VILA MARTA

COMPETENCIAS DE LA TITULACIÓN A LAS QUE CONTRIBUYE LA ASIGNATURA

Específicas:

1. La capacidad de analizar, diseñar y mantener aplicaciones informáticas así como el conocimiento de los principios y herramientas de la ingeniería del software y su aplicación.
2. El conocimiento y la capacidad de usar las herramientas y la instrumentación existentes para el análisis, el diseño, el desarrollo y la verificación de sistemas electrónicos, informáticos y de comunicaciones.
3. La capacidad para desarrollar las actividades propias del grado considerando los estándares, reglamentos y normas reguladoras correspondientes.
4. El conocimiento de las bases de la programación concurrente, paralela y distribuida así como la capacidad de aplicar-en los problemas que proceda.
5. La capacidad para analizar, diseñar e implementar, seleccionar y usar sistemas de tratamiento de datos, control y automatización? o en tiempo real, especialmente en sistemas empujados.
6. Capacidad para desarrollar las actividades propias del grado considerando los estándares, reglamentos y normas reguladoras correspondientes.

Transversales:

7. TRABAJO EN EQUIPO - Nivel 3: Dirigir y dinamizar grupos de trabajo, resolviendo posibles conflictos, valorando el trabajo hecho con las otras personas y evaluando la efectividad del equipo así como la presentación de los resultados generados.
8. USO SOLVENTE DE LOS RECURSOS DE INFORMACIÓN - Nivel 3: Planificar y utilizar la información necesaria para un trabajo académico (por ejemplo, para el trabajo de fin de grado) a partir de una reflexión crítica sobre los recursos de información utilizados.
9. APRENDIZAJE AUTÓNOMO - Nivel 3: Aplicar los conocimientos alcanzados en la realización de una tarea en función de la pertinencia y la importancia, decidiendo la manera de llevarla a cabo y el tiempo que es necesario dedicarle y seleccionando las fuentes de información más adecuadas.

METODOLOGÍAS DOCENTES

La asignatura se estructura en dos clases de dos horas por semana. De estas cuatro horas presenciales semanales una se dedica a presentar los principales contenidos de manera expositiva, la segunda a la resolución de problemas bajo demanda del estudiantado y las dos restantes a resolver problemas prácticos en el laboratorio informático.

Al estudiante se le indican semanalmente las tareas de estudio y solución de problemas que debe hacer. Estas tareas se aconseja hacerlas, al menos parcialmente, trabajando en equipo. Periódicamente se evalúa el progreso de cada estudiante individualmente.

OBJETIVOS DE APRENDIZAJE DE LA ASIGNATURA

Después de superar esta asignatura el estudiante:

1. Conocerá los principios del tratamiento de información en tiempo real y podrá diseñar e implementar aplicaciones de complejidad moderada sometidas a condiciones de tiempo real.
2. Conocerá los paradigmas de la programación concurrente, paralela y distribuida y los podrá aplicar a la solución de problemas.
3. Sabrá obtener e interpretar información técnica y será capaz de comunicar resultados de forma oral y escrita.
4. Poder redactar emorias técnicas sencillas, también en una tercera lengua, y presentarlas oralmente.

HORAS TOTALES DE DEDICACIÓN DEL ESTUDIANTE

Tipo	Horas	Porcentaje
Horas aprendizaje autónomo	90,0	60.00
Horas grupo grande	30,0	20.00
Horas grupo pequeño	30,0	20.00

Dedicación total: 150 h

CONTENIDOS

TEMA 1: Introducción a la Concurrencia

Descripción:

Se introduce el concepto de programa concurrente. Se presentan varios modelos de concurrencia y sus principales características. Se hace énfasis en los sistemas distribuidos y en el modelo CSP.

Palabras clave: Concurrencia, paralelismo, modelo de concurrencia, sistema distribuido, CSP.

Objetivos específicos:

Una vez asumido el tema:

1. El estudiante debe entender la noción de concurrencia y tener una buena perspectiva sobre sus usos, ventajas y desventajas así como conocer los principales modelos existentes.

Actividades vinculadas:

Todas las que constan.

TEMA 2: Programación concurrente

Descripción:

El objetivo del tema es dar a conocer al estudiante los principales conceptos y prácticas en el ámbito de la programación concurrente sobre un modelo de tipo Actor. A tal efecto se usa Erlang como lenguaje de programación y herramienta de trabajo para ir descubriendo los principales conceptos.

Palabras clave: Programación concurrente, proceso, comunicación entre procesos, paso de mensajes, arquitectura cliente-servidor, recursos compartidos, deadlock, inanición, actor.

Paraules clau: Programació concurrent, procés, comunicació entre processos, pas de missatges, arquitectura client-servidor, recursos compartits, deadlock, inanició, actor.

Objetivos específicos:

Una vez asumido el tema, el estudiante:

1. Debe saber escribir pequeños programas concurrentes usando Erlang.
2. Ha de entender los conceptos más importantes relacionados con la concurrencia.
3. Debe conocer los principales riesgos y ventajas que se asumen en implementaciones concurrentes.

Actividades vinculadas:

Todas las que constan.

TEMA 3: Sistemas en Tiempo real

Descripción:

El tema presenta y ejemplifica el concepto de sistema en tiempo real y las diversas variantes. Se introduce el criterio de corrección para sistemas de tiempo real y también se dibuja la manera de presentar este tema en esta asignatura.

Palabras clave: Tiempo real. Tiempo real soft y hard. Corrección.

Objetivos específicos:

Una vez asumido el tema, el estudiante:

1. Debe conocer el concepto de sistema en tiempo real y tienen que poder discernir qué sistemas tienen condiciones de tiempo real.
2. Deben entender las características de los diferentes tipos de sistemas en tiempo real.
3. Deben comprender qué condiciones sobre la corrección añaden los sistemas de tiempo real.

Actividades vinculadas:

Todas las que constan.

TEMA 4: Diseño e Implementación de Sistemas en Tiempo Real

Descripción:

El objetivo de este tema es presentar algunos de los modelos y herramientas habituales para los sistemas en tiempo real así como su utilidad de cara a especificar e implementar estos sistemas.

Particularmente se presentarán los "Timed automata" y también los "protothreads". Palabras clave: Implementación de sistemas en tiempo real. Protothreads. Timed automata. Sistemas conducidos por eventos. Sistemas conducidos por el tiempo.

Objetivos específicos:

Una vez asumido el tema, el estudiante:

1. Podrá diseñar e implementar sistemas en tiempo real de complejidad media.

Actividades vinculadas:

Todas las que constan.

TEMA 5: Planificación de procesos de Tiempo Real

Descripción:

El objetivo de este tema es presentar la problemática del scheduling de procesos bajo condiciones de tiempo real así como las diversas soluciones propuestas para este problema. Simultáneamente se situará el problema en el contexto de los sistemas operativos.

Palabras clave: Algoritmos de scheduling. Scheduling de procesos en tiempo real.

Scheduling estático y dinámico. Sistemas operativos para tiempo real.

Objetivos específicos:

Una vez alcanzado el tema, el estudiante:

1. Conocerá las principales dificultades de la planificación de procesos en tiempo real y algunos de los algoritmos que permiten acercarse a la solución óptima.
2. Podrá implementar algunos de los algoritmos de planificación de procesos en tiempo real.

Actividades vinculadas:

Todas las que constan.

ACTIVIDADES

ACTIVIDAD 1: EXAMEN

Descripción:

La asignatura contempla un examen final que consiste en un conjunto de ejercicios a resolver individualmente sobre papel sin apoyo de ningún tipo de material y en un tiempo acotado.

Entregable:

Se entrega la solución individual del examen y se evalúa la misma.

Dedicación: 2h

Grupo grande/Teoría: 2h

ACTIVIDAD 2: ESTUDIO DE CONTENIDOS

Descripción:

El estudio de los contenidos es la actividad individual o colectiva que conduce a entender y asumir los conocimientos, vocabulario y técnicas que forman parte de los contenidos de la asignatura.

Material:

Los materiales de apoyo son:

- Referencias principales de la asignatura.
- Colección de problemas de la asignatura.

Dedicación: 25h

Aprendizaje autónomo: 25h

ACTIVIDAD 3: CLASE EXPOSITIVA

Descripción:

Son clases presenciales específicamente dedicadas a la comprensión de los contenidos de la asignatura, especialmente aquellos de carácter mas bien teórico.

Material:

Los materiales de apoyo son:

- Referencias principales de la asignatura.
- Colección de problemas de la asignatura.

Dedicación: 14h

Grupo grande/Teoría: 14h

ACTIVIDAD 4: CLASE DE PROBLEMAS

Descripción:

Son clases presenciales específicamente dedicadas a la resolución de problemas. Se hacen en un aula ordinaria y son complementarias de la actividad en el laboratorio. Son clases que requieren la participación de los estudiantes.

Material:

Los materiales de apoyo son:

- Referencias principales de la asignatura.
- Colección de problemas de la asignatura.

Dedicación: 14h

Grupo grande/Teoría: 14h

ACTIVIDAD 5: TRABAJO DE LABORATORIO

Descripción:

El estudiante tiene como objetivo la solución de pequeñas prácticas que complementan los contenidos y colaboran en la mejor comprensión de los mismos. Las prácticas se realizan en el laboratorio y conllevan la implementación real de programas sobre el computador y su comprobación. La actividad puede comportar la finalización de las prácticas en tiempo de aprendizaje autónomo.

Material:

Los materiales de apoyo son:

- Referencias principales de la asignatura.
- Colección de problemas de la asignatura.
- Manuales del software utilizado.

Entregable:

Periódicamente se entregan los resultados de diferentes prácticas que se van haciendo durante el curso.

Dedicación: 65h

Grupo pequeño/Laboratorio: 30h

Aprendizaje autónomo: 35h

ACTIVIDAD 6: RESOLUCIÓN DE PROBLEMAS

Descripción:

Es una actividad que hace el estudiante autónomamente y que consiste en la solución de problemas de programación, generalmente sin ser necesario el apoyo del computador.

Material:

Los materiales de apoyo son:

- Referencias principales de la asignatura.
- Colección de problemas de la asignatura.

Entregable:

La actividad conlleva la entrega durante el curso de algunos problemas que se corrigen convenientemente y forman parte de la evaluación de la asignatura.

Dedicación: 30h

Aprendizaje autónomo: 30h

SISTEMA DE CALIFICACIÓN

La calificación se realiza en base a 3 elementos:

1. La evaluación del trabajo autónomo del estudiante (A). Este componente contiene tanto el progreso hecho en los aspectos teóricos como en los prácticos. Su medición se realiza en base de ejercicios obligatorios entregados durante el curso.
2. La evaluación del trabajo práctico (P). Se realiza a partir de la entrega de las prácticas que se van realizando durante el curso.
3. La evaluación final (F). Se hace a través de un examen final que tiene naturaleza global e integra todos los conocimientos y destrezas de carácter teórico adquiridos durante el curso.

A partir de estos elementos se calcula la nota final con las siguientes ponderaciones:

$$\text{Final} = 0.20 A + 0.45 P + 0.35 F$$

NORMAS PARA LA REALIZACIÓN DE LAS PRUEBAS.

Las actividades se realizarán siguiendo los usos y costumbres del trabajo académico y, particularmente, se respetarán las siguientes pautas:

1. Aquellas actividades que sean explícitamente declaradas como individuales, sean de naturaleza presencial o no, se realizarán sin ninguna colaboración por parte de otras personas.
2. Las fechas, formatos y demás condiciones de entrega que se establezcan serán de obligado cumplimiento.
3. El Uso del laboratorio informático se reservará exclusivamente para las actividades académicas y en ningún caso se podrá hacer un uso abusivo.

BIBLIOGRAFÍA

Básica:

- Brooke, P. J.; Paige, R. F. Practical distributed processing [en línea]. London: Springer, 2008 [Consulta: 07/10/2022]. Disponible a: <https://ebookcentral-proquest-com.recursos.biblioteca.upc.edu/lib/upcatalunya-ebooks/detail.action?pq-origsite=primo&docID=6310529>. ISBN 9781846288401.
- Cesarini, F.; Thomson, S. Erlang programming [en línea]. Farnham: O'Reilly, 2009 [Consulta: 27/05/2022]. Disponible a: <https://ebookcentral-proquest-com.recursos.biblioteca.upc.edu/lib/upcatalunya-ebooks/detail.action?docID=443218>. ISBN 9780596518189.
- Kopetz, H. Real-time systems: design principles for distributed embedded applications [en línea]. 2n ed. Nova York: Springer, 2011 [Consulta: 10/06/2022]. Disponible a: <https://link-springer-com.recursos.biblioteca.upc.edu/book/10.1007/b116085>. ISBN 9781441982360.
- Buttazzo, Giorgio. Hard real-time computing systems: predictable scheduling algorithms and applications [en línea]. 3rd ed. New York: Springer, 2011 [Consulta: 27/05/2022]. Disponible a: <https://ebookcentral-proquest-com.recursos.biblioteca.upc.edu/lib/upcatalunya-ebooks/detail.action?docID=3067230>. ISBN 9781461406754.
- Galli, Ricardo. Principios y algoritmos de concurrencia. Palma de Mallorca: CreateSpace Independent Publishing Platform, 2015. ISBN 9781517029753.