

| MÓDULO | MATERIA | CURSO | SEMESTRE | CRÉDITOS | TIPO |
|--|--|-------|--|----------|-------------|
| Formación específica de Rama | Sistemas Operativos, Sistemas Distribuidos y Redes | 2º | 3º | 6 | Obligatoria |
| PROFESOR(ES) | | | DIRECCIÓN COMPLETA DE CONTACTO PARA TUTORÍAS | | |
| <p>Grupos de Teoría</p> <p>Grupo A: Carlos Ureña Almagro Grupo B: Manuel Noguera García Grupo C: Carlos Ureña Almagro Grupo D: Pedro Villar Castro</p> <p>Grupos de Prácticas</p> <p>A1: Carlos Ureña Almagro A2: Carlos Ureña Almagro A3: Ana Mª Sánchez López</p> <p>B1: Pedro Villar Castro B2: Manuel Noguera García B3: Ana Mª Sánchez López</p> <p>C1: Pedro Villar Castro C2: Gabriel J. Guerrero Contreras C3: Rafael A. Prieto De Lope</p> <p>D1: Pedro Villar Castro D2: Gabriel J. Guerrero Contreras</p> | | | <p>E.T.S. de Ingeniería en Informática y Telecomunicaciones. Planta 3ª C./ Periodista Daniel Saucedo Aranda s/n. 18071 – Granada. Universidad de Granada</p> <p>Gabriel J. Guerrero Contreras: giguerrero@ugr.es Manuel Noguera García : Desp. 22 mnoquera@ugr.es. Tfno: 958242810 Rafael A. Prieto De Lope: rapride@correo.ugr.es Ana Mª Sánchez López: Desp. 26 amlopez@ugr.es. Tfno: 958240579 Carlos Ureña Almagro: Desp. 34 curena@ugr.es. Tfno: 958240577 Pedro Villar Castro: Desp. 18 pwillarc@ugr.es. Tfno: 958240637</p> | | |
| | | | HORARIO DE TUTORÍAS | | |
| | | | Los horarios de tutorías del profesorado pueden consultarse en la web de grado: http://grados.ugr.es/informatica/pages/infoacademica/profesorado/*/ 24 | | |
| GRADO EN EL QUE SE IMPARTE | | | | | |
| Grado en Ingeniería Informática | | | | | |
| PRERREQUISITOS Y/O RECOMENDACIONES | | | | | |
| Los alumnos no tendrán que tener asignaturas, materias o módulos aprobados como requisito indispensable para cursar el módulo. No obstante se recomienda la superación de los contenidos y adquisición de competencias de las materias de formación básica, teniendo especial importancia la superación de las materias de "Fundamentos de Programación", "Fundamentos de software" y "Metodología de la Programación". | | | | | |



BREVE DESCRIPCIÓN DE CONTENIDOS (SEGÚN MEMORIA DE VERIFICACIÓN DEL GRADO)

Exclusión mutua, sincronización y comunicación entre procesos. Propiedades de seguridad y vivacidad. Algoritmos para modelos basados en memoria compartida y paso de mensajes. Semáforos y monitores. Bibliotecas de programación concurrente y distribuida. Técnicas para el diseño de aplicaciones de tiempo-real.

COMPETENCIAS GENERALES Y ESPECÍFICAS

Competencias Generales del Título

E6. Capacidad para concebir y desarrollar sistemas o arquitecturas informáticas centralizadas o distribuidas integrando hardware, software y redes.

Competencias Específicas del módulo

R6. Conocimiento y aplicación de los procedimientos algorítmicos básicos de las tecnologías informáticas para diseñar soluciones a problemas, analizando la idoneidad y complejidad de los algoritmos propuestos.

R8. Capacidad para analizar, diseñar, construir y mantener aplicaciones de forma robusta, segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados.

R11. Conocimiento y aplicación de las características, funcionalidades y estructura de los Sistemas Distribuidos, las Redes de Computadores e Internet y diseñar e implementar aplicaciones basadas en ellas.

R14. Conocimiento y aplicación de los principios fundamentales y técnicas básicas de la programación paralela, concurrente, distribuida y de tiempo real.

Competencias Básicas

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.

Competencias Transversales

T3. Capacidad de comunicación oral y escrita en el ámbito académico y profesional con especial énfasis, en la redacción de documentación técnica



OBJETIVOS (EXPRESADOS COMO RESULTADOS ESPERABLES DE LA ENSEÑANZA)

- Comprender la importancia de la programación concurrente en las aplicaciones de hoy en día.
- Identificar las principales características de los distintos tipos de sistemas concurrentes que existen.
- Conocer y entender los problemas que plantea el desarrollo de programas concurrentes y que no aparecen en la programación secuencial.
- Entender los conceptos de sincronización y exclusión mutua entre procesos.
- Identificar las propiedades de seguridad y vivacidad que un sistema concurrente debe cumplir y ser capaz de razonar si dichas propiedades se cumplen.
- Conocer los principales modelos de programación concurrente, paralela y distribuida.
- Adquirir experiencia y conocimiento en los mecanismos de sincronización y comunicación que se utilizan en la actualidad para desarrollar programas concurrentes tanto para sistemas de memoria compartida como para sistemas distribuidos.
- Entender el funcionamiento de semáforos y monitores como mecanismos de sincronización para memoria compartida y comprender cómo se pueden resolver problemas de programación concurrente usando monitores.
- Ser capaz de desarrollar algoritmos para sistemas basados en memoria compartida y para sistemas distribuidos que resuelvan problemas modelo en programación concurrente.
- Conocer y ser capaz de usar bibliotecas y plataformas estandarizadas para la implementación de programas concurrentes basados en memoria compartida y para sistemas distribuidos.
- Conocer las técnicas más destacadas para el diseño de sistemas de tiempo real.



TEMARIO DETALLADO DE LA ASIGNATURA

Tema 1. Introducción a la Programación Concurrente

- 1.1. Conceptos básicos y motivación.
- 1.2. Modelo abstracto de la Programación Concurrente. Consideraciones sobre el hardware.
- 1.3. Exclusión mutua y sincronización.
- 1.4. Propiedades de sistemas concurrentes.
- 1.5. Verificación de programas concurrentes.
- 1.6. Resolución de ejercicios.

Tema 2. Algoritmos y mecanismos de sincronización basados en memoria compartida

- 2.1. Introducción a la sincronización en sistemas con memoria compartida
- 2.2. Algoritmos básicos de exclusión mutua en sistemas con memoria compartida.
- 2.3. Soluciones hardware para la exclusión mutua. Cerrojos.
- 2.4. Semáforos. Estructura y operaciones.
- 2.5. Monitores como mecanismo de alto nivel. Definición y características. Semántica de las señales de los monitores. Implementación de monitores.
- 2.6. Resolución de ejercicios.

Tema 3. Sistemas basados en paso de mensajes

- 3.1. Mecanismos básicos en sistemas basados en paso de mensajes.
- 3.2. Modelos y lenguajes de programación distribuida.
- 3.3. Bibliotecas de paso de mensajes y patrones de interacción.
- 3.4. Mecanismos de alto nivel en sistemas distribuidos. RPC y RMI.
- 3.5. Resolución de ejercicios.

Tema 4. Introducción a los sistemas de tiempo real

- 4.1. Concepto de sistema de tiempo real. Medidas de tiempo y modelo de tareas.
- 4.2. Planificación de tareas periódicas con asignación de prioridades.
- 4.3. Modelos generales y específicos de tareas.
- 4.4. Resolución de ejercicios.

TEMARIO DE PRÁCTICAS

Práctica 1: Resolución de problemas de sincronización con semáforos.

Práctica 2: Programación de monitores con hebras.

Práctica 3: Programación de aplicaciones distribuidas.

Práctica 4: Programación de tareas periódicas con prioridades.

SEMINARIOS

Seminario práctico 1: Introducción a la programación mutihebra usando semáforos.

Seminario práctico 2: Introducción a la programación multihebra con monitores.

Seminario práctico 3: Introducción al uso de una interfaz de paso de mensajes.



BIBLIOGRAFÍA

BIBLIOGRAFÍA FUNDAMENTAL:

- J. T. Palma, C. Garrido, F. Sánchez, A. Quesada. Programación Concurrente. Thomson-Paraninfo. 2003.
- G. R. Andrews. Foundations of Multithreaded, Parallel, and Distributed Programming. Addison Wesley, 2000.
- F. Almeida, D. Gimenez, J. M. Mantas, A.M. Vidal . Introduccion a la Programacion Paralela. Paraninfo Cengage Learning, 2008.
- V. Kumar , A. Grama, A. Gupta, G. Karypis. *Introduction to Parallel Computing*. Benjamin/Cummings Publishing Company, 2003.
- George Coulouris, Jean Dollimore, Tim Kindberg, Gordon Blair. Distributed Systems: Concepts and Design (5ª Edición). Addison-Wesley, 2011.
- A. Burns, A. Wellings. *Sistemas de Tiempo Real y Lenguajes de Programación (3ª Edición)*. Addison Wesley, 2003.

BIBLIOGRAFÍA COMPLEMENTARIA:

- C. Breshears. The Art of Concurrency: A Thread Monkey's Guide to Writing Parallel Applications. O'Reilly Media. 2009.
- N.A. Lynch. *Distributed Algorithms*. Morgan Kaufmann. 1996.

ENLACES RECOMENDADOS

A principio de curso se avisará de la plataforma web y páginas web auxiliares donde se encontrarán los enlaces recomendados para la asignatura.

METODOLOGÍA DOCENTE

1 **Lección magistral (Clases teóricas-expositivas) (grupo grande)**

Descripción: Presentación en el aula de los conceptos propios de la materia haciendo uso de metodología positiva con lecciones magistrales participativas y medios audiovisuales. Evaluación y examen de las capacidades adquiridas.



Propósito: Transmitir los contenidos de la materia motivando al alumnado a la reflexión, facilitándole el descubrimiento de las relaciones entre diversos conceptos y formarle una mentalidad crítica
Contenido en ECTS: 30 horas presenciales (1.2 ECTS)
Competencias: E6, CB2, T3, R6, R8, R11, R14

2. Actividades prácticas (Clases prácticas de laboratorio) (grupo pequeño)

Descripción: Actividades a través de las cuales se pretende mostrar al alumnado cómo debe actuar a partir de la aplicación de los conocimientos adquiridos.

Propósito: Desarrollo en el alumnado de las habilidades instrumentales de la materia.

Contenido en ECTS: 15 horas presenciales (0.6 ECTS)

Competencias: E6, CB2, T3, R6, R8, R11, R14

3. Seminarios (grupo pequeño)

Descripción: Modalidad organizativa de los procesos de enseñanza y aprendizaje donde tratar en profundidad una temática relacionada con la materia. Incorpora actividades basadas en la indagación, debate, la reflexión y el intercambio.

Propósito: Desarrollo en el alumnado de las competencias cognitivas y procedimentales de la materia.

Contenido en ECTS: 10 horas presenciales (0.4 ECTS)

Competencias: E6, CB2, T3, R6, R8, R11, R14

4. Actividades no presenciales individuales (Estudio y trabajo autónomo)

Descripción: 1) Actividades (guiadas y no guiadas) propuestas por el profesor a través de las cuales y de forma individual se profundiza en aspectos concretos de la materia posibilitando al estudiante avanzar en la adquisición de determinados conocimientos y procedimientos de la materia, 2) Estudio individualizado de los contenidos de la materia 3) Actividades evaluativas (informes, exámenes, ...).

Propósito: Favorecer en el estudiante la capacidad para autorregular su aprendizaje, planificándolo, diseñándolo, evaluándolo y adecuándolo a sus especiales condiciones e intereses.

Contenido en ECTS: 45 horas no presenciales (1.8 ECTS)

Competencias: E6, CB2, T3, R6, R8, R11, R14

5. Actividades no presenciales grupales (Estudio y trabajo en grupo)

Descripción: Actividades (guiadas y no guiadas) propuestas por el profesor a través de las cuales y de forma grupal se profundiza en aspectos concretos de la materia posibilitando a los estudiantes avanzar en la adquisición de determinados conocimientos y procedimientos de la materia.

Propósito: Favorecer en los estudiantes la generación e intercambio de ideas, la identificación y análisis de diferentes puntos de vista sobre una temática, la generalización o transferencia de conocimiento y la valoración crítica del mismo.

Contenido en ECTS: 45 horas no presenciales (1.8 ECTS)

Competencias: E6, CB2, T3, R6, R8, R11, R14

6. Tutorías académicas (grupo pequeño)

Descripción: manera de organizar los procesos de enseñanza y aprendizaje que se basa en la interacción directa entre el estudiante y el profesor.

Propósito: 1) Orientar el trabajo autónomo y grupal del alumnado, 2) profundizar en distintos aspectos de la materia y 3) orientar la formación académica integral del estudiante.

Contenido en ECTS: 5 horas presenciales, grupales e individuales (0.2 ECTS)

Competencias: E6, CB2, T3, R6, R8, R11, R14

EVALUACIÓN (INSTRUMENTOS DE EVALUACIÓN, CRITERIOS DE EVALUACIÓN Y PORCENTAJE SOBRE LA CALIFICACIÓN FINAL, ETC.)

Todo lo relativo a la evaluación se regirá por la Normativa de evaluación y calificación de los estudiantes vigente en la Universidad de Granada, que puede consultarse en:
[http://secretariageneral.ugr.es/bougr/pages/bougr71/ncg712/!](http://secretariageneral.ugr.es/bougr/pages/bougr71/ncg712/)



Preferentemente, la evaluación se ajustará al sistema de evaluación continua del aprendizaje del estudiante siguiendo el artículo 7 de la anterior Normativa.

Se utilizarán alguna o algunas de las siguientes técnicas de evaluación:

- Para la parte teórica se realizarán tres sesiones de evaluación y entregas de ejercicios sobre el desarrollo y los resultados de las actividades propuestas.
- Para la parte práctica se realizarán prácticas de laboratorio, resolución de problemas y desarrollo de proyectos (individuales o en grupo), y se valorarán las entregas de los informes/memorias realizados por los alumnos, o en su caso las entrevistas personales con los alumnos y las sesiones de evaluación.
- En el caso de la evaluación continua, los seminarios se evaluarán teniendo en cuenta la asistencia, los problemas propuestos que hayan sido resueltos y entregados por los alumnos, en su caso, las entrevistas efectuadas durante el curso y la presentación oral de los trabajos desarrollados.

Para los estudiantes que se acojan a la evaluación única final, esta modalidad de evaluación estará formada por todas aquellas pruebas que el profesor estime oportunas, de forma que se pueda acreditar que el estudiante ha adquirido la totalidad de las competencias generales y específicas descritas en el apartado correspondiente de esta Guía Docente.

La calificación global corresponderá por tanto a la puntuación ponderada de los diferentes aspectos y actividades que integran el sistema de evaluación. Por tanto, el resultado de la evaluación será una calificación numérica obtenida mediante la suma ponderada de las calificaciones correspondientes a una parte teórica, una parte práctica y, en el caso de la evaluación continua, una parte relacionada con el trabajo autónomo de los alumnos, los seminarios impartidos y el aprendizaje basado en proyectos. La adaptación del sistema de evaluación general propuesto a las características de esta asignatura, con indicación explícita del peso de la evaluación de cada actividad formativa, se ajustará a lo indicado en la siguiente tabla:

| Actividades Formativas | Ponderación |
|-------------------------|--------------|
| Parte Teórica | 60% - 65% |
| Parte Práctica | 30% - 35% |
| Otros (seminarios, ...) | Hasta un 10% |

Para aprobar la asignatura es necesario tener una calificación numérica superior o igual a 5 (sobre 10). No obstante, además del requisito anterior, se establece como requisito adicional para superar la asignatura que tanto la calificación correspondiente a la parte teórica como la correspondiente a la parte práctica sean mayores o iguales a 4 (sobre 10).

El sistema de calificaciones se expresará mediante calificación numérica de acuerdo con lo establecido en el art. 5 del R. D 1125/2003, de 5 de septiembre, por el que se establece el sistema europeo de créditos y el sistema de



calificaciones en las titulaciones universitarias de carácter oficial y validez en el territorio nacional.

RÉGIMEN DE ASISTENCIA

- La asistencia a las clases teóricas no será obligatoria, aunque la participación activa en clase y la entrega de ejercicios planteados por el profesor se tendrá en cuenta dentro del sistema de evaluación continua de la asignatura.
- La asistencia a las clases prácticas no será obligatoria, exceptuando las sesiones en las que se programen pruebas de evaluación. En cualquier caso, la asistencia y participación activa en clase se tendrá en cuenta dentro del sistema de evaluación continua de la asignatura.

INFORMACIÓN ADICIONAL

Definición de grupo grande y grupo pequeño:

Los grupos grandes son grupos de 45 a 60 estudiantes.

Los grupos pequeños son grupos de 15 a 20 estudiantes.



ugr | Universidad
de Granada

INFORMACIÓN SOBRE TITULACIONES DE LA UGR
<http://grados.ugr.es>