



DATOS IDENTIFICATIVOS

Concurrencia y distribución

Asignatura	Concurrencia y distribución			
Código	O06G150V01602			
Titulación	Grado en Ingeniería Informática			
Descriptores	Creditos ECTS	Seleccione	Curso	Cuatrimestre
	6	OB	3	2c
Lengua	Castellano			
Impartición	Gallego			
Departamento	Informática			
Coordinador/a	Formella , Arno			
Profesorado	Formella , Arno Garcia Rosello, Emilio			
Correo-e	formella@ei.uvigo.es			
Web	http://www.ei.uvigo.es/%7Eformella/doc/cdg12			

Descripción general El contenido forma la base necesaria para comprender el funcionamiento de aplicaciones concurrentes y/o distribuidas, la evaluación de algoritmos concurrentes, la descripción de datos y de información en entornos distribuidos, el funcionamiento de procesadores modernos, y las capacidades de la programación con procesos/hilos incluso en forma distribuida.

Las clases se darán principalmente en castellano. El/la estudiante puede elegir si trabaja en gallego, castellano, alemán, y/o en inglés. Cierta información adicional (como por ejemplo manuales e información complementaria) se dará en inglés.

Las partes del presente guía en gallego y inglés no están revisadas y por eso sirven solamente de carácter informativo.

Competencias de titulación

Código	
A4	Conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería
A5	Conocimiento de la estructura, organización, funcionamiento e interconexión de los sistemas informáticos, los fundamentos de su programación, y su aplicación para la resolución de problemas propios de la ingeniería
A7	Capacidad para diseñar, desarrollar, seleccionar y evaluar aplicaciones y sistemas informáticos, asegurando su fiabilidad, seguridad y calidad, conforme a principios éticos y a la legislación y normativa vigente
A8	Capacidad para planificar, concebir, desplegar y dirigir proyectos, servicios y sistemas informáticos en todos los ámbitos, liderando su puesta en marcha y su mejora continua y valorando su impacto económico y social
A12	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
A13	Conocimiento, diseño y utilización de forma eficiente de los tipos y estructuras de datos más adecuados a la resolución de un problema
A14	Capacidad para analizar, diseñar, construir y mantener aplicaciones de forma robusta, segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados
A15	Capacidad de conocer, comprender y evaluar la estructura y arquitectura de los computadores, así como los componentes básicos que los conforman
A16	Conocimiento de las características, funcionalidades y estructura de los Sistemas Operativos y diseñar e implementar aplicaciones basadas en sus servicios
A19	Conocimiento y aplicación de las herramientas necesarias para el almacenamiento, procesamiento y acceso a los Sistemas de información, incluidos los basados en web
A21	Conocimiento y aplicación de los principios fundamentales y técnicas básicas de los sistemas inteligentes y su aplicación práctica

A22	Conocimiento y aplicación de los principios, metodologías y ciclos de vida de la ingeniería de software
A25	Capacidad para desarrollar, mantener y evaluar servicios y sistemas software que satisfagan todos los requisitos del usuario y se comporten de forma fiable y eficiente, sean asequibles de desarrollar y mantener y cumplan normas de calidad, aplicando las teorías, principios, métodos y prácticas de la Ingeniería del Software
A26	Capacidad para valorar las necesidades del cliente y especificar los requisitos software para satisfacer estas necesidades, reconciliando objetivos en conflicto mediante la búsqueda de compromisos aceptables dentro de las limitaciones derivadas del coste, del tiempo, de la existencia de sistemas ya desarrollados y de las propias organizaciones
A27	Capacidad de dar solución a problemas de integración en función de las estrategias, estándares y tecnologías disponibles
A28	Capacidad de identificar y analizar problemas y diseñar, desarrollar, implementar, verificar y documentar soluciones software sobre la base de un conocimiento adecuado de las teorías, modelos y técnicas actuales
A30	Capacidad para diseñar soluciones apropiadas en uno o más dominios de aplicación utilizando métodos de la ingeniería del software que integren aspectos éticos, sociales, legales y económicos
A31	Capacidad para comprender el entorno de una organización y sus necesidades en el ámbito de las tecnologías de la información y las comunicaciones
A33	Capacidad para emplear metodologías centradas en el usuario y la organización para el desarrollo, evaluación y gestión de aplicaciones y sistemas basados en tecnologías de la información que aseguren la accesibilidad, ergonomía y usabilidad de los sistemas
A35	Capacidad para seleccionar, desplegar, integrar y gestionar sistemas de información que satisfagan las necesidades de la organización, con los criterios de coste y calidad identificados
A36	Capacidad de concebir sistemas, aplicaciones y servicios basados en tecnologías de red, incluyendo Internet, web, comercio electrónico, multimedia, servicios interactivos y computación móvil
B1	Capacidad de análisis, síntesis y evaluación
B2	Capacidad de organización y planificación
B3	Comunicación oral y escrita en la lengua nativa
B5	Capacidad de abstracción: capacidad de crear y utilizar modelos que reflejen situaciones reales
B6	Capacidad de diseñar y realizar experimentos sencillos y analizar e interpretar sus resultados
B7	Capacidad de buscar, relacionar y estructurar información proveniente de diversas fuentes y de integrar ideas y conocimientos
B8	Resolución de problemas
B9	Capacidad de tomar decisiones
B10	Capacidad para argumentar y justificar lógicamente las decisiones tomadas y las opiniones
B11	Capacidad de actuar autónomamente
B12	Capacidad de trabajar en situaciones de falta de información y/o bajo presión
B15	Capacidad de relación interpersonal
B16	Razonamiento crítico
B18	Aprendizaje autónomo
B20	Creatividad
B22	Tener iniciativa y ser resolutivo
B24	Tener motivación por la calidad y la mejora continua

Competencias de materia

Resultados previstos en la materia	Resultados de Formación y Aprendizaje	
Analizar y describir los aspectos software/hardware y las propiedades de sistemas concurrentes y distribuidos.	A4	B1
	A5	B2
	A7	B3
	A12	B5
	A14	B7
	A15	B9
	A16	B10
	A22	B11
	A25	B12
	A26	B15
	A27	B16
	A28	B18
	A31	B22
	A33	B24
	A35	
A36		

Diseñar aplicaciones concurrentes y distribuidos.

A4 B1
 A5 B2
 A7 B3
 A12 B5
 A13 B6
 A14 B7
 A15 B8
 A16 B9
 A19 B10
 A22 B11
 A25 B12
 A26 B15
 A27 B16
 A28 B18
 A33 B20
 A35 B22
 A36 B24

Tomar conciencia de la importancia y del despliegue de sistemas concurrentes y distribuidos como tecnología moderna.

A4 B1
 A5 B3
 A7 B5
 A8 B7
 A12 B9
 A14 B10
 A15 B11
 A16 B12
 A19 B15
 A21 B16
 A22 B18
 A25 B20
 A26 B22
 A27 B24
 A30
 A31
 A33
 A35
 A36

Contenidos

Tema

Sistemas concurrentes y distribuidos	Concepto de la programación concurrente y distribuida Introducción al modelado de sistemas concurrentes y distribuidos Arquitecturas hardware para la concurrencia y distribución Herramientas para del desarrollo de aplicaciones concurrentes y distribuidos
Procesos	Concepto de procesos Planificador Atomicidad y exclusión mutua Concurrencia transaccional Reloj y estado distribuido
Sincronización y comunicación	Sincronización y comunicación en sistemas concurrentes y distribuidos Sincronización y comunicación a nivel bajo Sincronización y comunicación a nivel alto Seguridad y vivacidad en sistemas concurrentes y distribuidos
Herramientas de programación y desarrollo de aplicaciones	Programación concurrente y distribuida con JAVA Programación concurrente y distribuida con C/C++ Patrones de diseño para el desarrollo de aplicaciones concurrentes y distribuidos Herramientas y metodologías de diseño, verificación y depuración de aplicaciones concurrentes y distribuidos

Planificación

	Horas en clase	Horas fuera de clase	Horas totales
Actividades introductorias	1.5	0	1.5
Sesión magistral	18	9	27
Estudios/actividades previos	0	16	16
Prácticas en aulas de informática	26.5	26.5	53

Resolución de problemas y/o ejercicios	0	19.5	19.5
Presentaciones/exposiciones	0	1.75	1.75
Tutoría en grupo	1.25	1.25	2.5
Pruebas de respuesta corta	1.5	0	1.5
Pruebas de respuesta larga, de desarrollo	2	0	2
Informes/memorias de prácticas	0	12	12
Pruebas prácticas, de ejecución de tareas reales y/o simuladas.	1	0	1
Resolución de problemas y/o ejercicios	0	12	12
Otras	0.25	0	0.25

*Los datos que aparecen en la tabla de planificación son de carácter orientativo, considerando la heterogeneidad de alumnado

Metodologías	
	Descripción
Actividades introductorias	Presentación de la asignatura y aclaración de todos los aspectos relacionados con la guía docente. Planificación temporal de las actividades presenciales. Introducción de las herramientas de control y evaluación. Recomendaciones específicas para lograr los objetivos de la asignatura.
Sesión magistral	Exposición de los contenidos teóricos de la asignatura. Presentación de ejemplos y casos específicos. Controles de estudios y lecturas previos. Controles del avance de la adquisición de conocimiento por parte del estudiante. Interacción con/entre los estudiantes mediante actividades específicas.
Estudios/actividades previas	Lectura de documentos relacionados con el contenido de la asignatura. Análisis y diseño de las tareas de los ejercicios en el laboratorio.
Prácticas en aulas de informática	Implementación y depuración de los ejercicios planteados como tareas de programación. Realización de pruebas de funcionamiento y/o rendimiento de aplicaciones concurrentes y distribuidos con un análisis crítico de las observaciones.
Resolución de problemas y/o ejercicios	Desarrollo de propuestas de resolución de problemas. Comprobaciones de corrección y análisis de rendimiento. Implementación de soluciones alternativas. Análisis crítico de soluciones propuestas.
Presentaciones/exposiciones	Exposiciones breves de los hitos alcanzados en las tareas de programación y ejercicios relacionados.
Tutoría en grupo	Control del avance del aprendizaje. Recomendaciones para lograr los objetivos de la asignatura a nivel individual. Apoyo y ayuda en la resolución de las tareas propuestas.

Atención personalizada	
Metodologías	Descripción
Actividades introductorias	Se llevará a cabo un control individualizado de la asistencia a las actividades presenciales. Se llevará a cabo un control individualizado de los trabajos/ejercicios/pruebas. Durante las actividades presenciales en los laboratorios el profesor interactúa de forma proactiva con los estudiantes. Existirán horas tutoriales para la atención individualizada relacionada con cualquier aspecto de docencia y aprendizaje.
Sesión magistral	Se llevará a cabo un control individualizado de la asistencia a las actividades presenciales. Se llevará a cabo un control individualizado de los trabajos/ejercicios/pruebas. Durante las actividades presenciales en los laboratorios el profesor interactúa de forma proactiva con los estudiantes. Existirán horas tutoriales para la atención individualizada relacionada con cualquier aspecto de docencia y aprendizaje.
Prácticas en aulas de informática	Se llevará a cabo un control individualizado de la asistencia a las actividades presenciales. Se llevará a cabo un control individualizado de los trabajos/ejercicios/pruebas. Durante las actividades presenciales en los laboratorios el profesor interactúa de forma proactiva con los estudiantes. Existirán horas tutoriales para la atención individualizada relacionada con cualquier aspecto de docencia y aprendizaje.
Tutoría en grupo	Se llevará a cabo un control individualizado de la asistencia a las actividades presenciales. Se llevará a cabo un control individualizado de los trabajos/ejercicios/pruebas. Durante las actividades presenciales en los laboratorios el profesor interactúa de forma proactiva con los estudiantes. Existirán horas tutoriales para la atención individualizada relacionada con cualquier aspecto de docencia y aprendizaje.
Pruebas	Descripción
Pruebas de respuesta corta	

Pruebas de respuesta larga, de desarrollo

Informes/memorias de prácticas

Pruebas prácticas, de ejecución de tareas reales y/o simuladas.

Resolución de problemas y/o ejercicios

Otras

Evaluación

	Descripción	Calificación
Pruebas de respuesta corta	(P1) Conjunto de preguntas cortas para el control de la realización de actividades, tareas, y estudios.	20
Pruebas de respuesta larga, de desarrollo	(P2) Conjunto de preguntas largas que relacionan los diferentes apartados de los contenidos y miden el nivel de la adquisición de las competencias de la materia.	40
Informes/memorias de prácticas	(P3) Elaboración de informes (según guía) que recogen los principales desarrollos y resultados obtenidos por el/la estudiante. Partes de dichos informes se elaborará en pequeños grupos.	25
Pruebas prácticas, de ejecución de tareas reales y/o simuladas.	(P4) Demostración de los desarrollos e implementaciones de las tareas de programación y experimentos de estudio.	25
Resolución de problemas y/o ejercicios	(P5) Elaboración de algoritmos y sus análisis con cierto nivel de formalismo para comprobar la corrección y estudiar el rendimiento.	7.5
Otras	(P6) Breves presentaciones orales con medios audiovisuales de desarrollos y resultados obtenidos por el/la estudiante.	7.5

Otros comentarios sobre la Evaluación

Se entiende que un/una estudiante que no participa en por lo menos 80% de las actividades presenciales es un no asistente y por consecuencia se sumete a la forma de evaluación para no asistentes.

La nota final se obtiene de la siguiente manera asumiendo que cada parte (P1-P6) se evalúa con una escala de 0-10:

La asignatura es suspensa si P2

Fuentes de información

J.T. Palma Méndez, M.C. Garrido Carrera, F. Sánchez Figueroa, A. Quesada Arencibia., **Programacion Concurrente**,

D. Lea, **Programación concurrente en Java**,

G. Coulouris, J. Dollimore, T. Kindberg, **Sistemas distribuidos : conceptos y diseño**,

M.L. Liu, **Computación distribuida : fundamentos y aplicaciones**,

D. Schmidt, M. Stal, H. Rohnert, F. Buschman, **Pattern-oriented Software Architecture, Pattern for Concurrent and Networked Objects**,

Varios, **Internet**,

M. Herlihy, N. Shavit, **The Art of Multiprocessor Programming**,

Recomendaciones

Asignaturas que se recomienda haber cursado previamente

Informática: Algoritmos y estructuras de datos I/O06G150V01201

Informática: Arquitectura de computadoras I/O06G150V01203

Informática: Programación I/O06G150V01104

Programación II/O06G150V01205

Algoritmos y estructuras de datos II/O06G150V01302

Arquitectura de computadoras II/O06G150V01303

Arquitecturas paralelas/O06G150V01401

Ingeniería del software I/O06G150V01304

Ingeniería del software II/O06G150V01403

Redes de computadoras I/O06G150V01404

Sistemas operativos I/O06G150V01305

Otros comentarios

Se menciona que prácticamente todas las asignaturas optativas en uno u otro aspecto requieren del concepto de concurrencia y distribución en sistemas modernos para lograr sus objetivos específicos.
