

PLAN DOCENTE DE LA ASIGNATURA: **PROGRAMACIÓN CONCURRENTE Y DISTRIBUIDA**

CÓDIGO: **502366**

CURSO ACADÉMICO: **2025/2026**

PLAN DOCENTE DE LA ASIGNATURA

Curso académico 2025/2026

Identificación y características de la asignatura					
Código	502366				
Denominación (español)	Programación Concurrente y Distribuida				
Denominación (inglés)	Concurrent and Distributed Programming				
Titulaciones	Grado en Ingeniería Informática en Tecnología de la Información (GIITI). Doble Grado GITT/GIITI				
Centro	Centro Universitario de Mérida				
Semestre	5º/9º				
Módulo	Módulo Común a la Rama de Informática				
Materia	Programación				
Carácter	Optativa	ECTS	6	Semestre	5/9
Profesor/es					
Nombre		Despacho	Correo-e	Página web	
Pendiente de confirmar					
Área de conocimiento	Lenguaje y Sistemas Informáticos				
Departamento	Ingeniería de Sistemas Informáticos y Telemáticos				
Profesor coordinador (si hay más de uno)					
Competencias*					
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				
✓	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				
✓	CB4 - Que los estudiantes puedan transmitir información, ideas, problemas y soluciones a un público tanto especializado como no especializado				
✓	CB5 - Que los estudiantes hayan desarrollado aquellas habilidades de aprendizaje necesarias para emprender estudios posteriores con un alto grado de autonomía				

* Los apartados relativos a competencias, breve descripción del contenido, actividades formativas, metodologías docentes, resultados de aprendizaje y sistemas de evaluación deben ajustarse a lo recogido en la memoria verificada del título y en la normativa de evaluación (DOE 12 de diciembre de 2016)

Competencias generales	
✓	CG5 - Capacidad para concebir, desarrollar y mantener sistemas, servicios y aplicaciones informáticas empleando los métodos de la ingeniería del software como instrumento para el aseguramiento de su calidad, de acuerdo con los conocimientos adquiridos según lo establecido en el anexo-2 de la Resolución de 8 de junio de 2009 de la Secretaría General de Universidades (BOE de 4 de Agosto de 2009) en el ámbito de las Tecnologías de la Información.
✓	CG6 - Capacidad para concebir y desarrollar sistemas o arquitecturas informáticas centralizadas o distribuidas integrando hardware, software y redes de acuerdo con los conocimientos adquiridos según lo establecido en el anexo-2 de la Resolución de 8 de junio de 2009 de la Secretaría General de Universidades (BOE de 4 de Agosto de 2009) en el ámbito de las Tecnologías de la Información.
✓	CG8 - Conocimiento de las materias básicas y tecnologías, que capaciten para el aprendizaje y desarrollo de nuevos métodos y tecnologías, así como las que les doten de una gran versatilidad para adaptarse a nuevas situaciones.
✓	CG9 - Capacidad para resolver problemas con iniciativa, toma de decisiones, autonomía y creatividad. Capacidad para saber comunicar y transmitir los conocimientos, habilidades y destrezas de la profesión de Ingeniero Técnico en Informática.
✓	CG10 - Conocimientos para la realización de mediciones, cálculos, valoraciones, tasaciones, peritaciones, estudios, informes, planificación de tareas y otros trabajos análogos de informática, de acuerdo con los conocimientos adquiridos según lo establecido en el anexo-2 de la Resolución de 8 de junio de 2009 de la Secretaría General de Universidades (BOE de 4 de Agosto de 2009) en el ámbito de las Tecnologías de la Información.
Competencias específicas	
	CE3 - Capacidad para comprender y dominar los conceptos básicos de matemática discreta, lógica, algorítmica y complejidad computacional, y su aplicación para la resolución de problemas propios de la ingeniería.
	CE12 - 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.
	CE13 - Conocimiento, diseño y utilización de forma eficiente los tipos y estructuras de datos más adecuados a la resolución de un problema.
✓	CE14 - Capacidad para analizar, diseñar, construir y mantener aplicaciones de forma robusta, segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados.
✓	CE20 - Conocimiento y aplicación de los principios fundamentales y técnicas básicas de la programación paralela, concurrente, distribuida y de tiempo real.
	CE21 - Conocimiento y aplicación de los principios fundamentales y técnicas básicas de los sistemas inteligentes y su aplicación práctica.
Competencias transversales	
	CT4. Resolución de problemas
	CT5. Toma de decisiones
	CT6. Orientación al aprendizaje
	CT11. Comunicación en lengua extranjera
✓	CT12. Diversidad e interculturalidad
✓	CT13. Resistencia y adaptación al entorno
	CT15. Comunicación interpersonal
Contenidos	
Breve descripción del contenido*	
Programación paralela, concurrente y distribuida. Concurrencia y Sincronización.	

Temario de la asignatura	
TEMARIO DE LA PARTE TEÓRICA	
Denominación del tema 1: CONCEPTOS FUNDAMENTALES DE LA PROGRAMACIÓN CONCURRENTES Y DISTRIBUIDA	
Contenidos del tema 1:	
1.1 Introducción 1.2 Concepto de programación concurrente 1.3 Beneficios de la programación concurrente 1.4 Concurrencia y arquitecturas hardware 1.5 Especificación de ejecución concurrente 1.6 Características de los sistemas concurrentes 1.7 Problemas inherentes a la programación concurrente 1.8 Corrección de programas concurrentes 1.9 Concepto de programación distribuida	
Denominación del tema 2: PROCESOS E HILOS	
Contenidos del tema 2:	
1.1 Procesos 1.2 Procesos en PASCAL-FC 1.3 Hilos 1.4 Hilos en Java	
Denominación del tema 3: PRIMERAS APROXIMACIONES A LA SOLUCIÓN DE LOS PROBLEMAS DE LA PROGRAMACIÓN CONCURRENTES	
Contenidos del tema 3:	
1.1 Introducción 1.2 Tipos de sincronización 1.3 Solución 1.4 Espera ocupada para exclusión mutua 1.5 Algoritmos no eficientes 1.6 Algoritmos eficientes 1.7 Soluciones Hardware	
Denominación del tema 4: PRIMITIVAS DE SINCRONIZACIÓN BASADAS EN MEMORIA COMPARTIDA. SEMÁFOROS	
Contenidos del tema 4:	
1.1 Introducción 1.2 Definición de semáforos 1.3 Resolución de problemas 1.4 Problemas clásicos en Pascal-FC y Java 1.5 Problemas propuestos	
Denominación del tema 5: PRIMITIVAS DE SINCRONIZACIÓN BASADAS EN MEMORIA COMPARTIDA. MONITORES	
Contenidos del tema 5:	
1.1 Introducción 1.2 Definición de monitor 1.3 Condición de sincronización en monitores 1.4 Problemas clásicos 1.5 Implementación de monitores con semáforos 1.6 Semántica de la operación resume 1.7 Implementación de monitores en Java	
Denominación del tema 6: MECANISMOS DE PASO DE MENSAJE	
Contenidos del tema 6:	
1.1 Introducción 1.2 Identificación en el proceso de comunicación 1.3 Sincronización 1.4 Canal de comunicación y mensajes 1.5 Condiciones de error en los sistemas de paso de mensaje 1.6 Modelo de comunicación: Espera selectiva	
Denominación del tema 7: PASO DE MENSAJE ASÍNCRONO	

Contenidos del tema 7:

1.1 Introducción

1.2 Resolución de problemas empleando paso de mensaje asíncrono

1.3 Paso de mensaje asíncrono con Java

Denominación del tema 8: PASO DE MENSAJE SÍNCRONO CON CANALES

Contenidos del tema 8:

1.1 Introducción

1.2 Comunicación mediante canales

1.3 Resolución de problemas empleando paso de mensaje síncrono

1.4 Paso de mensaje síncrono con Java

Denominación del tema 9: INVOCACIÓN REMOTA Y LLAMADA A PROCEDIMIENTO REMOTO

Contenidos del tema 9:

1.1 Introducción

1.2 Invocación remota

1.3 Problemas clásicos

Denominación del tema 10: PLATAFORMAS DE OBJETOS DISTRIBUIDOS

Contenidos del tema 10:

1.1 Introducción

1.2 Sistemas distribuidos basados en Middleware

1.3 Modelos de infraestructuras para sistemas distribuidos

1.4 Objetivos distribuidos

1.5 Ventajas e inconvenientes

1.6 Componentes de aplicaciones distribuidas

1.7 Pasos para desarrollar una aplicación distribuida

TEMARIO DE LA PARTE PRÁCTICA

• Se propondrá, al iniciar la asignatura, un supuesto práctico, en el que los alumnos aplicarán los conocimientos que vayan adquiriendo de cada tema.

Para practicar la programación concurrente se trabajará en el supuesto práctico, con hilos, semáforos y monitores. Para practicar la programación distribuida, se trabajará el paso de mensaje asíncrono, síncrono con canales y RMI.

Actividades formativas*

Horas de trabajo del alumno por tema		Horas teóricas	Actividades prácticas				Actividad de seguimiento	No presencia I
Tema	Total	GG	CH	L	O	S	TP	EP
1	6	2			1			3
2	19	6			3			10
3	10	3			1		1	5
4	17	6			1			10
5	18	6			1		1	10
6	12	3			1			8
7	9	3			1			5
8	13	3			2			8
9	12	3			1			8
10	23	6			4		1	12
Evaluación **	11	2			1			8
TOTAL	150	43			17		3	87

GG: Grupo Grande (85 estudiantes).

PCH: prácticas clínicas hospitalarias (7 estudiantes)

LAB: prácticas laboratorio o campo (15 estudiantes)

ORD: prácticas sala ordenador o laboratorio de idiomas (20 estudiantes)

SEM: clases problemas o seminarios o casos prácticos (40 estudiantes).

TP: Tutorías Programadas (seguimiento docente, tipo tutorías ECTS).
EP: Estudio personal, trabajos individuales o en grupo, y lectura de bibliografía.

Metodologías docentes*

1. Clases expositivas de teoría y problemas: Presentación de los contenidos de la asignatura y planificación de la participación de todos los estudiantes en las distintas tareas. Discusión de aspectos teóricos. Adicionalmente se realizarán charlas divulgativas realizadas por expertos y/o empresas de la materia.
2. Enseñanza participativa: Trabajos prácticos en grupos medianos o pequeños.
3. Tutorización: Actividad de seguimiento para tutela de trabajos dirigidos, consultas de dudas y asesoría en grupos pequeños o individuales.
4. Aprendizaje autónomo mediante el análisis de documentos escritos, la elaboración de memorias, el estudio de la materia impartida y desarrollo de los supuestos prácticos planteados.
5. Aprendizaje virtual. Uso de herramientas virtuales de comunicación entre profesor y estudiante e incluso entre los estudiantes entre sí.

A continuación, se detallan algunas de las actividades formativas que se plantearán a lo largo del curso para alcanzar los objetivos de aprendizaje de la asignatura. Aunque cada actividad sólo se detalla dentro de una modalidad (presenciales en grupo grande, presenciales en laboratorio, tutorías ECTS y no presenciales), algunas de ellas se desarrollarán en varias modalidades, pudiendo éstas, ser realizadas de forma individual y/o en grupo.

Presenciales en grupo grande

Orientadas principalmente a la adquisición de los conceptos teóricos de la asignatura, en estas actividades se combinan estudio previo por parte del estudiante más aprendizaje activo/inductivo/grupal, más clases expositivas con la resolución de problemas individualmente y/o en grupo con metodologías activas de aprendizaje.

Guía y pautas para realizar presentaciones orales de forma efectiva, enfocándolas a la defensa del trabajo fin de grado.

Presenciales en laboratorio

Las sesiones de laboratorio estarán a disposición de los alumnos antes del inicio de cada sesión.

Cada sesión dispone de un guion que contiene los objetivos y los trabajos que se deben desarrollar. El estudiante puede realizar parte de alguna de las sesiones de forma remota, de manera que en el momento de la sesión presencial de laboratorio, sea capaz de implantar la solución software real.

Tutorías ECTS

Las actividades formativas que se plantean en este bloque están orientadas, principalmente, a realizar el seguimiento de la adquisición de las competencias transversales.

No Presenciales

Dentro de las actividades no presenciales planteadas se encuentran las siguientes:

- Estudio individual.
- Reuniones de grupo
- Búsqueda de información
- Plantear preguntas de test
- Seguimiento de problemas resueltos
- Acceso a documentación del aula virtual
- Comunicación con profesores y compañeros mediante foros
- Cuestionarios de evaluación y autoevaluación

- Comunicación con profesores y compañeros mediante foros
- Cuestionarios de evaluación y autoevaluación.

Resultados de aprendizaje*

El alumno será capaz de:

- Aprender a identificar problemas de naturaleza concurrente y distribuida.
- Conocer, dominar y aplicar correctamente las principales primitivas de sincronización.
- Desarrollar correctamente algoritmos distribuidos.

Con respecto a las competencias transversales:

- Aceptar y comprender las afiliaciones culturales y/o sociales como relaciones estructurales, volitivas y razonables de la condición humana. (CT12, 2do nivel de dominio)
- Actuar con eficacia alcanzando los objetivos que se ha marcado en situaciones de presión de tiempo, desacuerdo, oposición y adversidad. (CT13, 2do nivel de dominio)

Sistemas de evaluación*

Continua

Se valorarán los conocimientos teóricos-prácticos adquiridos por el estudiante en relación con las competencias y objetivos de la asignatura, así como su grado de participación en el proceso educativo. El sistema de evaluación contempla la evaluación continua y la realización de exámenes finales. La evaluación continua considerará los siguientes aspectos:

Teoría: (40% de la nota)

Examen teórico tipo test y/o preguntas cortas y/o ejercicio prácticos, en el que deberá obtener como mínimo un 4 sobre 10 para poder hacer la media con la nota de prácticas.

Exposición oral (20%) (No recuperable): Cada estudiante o grupo (2 alumnos) deberá exponer un caso práctico del contenido de la asignatura, se valorará originalidad e investigación sobre el mismo, se puntuará la exposición según rúbrica que proporcionará el docente y siguiendo la guía y pautas para realizar presentaciones orales (10%) y contenido 10%. La asistencia a la exposición de todos los trabajos es obligatoria.

La **no** exposición del tema propuesto o no asistir a todos los trabajos, implicará que el estudiante no tiene la calificación de este apartado.

Prácticas: (40% de la nota)

Cada estudiante deberá entregar obligatoriamente las prácticas propuestas en cada sesión. Podrá existir un examen de práctica final para demostrar la autoría de la misma. Si es considerado como apto se procederá a corregir la práctica final. Se debe tener aprobada la práctica final para aprobar la asignatura así como el examen de modificación.

Sistemas de evaluación	Porcentaje
Examen.	40%

		(Entre el 0 y el 70%)	
	Exposición oral de trabajos realizados.	20% (Entre el 0 y el 40%)	
	Realización de trabajos dirigidos (informes, casos prácticos, ejercicios y problemas).	40% (Entre el 0 y el 80%)	
Única prueba final de carácter global			
El estudiante comunicará al profesor por escrito el tipo de evaluación elegido en las tres primeras semanas de cada semestre y el profesor remitirá la correspondiente relación a la Comisión de Calidad. Cuando un estudiante no realice esta comunicación, se entenderá que opta por la evaluación continua. Una vez elegido el tipo de evaluación, el estudiante no podrá cambiar en la convocatoria ordinaria de ese semestre y se atenderá a la normativa de evaluación para la convocatoria extraordinaria. Para aquellos alumnos que no deseen acogerse al sistema de evaluación continua, deberán comunicarlo al profesor durante las tres primeras semanas del semestre, según normativa de la Uex. Dichos alumnos tendrán derecho a examinarse de los contenidos de la asignatura, tanto teóricos como prácticos, mediante una prueba específica destinada a tal fin, mediante la cual podrán obtener el 100% de la nota final. Teoría: (60% de la nota) Examen teórico tipo test y preguntas cortas, en el que deberá obtener como mínimo un 4 sobre 10 para poder hacer la media con la nota de prácticas. Prácticas: (40% de la nota) Cada estudiante deberá entregar obligatoriamente las prácticas propuestas en cada sesión y deberán defenderlas mediante examen práctico en la fecha indicada por el docente. Deberá obtener como mínimo un 4 sobre 10 para poder hacer la media con la nota de teoría.			
Bibliografía			
Bibliografía básica			
[Ben90] M. Ben-Ari. Principles of concurrent and Distributed Programming. Prentice-Hall [Bur93] A. Burns, G.L. Davies. Concurrent Programming. Addison-Wesley [Dav92a] G.L. Davies. Pascal-FC, versión 5. User Guide for Pc compatibles. Universidad de Bradford, UK [Dav92b] G.L. Davies. Pascal-FC, versión 5. Language Reference Manual. Universidad de Bradford., UK [Dei98] H.M. Deitel, P.J. Deitel. Cómo programar en Java. Prentice Hall [Har98] S. J. Hartley. Concurrent Programming. The Java Programming Language. Oxford University Press. [Pal03] J.T. Palma, M.C. Garrido, F. Sánchez, A. Quesada. Programación Concurrente. Editorial Thomson-Paraninfo (www.paraninfo.es)			
Bibliografía complementaria			
[And91] G.R. Andrews. Concurrent Programming. Principle and Practice. Addison-Wesley [Dea00] D. Lea. Programación Concurrente en Java. Principios y patrones de diseño. Addison Wesley			
Otros recursos y materiales docentes complementarios			

Recursos Virtuales

- Aula virtual de la asignatura

Otros Recursos

Medios materiales utilizados:

- Pizarra
- Cañón de video
- Ordenador
- Internet

Materiales y recursos utilizados:

Los materiales y recursos utilizados estarán disponibles en el espacio reservado para la asignatura en el Campus Virtual. Concretamente los alumnos dispondrán de:

- Presentaciones para cada tema del programa
- Guiones de las sesiones de laboratorio
- Foros de preguntas y respuestas
- Tablón de anuncios de novedades
- Conjunto de referencias web
- Tareas virtuales para la entrega de las prácticas propuestas