



UNIVERSIDAD
POLITÉCNICA
DE MADRID

PROCESO DE
COORDINACIÓN DE LAS
ENSEÑANZAS PR/CL/001



E.T.S. de Ingeniería de
Sistemas Informáticos

ANX-PR/CL/001-01

GUÍA DE APRENDIZAJE

ASIGNATURA

615000305 - Programacion Concurrente Y Avanzada

PLAN DE ESTUDIOS

61CI - Grado En Ingenieria De Computadores

CURSO ACADÉMICO Y SEMESTRE

2021/22 - Segundo semestre

Índice

Guía de Aprendizaje

1. Datos descriptivos.....	1
2. Profesorado.....	1
3. Conocimientos previos recomendados.....	2
4. Competencias y resultados de aprendizaje.....	2
5. Descripción de la asignatura y temario.....	3
6. Cronograma.....	7
7. Actividades y criterios de evaluación.....	10
8. Recursos didácticos.....	14
9. Otra información.....	15

1. Datos descriptivos

1.1. Datos de la asignatura

Nombre de la asignatura	615000305 - Programacion Concurrente y Avanzada
No de créditos	6 ECTS
Carácter	Obligatoria
Curso	Segundo curso
Semestre	Cuarto semestre
Período de impartición	Febrero-Junio
Idioma de impartición	Castellano
Titulación	61CI - Grado en Ingeniería de Computadores
Centro responsable de la titulación	61 - Escuela Tecnica Superior De Ingenieria De Sistemas Informaticos
Curso académico	2021-22

2. Profesorado

2.1. Profesorado implicado en la docencia

Nombre	Despacho	Correo electrónico	Horario de tutorías *
Jorge Enrique Perez Martinez (Coordinador/a)	4415	jorgeenrique.perez@upm.es	Sin horario.
Maria Isabel Muñoz Fernandez	4412	isabel.munoz@upm.es	Sin horario.
Sergio Arevalo Viñuales	4413	sergio.arevalo@upm.es	Sin horario.

Stanislav Vakaruk		stanislav.vakaruk@upm.es	Sin horario.
Victor Rodriguez Fernandez		victor.rfernandez@upm.es	Sin horario.
Jesus Mayor Marquez		jesus.mayor@upm.es	Sin horario.

* Las horas de tutoría son orientativas y pueden sufrir modificaciones. Se deberá confirmar los horarios de tutorías con el profesorado.

3. Conocimientos previos recomendados

3.1. Asignaturas previas que se recomienda haber cursado

- Fundamentos De Programacion
- Programacion Orientada A Objetos

3.2. Otros conocimientos previos recomendados para cursar la asignatura

El plan de estudios Grado en Ingeniería de Computadores no tiene definidos otros conocimientos previos para esta asignatura.

4. Competencias y resultados de aprendizaje

4.1. Competencias

CC14 - Conocimiento y aplicación de los principios fundamentales y técnicas básicas de la programación paralela, concurrente, distribuida y de tiempo real

CT7 - Aprendizaje autónomo: El estudiante debe responsabilizarse de su propio aprendizaje, lo que le lleva a utilizar procesos cognitivos de forma estratégica y flexible, en función del objetivo de aprendizaje

4.2. Resultados del aprendizaje

RA311 - Detecta situaciones de interbloqueo y aplica mecanismos para afrontar dichas situaciones

RA312 - Aplica las teorías y técnicas necesarias para desarrollar correctamente programas concurrentes

RA306 - Identifica los algoritmos elementales para resolver los problemas básicos de la programación concurrente y las limitaciones de dichas soluciones para resolver problemas de mayor envergadura

RA305 - Desarrolla aplicaciones que ejecutan hilos concurrentes usando distintas herramientas

RA309 - Utiliza los distintos modelos de paso de mensajes para resolver problemas de programación concurrente, eligiendo el más adecuado a cada situación

RA310 - Utiliza los monitores como herramienta de alto nivel para resolver problemas de programación concurrente

RA317 - Dado un objetivo de aprendizaje a corto plazo, identifica de forma autónoma los conocimientos y habilidades necesarios y establece un plan para lograrlo. El plan integra la selección de fuentes de información, métodos de búsqueda de la información oportuna y criterios para sintetizar la información recopilada

RA307 - Utiliza los semáforos como herramienta de bajo nivel para resolver problemas de programación concurrente

RA316 - Explica la utilidad de la programación concurrente en los contextos de la programación paralela y los sistemas de tiempo real

5. Descripción de la asignatura y temario

5.1. Descripción de la asignatura

Se puede definir un "programa concurrente" como un conjunto de programas secuenciales que pueden ser ejecutados en paralelo (M. Ben-Ari). Este conjunto de programas secuenciales (o procesos) puede ser ejecutados en varios procesadores (paralelismo) o en un solo procesador (seudoparalelismo). En cualquiera de los dos casos, en el momento en que exista una cooperación entre dos o más procesos o existan recursos compartidos aparece la necesidad de sincronizar adecuadamente dichos procesos concurrentes para conseguir los resultados esperados..

Esta asignatura está orientada a estudiar las herramientas necesarias para la sincronización y coordinación de

procesos concurrentes, así como los entornos de programación que nos ofrecen dichas herramientas. Además se estudian las teorías y las técnicas adecuadas para resolver problemas basados en la concurrencia.

5.2. Temario de la asignatura

1. Introducción a la programación concurrente
 - 1.1. Niveles del SW de un ordenador
 - 1.2. Lenguajes de programación
 - 1.3. Gestión de memoria en el sistema operativo
 - 1.4. Gestión de procesos en el sistema operativo
 - 1.5. Threads dentro de un proceso
 - 1.6. Aplicaciones concurrentes
2. Abstracción de la programación concurrente
 - 2.1. El papel de la abstracción
 - 2.2. Ejecución concurrente como entrelazado de sentencias atómicas
 - 2.3. Justificación de la abstracción
 - 2.4. Entrelazado arbitrario
 - 2.5. Sentencias atómicas
 - 2.6. Corrección
 - 2.7. Justicia
 - 2.8. Instrucciones de código máquina
 - 2.9. Variables volátiles y no atómicas
3. El problema de la sección crítica
 - 3.1. Introducción y definición del problema
 - 3.2. Primer intento
 - 3.3. Prueba de corrección con diagramas de estado
 - 3.4. Otros intentos
 - 3.5. Algoritmo de Dekker
4. Semáforos

- 4.1. Estados de un proceso
- 4.2. Definición del tipo semáforo
- 4.3. Problema de la sección crítica para dos procesos
- 4.4. Invariantes de los semáforos
- 4.5. Problema de la sección crítica para N procesos
- 4.6. Problemas de orden de ejecución
- 4.7. Problema del productor-consumidor
- 4.8. Definiciones de semáforos
- 4.9. Problema de la cena de los filósofos
- 5. Monitores
 - 5.1. Introducción
 - 5.2. Declaración y uso de monitores
 - 5.3. Variables de condición
 - 5.4. Problema del productor-consumidor
 - 5.5. Problema de la reanudación inmediata
 - 5.6. Problema del lector-escritor
 - 5.7. Problema de la cena de los filósofos
 - 5.8. Monitores en Scala
- 6. Canales y mensajes
 - 6.1. Modelos de comunicación
 - 6.2. Canales
 - 6.3. Interacción cliente-servidor
 - 6.4. Canales y actores
- 7. Interbloqueos
 - 7.1. Modelo del sistema
 - 7.2. Caracterización de los interbloqueos
 - 7.3. Métodos para el manejo de interbloqueos
 - 7.4. Prevención de interbloqueos
 - 7.5. Cómo evitar interbloqueos

- 7.6. Detección de interbloqueos
- 7.7. Recuperación de interbloqueos
- 8. Programación asíncrona con futuros
 - 8.1. Futuros
 - 8.2. Callbacks
 - 8.3. Composición de futuros
- 9. Programación avanzada
 - 9.1. Programación paralela
 - 9.2. Programación distribuida
 - 9.3. Memoria transaccional software

6. Cronograma

6.1. Cronograma de la asignatura *

Sem	Actividad presencial en aula	Actividad presencial en laboratorio	Tele-enseñanza	Actividades de evaluación
1	Presentación. Tema 1. Introducción a la programación concurrente (RA 312, CT7) Duración: 02:00 LM: Actividad del tipo Lección Magistral			
2	Tema 2. Abstracción de la PC (RA312, CT7) Duración: 02:00 LM: Actividad del tipo Lección Magistral	Lab 1: Kojo (RA312, CT7) Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		
3	Tema 3. El problema de la región crítica (I) (RA305, RA306, CT7) Duración: 02:00 LM: Actividad del tipo Lección Magistral	Lab 2: Scala (I) (RA312) Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		
4	Tema 3. El problema de la región crítica (II) (RA305, RA306, CT7) Duración: 02:00 LM: Actividad del tipo Lección Magistral	Lab 3: Scala (II) (RA312, CT7) Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		
5	Tema 4. Semáforos (I) (RA305, RA307, CT7) Duración: 02:00 LM: Actividad del tipo Lección Magistral	Lab 4: Threads (RA305) Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		
6	Tema 4. Semáforos (II) (RA305, RA307, CT7) Duración: 02:00 LM: Actividad del tipo Lección Magistral	Lab 5: Semáforos (I) (RA305, RA307) Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		
7	Ejercicios (RA312, RA305, RA306, RA307, CT7, CC14) Duración: 02:00 LM: Actividad del tipo Lección Magistral	Lab 6: Semáforos II (RA305, RA307, CT7) Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		
8				Prueba de evaluación escrita de temas 1-4 (RA312, RA305, RA306, RA307, CC14) EX: Técnica del tipo Examen Escrito Evaluación continua Presencial Duración: 01:30 Prueba de evaluación escrita de prácticas Lab 1-6 (RA312, RA305, RA306, RA307, CC14) EX: Técnica del tipo Examen Escrito Evaluación continua Presencial Duración: 01:30

9	Tema 5. Monitores (I) (RA305, RA310, CT7) Duración: 02:00 LM: Actividad del tipo Lección Magistral			
10	Tema 5. Monitores (II) (RA305, RA310, CT7) Duración: 02:00 LM: Actividad del tipo Lección Magistral	Lab 7: Monitores en jBACI (RA305, RA310) Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		
11	Tema 6. Canales y paso de mensajes (I) (RA305, RA309) Duración: 02:00 LM: Actividad del tipo Lección Magistral	Lab 8: Monitores en Scala (RA305, RA310, CT7) Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		
12	Tema 6. Canales y paso de mensajes (II) (RA305, RA309, CT7) Duración: 02:00 LM: Actividad del tipo Lección Magistral	Lab 9: Actores (I) (RA305, RA309) Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		
13	Tema 8. Prog. Asíncrona con futuros (RA305, RA309, CT7) Duración: 02:00 LM: Actividad del tipo Lección Magistral	Lab 10: Actores (II) (RA305, RA309) Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		
14	Tema 9. Programación avanzada (RA316, CT7) Duración: 02:00 LM: Actividad del tipo Lección Magistral	Lab 11: Futuros (RA305, RA309) Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		
15	Tema 7. Interbloqueos (I) (RA311, CT7) Duración: 02:00 LM: Actividad del tipo Lección Magistral	Lab 12: Programación Paralela (RA316, CT7) Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		
16	Ejercicios (RA305, RA310, RA311, RA309, RA316, CC14, CT7) Duración: 02:00 LM: Actividad del tipo Lección Magistral			
17				<p>Prueba de evaluación escrita de temas 5-9 (Todos los RA, CC14) EX: Técnica del tipo Examen Escrito Evaluación continua Presencial Duración: 01:30</p> <p>Prueba de evaluación de prácticas Lab 7-12 (Todos los RA, CC14) EX: Técnica del tipo Examen Escrito Evaluación continua Presencial Duración: 01:30</p> <p>Examen final Teoría evaluación sólo prueba final (Todos los RA, CC14) EX: Técnica del tipo Examen Escrito Evaluación sólo prueba final Presencial Duración: 01:30</p> <p>Examen de prácticas evaluación sólo prueba final (Todos los RA, CC14) EX: Técnica del tipo Examen Escrito Evaluación sólo prueba final Presencial</p>

Duración: 01:30

Para el cálculo de los valores totales, se estima que por cada crédito ECTS el alumno dedicará dependiendo del plan de estudios, entre 26 y 27 horas de trabajo presencial y no presencial.

* El cronograma sigue una planificación teórica de la asignatura y puede sufrir modificaciones durante el curso derivadas de la situación creada por la COVID-19.

7. Actividades y criterios de evaluación

7.1. Actividades de evaluación de la asignatura

7.1.1. Evaluación continua

Sem.	Descripción	Modalidad	Tipo	Duración	Peso en la nota	Nota mínima	Competencias evaluadas
8	Prueba de evaluación escrita de temas 1-4 (RA312, RA305, RA306, RA307, CC14)	EX: Técnica del tipo Examen Escrito	Presencial	01:30	25%	0 / 10	CC14
8	Prueba de evaluación escrita de prácticas Lab 1-6 (RA312, RA305, RA306, RA307, CC14)	EX: Técnica del tipo Examen Escrito	Presencial	01:30	25%	0 / 10	CC14
17	Prueba de evaluación escrita de temas 5-9 (Todos los RA, CC14))	EX: Técnica del tipo Examen Escrito	Presencial	01:30	25%	0 / 10	CC14
17	Prueba de evaluación de prácticas Lab 7-12 (Todos los RA, CC14)	EX: Técnica del tipo Examen Escrito	Presencial	01:30	25%	0 / 10	CC14

7.1.2. Evaluación sólo prueba final

Sem	Descripción	Modalidad	Tipo	Duración	Peso en la nota	Nota mínima	Competencias evaluadas
17	Examen final Teoría evaluación sólo prueba final (Todos los RA, CC14)	EX: Técnica del tipo Examen Escrito	Presencial	01:30	50%	4 / 10	CC14
17	Examen de prácticas evaluación sólo prueba final (Todos los RA, CC14)	EX: Técnica del tipo Examen Escrito	Presencial	01:30	50%	4 / 10	CC14

7.1.3. Evaluación convocatoria extraordinaria

Descripción	Modalidad	Tipo	Duración	Peso en la nota	Nota mínima	Competencias evaluadas
Examen final Teoría evaluación sólo prueba final (Todos los RA)	EX: Técnica del tipo Examen Escrito	Presencial	01:30	50%	4 / 10	CC14
Examen de prácticas evaluación sólo prueba final (Todos los RA)	EX: Técnica del tipo Examen Escrito	Presencial	01:30	40%	4 / 10	CC14

7.2. Criterios de evaluación

Durante el curso, el alumno podrá optar a cursar la asignatura mediante evaluación continua o mediante solo prueba final. Si en ambos casos no supera la asignatura en la convocatoria de Junio, tendrá la opción de presentarse a la convocatoria extraordinaria de Julio. Para estas tres formas de evaluación se describen los algoritmos aplicados para el cálculo de la nota final. En los exámenes finales de las convocatorias de junio y julio el estudiante deberá demostrar las mismas capacidades que las exigidas en la evaluación continua, tanto en teoría y prácticas, así como la competencia transversal.

1. Nota final de junio (Por evaluación continua)

El estudiante que se acoja a la evaluación continua (**opción por omisión**) aprobará la asignatura durante el cuatrimestre si obtiene una nota final mayor o igual a 5 mediante las pruebas recogidas en dicha modalidad. Cuando la nota final sea inferior a 5, se guardarán para Julio aquellos bloques compensables. Si ambos están compensables, pero la nota final es inferior a 5, se deja al criterio del alumno examinarse del bloque o bloques que considere oportunos. La nota que se considerará para el cálculo de la nota final en un bloque siempre será la última obtenida en dicho bloque.

Aprobado := FALSE

Teoría := (primerParcialTeoría + segundoParcialTeoría) / 2

Prácticas := (primerParcialPrácticas + segundoParcialPrácticas) / 2

TeoriaCompensada := (Teoría >= 4)

PrácticasCompensadas := (Prácticas >= 4)

IF TeoríaCompensada **AND** PrácticasCompensadas **THEN begin**

.....NotaFinal := Teoria * 0,5 + Practicas * 0,5

.....Aprobado := (NotaFinal >= 5)

end

IF Aprobado **THEN** NotaFinal:= min (10, (NotaFinal + 0,5))

2. Nota Final de Junio (Por solo prueba final)

El estudiante que decida no seguir la evaluación continua deberá **solicitar dicha opción por escrito al coordinador** la asignatura **en las dos semanas siguientes al comienzo del cuatrimestre**. Se guardarán para la convocatoria de Julio los bloques compensables. Si ambos están compensables, pero la nota final es inferior a 5, se deja al criterio del alumno examinarse del bloque o bloques que considere oportunos. La nota que se considerará para el cálculo de la nota final en un bloque siempre será la última obtenida en dicho bloque.

Aprobado := FALSE

Teoría := (primerParcialTeoría + segundoParcialTeoría) / 2

Prácticas := (primerParcialPrácticas + segundoParcialPrácticas) / 2

TeoríaCompensada := (Teoría >= 4)

PrácticasCompensadas := (Practicas >= 4)

IF TeoríaCompensada **AND** PrácticasCompensadas **THEN begin**

.....NotaFinal := Teoria * 0,5 + Practicas * 0,5

.....Aprobado := (NotaFinal >= 5)

end

IF Aprobado **THEN** NotaFinal:= min (10, (NotaFinal + 0,5))

3. Nota final de julio:

El estudiante que opte por examinarse en la convocatoria extraordinaria de Julio no tendrá que examinarse del bloque de teoría o prácticas que tenga compensado. **NO se guarda para la convocatoria extraordinaria de Julio ninguna nota parcial de un bloque (teoría o prácticas) no compensando en Junio. En dicha situación, en la convocatoria de Julio habrá que examinarse de los contenidos del bloque completo.**

IF NOT TeoríaCompensada **THEN**

.....Teoría := ExamenTeoríaJulio

IF NOT PracticasCompensadas **THEN**

.....Prácticas := ExamenPrácticasJulio

IF (Teoría >= 4) **AND** (Prácticas >= 4) **THEN begin**

.....NotaFinal := 0,5 * Teoría + 0,5

.....Aprobado := (NotaFinal >= 5)

end

IF Aprobado **THEN** NotaFinal:= min (10, (NotaFinal + 0,5))

8. Recursos didácticos

8.1. Recursos didácticos de la asignatura

Nombre	Tipo	Observaciones
Principles of Concurrent and Distributed Programming (2nd. Ed.)	Bibliografía	Ben-Ari (2006); Addison-Wesley. Este libro se utilizará para estudiar la mayor parte de los aspectos básicos de la programación concurrente.
Programación Concurrente 2 Ed	Recursos web	Jorge E. Pérez Martínez, 1990. Ed. Rueda.
Programming in Scala (3rd. Ed.)	Bibliografía	Odersky, Spoon, Venners, 2016; Ed. Artima. Libro para conceptos de programación en Scala.
Learning concurrent programming in Scala, second edition	Bibliografía	Prokopec, 2017; Ed. PACKT. Libro complementario para conceptos de programación concurrente en Scala
The art of multiprocessor programming	Bibliografía	Herlihy, Shavitt, 2012; Ed. Morgan-Kaufmann
Concurrent programming	Bibliografía	Raynal, 2013; Ed. Springer
Programación concurrente, segunda edición	Bibliografía	Pérez-Martínez, 1990; Ed. Rueda
Plataforma moodle de la asignatura	Recursos web	Moodle del portal de la UPM Virtual
Laboratorio de la ETSISI, con PCs, proyector y pizarra	Equipamiento	Se utilizará para la realización de prácticas.

9. Otra información

9.1. Otra información sobre la asignatura

FORMACIÓN Y EVALUACIÓN DE LA COMPETENCIA TRANSVERSAL

Esta asignatura tiene asignada la competencia de aprendizaje autónomo. Dado que una parte importante del proceso de aprendizaje se realizará de forma autónoma (aprendizaje activo), la susodicha competencia transversal se entiende subsumida en el actual proceso de enseñanza-aprendizaje. Por tanto, aquellos estudiantes que superen la asignatura en cualquiera de las convocatorias verán incrementada su calificación final en 0,5 puntos al entender que el proceso de aprendizaje autónomo ha dado sus frutos y estos estudiantes han adquirido el nivel de desempeño suficiente en la indicada competencia transversal.

Con respecto a los objetivos de desarrollo sostenible de la ONU (<https://bit.ly//2qk9f28>) esta asignatura utiliza plataformas y lenguajes de acceso gratuito que promueven la consecución del objetivo 10. Reducción de la brecha digital.