ANX-PR/CL/001-01
GUÍA DE APRENDIZAJE

ASIGNATURA

615000242 - Programacion Concurrente Y Avanzada

PLAN DE ESTUDIOS

61IW - Grado En Ingenieria Del Software

CURSO ACADÉMICO Y SEMESTRE

2023/24 - Segundo semestre

Índice

Guía de Aprendizaje

1. Datos descriptivos....1
2. Profesorado....1
3. Conocimientos previos recomendados....4
4. Competencias y resultados de aprendizaje....4
5. Descripción de la asignatura y temario....5
6. Cronograma....9
7. Actividades y criterios de evaluación....12
8. Recursos didácticos....17
9. Otra información....18

1. Datos descriptivos

1.1. Datos de la asignatura

Nombre de la asignatura	615000242 - 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	61IW - Grado en Ingenieria del Software
Centro responsable de la titulación	61 - Escuela Tecnica Superior De Ingenieria De Sistemas Informaticos
Curso académico	2023-24

2. Profesorado

2.1. Profesorado implicado en la docencia

Nombre	Despacho	Correo electrónico	Horario de tutorías *
Maria Isabel Muñoz Fernandez	4412	isabel.munoz@upm.es	Sin horario. Sin horario. Las tutorías serán publicadas al principio del 2º Semestre en función de los horarios de impartición de las clases.

Jorge Enrique Perez Martinez	4415	jorgeenrique.perez@upm.es	Sin horario. Sin horario. Las tutorías serán publicadas al principio del 2º Semestre en función de los horarios de impartición de las clases.
Sergio Arevalo Viñuales	4413	sergio.arevalo@upm.es	Sin horario. Sin horario. Las tutorías serán publicadas al principio del 2º Semestre en función de los horarios de impartición de las clases.
Pilar Manzano Garcia	4412	pilar.manzano@upm.es	Sin horario. Sin horario. Las tutorías serán publicadas al principio del 2º Semestre en función de los horarios de impartición de las clases.
Carlos Camacho Gomez	4104	carlos.camacho@upm.es	Sin horario. Sin horario. Las tutorías serán publicadas al principio del 2º Semestre en función de los

			horarios de impartición de las clases
Gustavo Adolfo Hernandez Peñaloza (Coordinador/a)	4408	gustavo.hernandez.penaloza@upm.es	Sin horario. Sin horario. Las tutorías serán publicadas al principio del 2º Semestre en función de los horarios de impartición de las clases
Sergio Gil Borras	4211	sergio.gil@upm.es	Sin horario. Sin horario. Las tutorías serán publicadas al principio del 2º Semestre en función de los horarios de impartición de las clases

* 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 Ingenieria del Software 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

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

RA188 - Utiliza los monitores como herramienta de alto nivel para resolver problemas de programación concurrente

RA315 - Utiliza los distintos modelos de paso de mensajes para resolver problemas de programación concurrente, eligiendo el más adecuado en cada situación

RA194 - 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

RA184 - Aplica las teorías y técnicas necesarias para desarrollar correctamente programas concurrentes

RA314 - Detecta situaciones de interbloqueo y aplica mecanismos para afrontar dichas situaciones

RA185 - Desarrolla aplicaciones que ejecutan hilos concurrentes usando distintas herramientas

RA186 - 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

RA187 - Utiliza los semáforos como herramienta de bajo nivel para resolver problemas de programación concurrente

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 real) 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 en aula	Actividad en laboratorio	Tele-enseñanza	Actividades de evaluación
1	**Presentación. Tema 1. Introducción a la programación concurrente (RA 184, CT7)** Duración: 02:00 LM: Actividad del tipo Lección Magistral	**Lab 0: Introducción al modelo de prácticas, a los entornos que se utilizarán y a la descarga y utilización de los mismos** Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		
2	**Tema 2. Abstracción de la PC (RA184, CT7)** Duración: 02:00 LM: Actividad del tipo Lección Magistral	**Lab 1: Scala (0) (RA184)** Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		
3	**Tema 3. El problema de la región crítica (I) (RA185, RA186, CT7)** Duración: 02:00 LM: Actividad del tipo Lección Magistral	**Lab 2: Scala (I) (RA184)** Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		
4	**Tema 3. El problema de la región crítica (II) (RA185, RA186, CT7)** Duración: 02:00 LM: Actividad del tipo Lección Magistral	**Lab 3: Scala (II) (RA184, CT7)** Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		
5	**Tema 4. Semáforos (I) (RA185, RA187, CT7)** Duración: 02:00 LM: Actividad del tipo Lección Magistral	**Lab 4: Threads (RA185)** Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		
6	**Tema 4. Semáforos (II) (RA185, RA187, CT7)** Duración: 02:00 LM: Actividad del tipo Lección Magistral	**Lab 5: Semáforos (I) (RA185, RA187)** Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		
7	**Ejercicios (RA184, RA185. RA186, RA187, CT7, CC14)** Duración: 02:00 LM: Actividad del tipo Lección Magistral	**Lab 6: Semáforos (II) (RA185, RA187, CT7)** Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		
8				**Prueba de evaluación escrita de temas 1-4 (RA184, RA185. RA186, RA187, CC14)** EX: Técnica del tipo Examen Escrito Evaluación continua Presencial Duración: 01:30 **Prueba de evaluación escrita de prácticas Lab 1-6 (RA184, RA185. RA186, RA187. CC14)** EX: Técnica del tipo Examen Escrito Evaluación continua Presencial Duración: 01:30

9	**Tema 5. Monitores (I) (RA185, RA188, CT7)** Duración: 02:00 LM: Actividad del tipo Lección Magistral	**Lab E: Programación y validación en ordenador de las cuestiones contenidas en la prueba de evaluación** Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		
10	**Tema 5. Monitores (II) (RA185, RA188, CT7)** Duración: 02:00 LM: Actividad del tipo Lección Magistral	**Lab 7: Monitores en jBACI (RA185, RA188)** Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		
11	**Tema 6. Canales y paso de mensajes (I) (RA185, RA315, CT7)** Duración: 02:00 LM: Actividad del tipo Lección Magistral	**Lab 8: Monitores en Scala (RA185, RA188, CT7)** Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		
12	**Tema 6. Canales y paso de mensajes (II) (RA185, RA315, CT7)** Duración: 02:00 LM: Actividad del tipo Lección Magistral	**Lab 9: Actores (I) (RA185, RA315)** Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		
13	**Tema 7.Interbloqueos (RA314, CT7)** Duración: 02:00 LM: Actividad del tipo Lección Magistral	**Lab 10: Actores (II) (RA185, RA315, CT7)** Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		
14	**Tema 8. Prog. Asíncrona con futuros (RA185, RA315, CT7)** Duración: 02:00 LM: Actividad del tipo Lección Magistral	**Lab 11: Futuros (RA185, RA315)** Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		
15	**Tema 9. Programación avanzada (RA316, 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 (RA184, RA185. RA186, RA187, CT7, CC14)** Duración: 02:00 LM: Actividad del tipo Lección Magistral	**Lab 13: Programación Distribuida (RA315, CT7)** Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		
17				**Prueba de evaluación escrita de temas 5-9 (Todos los RA, CC14)** EX: Técnica del tipo Examen Escrito Evaluación continua y sólo prueba final Presencial Duración: 01:30 **Prueba de evaluación escrita de temas 1-4 (RA184, RA185. RA186, RA187, CC14)** EX: Técnica del tipo Examen Escrito Evaluación sólo prueba final Presencial Duración: 01:30 **Prueba de evaluación escrita de prácticas Lab 7-13 (Todos los RA, CC14)** EX: Técnica del tipo Examen Escrito Evaluación continua y sólo prueba final Presencial Duración: 01:30 **Prueba de evaluación escrita de prácticas Lab 1-6 (RA184, RA185. RA186,**

				RA187. CC14) EX: Técnica del tipo Examen Escrito Evaluación sólo prueba final Presencial 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.

7. Actividades y criterios de evaluación

7.1. Actividades de evaluación de la asignatura

7.1.1. Evaluación (progresiva)

Sem.	Descripción	Modalidad	Tipo	Duración	Peso en la nota	Nota mínima	Competencias evaluadas
8	Prueba de evaluación escrita de temas 1-4 (RA184, RA185. RA186, RA187, CC14)	EX: Técnica del tipo Examen Escrito	Presencial	01:30	25%	3 / 10	
8	Prueba de evaluación escrita de prácticas Lab 1-6 (RA184, RA185. RA186, RA187. CC14)	EX: Técnica del tipo Examen Escrito	Presencial	01:30	25%	3 / 10	
17	Prueba de evaluación escrita de temas 5-9 (Todos los RA, CC14)	EX: Técnica del tipo Examen Escrito	Presencial	01:30	25%	4 / 10	CC14
17	Prueba de evaluación escrita de prácticas Lab 7-13 (Todos los RA, CC14)	EX: Técnica del tipo Examen Escrito	Presencial	01:30	25%	4 / 10	CC14

7.1.2. Prueba evaluación global

Sem	Descripción	Modalidad	Tipo	Duración	Peso en la nota	Nota mínima	Competencias evaluadas
17	Prueba de evaluación escrita de temas 5-9 (Todos los RA, CC14)	EX: Técnica del tipo Examen Escrito	Presencial	01:30	25%	4 / 10	CC14
17	Prueba de evaluación escrita de temas 1-4 (RA184, RA185. RA186, RA187, CC14)	EX: Técnica del tipo Examen Escrito	Presencial	01:30	25%	3 / 10	CC14
17	Prueba de evaluación escrita de prácticas Lab 7-13 (Todos los RA, CC14)	EX: Técnica del tipo Examen Escrito	Presencial	01:30	25%	4 / 10	CC14

17	Prueba de evaluación escrita de prácticas Lab 1-6 (RA184, RA185. RA186, RA187. CC14)	EX: Técnica del tipo Examen Escrito	Presencial	01:30	25%	3 / 10	CC14

7.1.3. Evaluación convocatoria extraordinaria

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

7.2. Criterios de evaluación

Sea:

T1: examen de teoría del primer parcial, realizado en la semana 8

P1: examen de prácticas del primer parcial, realizado en la semana 8

T2: examen de teoría del segundo parcial, realizado en la semana 17

P2: examen de prácticas del segundo parcial, realizado en la semana 17

Siguiendo la normativa de evaluación, en la semana 17 se realizarán de nuevo exámenes para cubrir los contenidos teóricos y prácticos del primer parcial. A estos exámenes puede concurrir cualquier estudiante que lo desee manteniéndose la calificación más alta obtenida. Por tanto,

T1?: examen de teoría del primer parcial, realizado en la semana 17

P1?: examen de prácticas del primer parcial, realizado en la semana 17

Se define la función max como:

max (a, b: REAL): REAL; devuelve el mayor valor de ambos parámetros a y b

y definimos:

NotaTeoriaPrimerParcial:= max (T1, T1?)

NotaPrácticaPrimerParcial:= max (P1, P1?)

La asignatura calcula la semisuma de las calificaciones obtenidas en la parte teórica y la semisuma de las calificaciones obtenidas en la parte práctica en ambos parciales. Para realizar las medias es condición indispensable tener al menos una calificación de 3 en ambas pruebas del primer parcial y una calificación de al menos 4 en ambas pruebas del segundo parcial; es decir:

compensableNT:= FALSE

NotaTeoría: = (NotaTeoriaPrimerParcial + P2) / 2

IF (NotaTeoriaPrimerParcial >= 3) AND (T2 >= 4)

THEN CompensableNT:= True

compensableNP:= FALSE

NotaPráctica:= (NotaPrácticaPrimerParcial + P2) / 2

IF (NotaPrácticaPrimerParcial >= 3) AND (P2 >= 4)

THEN compensableNP:= TRUE

La nota total se calcula como la semisuma de la nota de teoría y la nota de prácticas, supuestas que ambas sean compensables. Definimos la función min como:

min (a, b: REAL): REAL; devuelve el menor valor de ambos parámetros a y b

IF compensableNT AND compensableNP

THEN notaTotal:= (NotaTeoría+NotaPractica)/2

ELSE notaTotal:= min (4.5, ((NotaTeoría+NotaPractica)/2))

CT:= 0,5

IF notaTotal >= 5

THEN notaFinal:= min (10, (notaTotal+CT))

ELSE notaFinal:= notaTotal

/* CT: Competencia Transversal: la superación de la asignatura supone la superación de esta CT y suma 0,5 puntos a la nota */

CONVOCATORIA EXTRAORDINARIA

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 (nota >= 3,5). La nota que se mantiene es la última que obtenga; es decir, si opta por examinarse de un bloque compensado y obtiene menos de un 3,5, habría suspendido la asignatura.

Teoria := NotaTeoría /* asignamos primero la nota de teoria de Junio */

Practica := NotaPractica /* asignamos primero la nota de practica de Junio */

IF NOT compensableNT THEN

  Teoría := ExamenTeoríaJulio

IF NOT compensableNP THEN

Prácticas := ExamenPrácticasJulio

IF (Teoría >= 3,5) AND (Prácticas >= 3,5)

THEN NotaFinal := (Teoría + Prácticas)/2

ELSE NotaFinal:= min (4.5, ((Teoría+Prácticas)/2))

siendo min (a, b: REAL): REAL; devuelve el menor valor de ambos parámetros a y b

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 Martinez, 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
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.

Se utilizará la plataforma Moodle como repositorio compartido de información.

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.