



Universitat de Lleida

GUIA DOCENT
**SISTEMES CONCURRENTS I
PARAL·LELS**

Coordinació: CORES PRADO, FERNANDO

Any acadèmic 2022-23

Informació general de l'assignatura

Denominació	SISTEMES CONCURRENTS I PARAL·LELS			
Codi	102022			
Semestre d'impartició	1R Q(SEMESTRE) AVALUACIÓ CONTINUADA			
Caràcter	Grau/Màster	Curs	Caràcter	Modalitat
	Doble titulació: Grau en Enginyeria Informàtica i Grau en Administració i Direcció d'Empreses	4	OBLIGATÒRIA	Presencial
	Grau en Enginyeria Informàtica	3	OBLIGATÒRIA	Presencial
	Màster Universitari en Enginyeria Informàtica		COMPLEMENTES DE FORMACIÓ	Presencial
Nombre de crèdits assignatura (ECTS)	6			
Tipus d'activitat, crèdits i grups	Tipus d'activitat	PRALAB		TEORIA
	Nombre de crèdits	3		3
	Nombre de grups	2		1
Coordinació	CORES PRADO, FERNANDO			
Departament/s	INFORMÀTICA I ENGINYERIA INDUSTRIAL			
Distribució càrrega docent entre la classe presencial i el treball autònom de l'estudiant	6 ECTS = 25x6 = 150 hores de treball 40% --> 60 hores presencials 60% --> 90 hores de treball autònom de l'estudiant			
Informació important sobre tractament de dades	Consulteu aquest enllaç per a més informació.			
Idioma/es d'impartició	Preferentment en Català, en Castellà si algú no entén bé la llengua Catalana.			
Distribució de crèdits	Marc Viladegut 3.0 Jordi Onrubia 6.0			

Professor/a (s/es)	Adreça electrònica professor/a (s/es)	Crèdits impartits pel professorat	Horari de tutoria/lloc
CORES PRADO, FERNANDO	fernando.cores@udl.cat	9	

Informació complementària de l'assignatura

L'assignatura és imminentment pràctica, per la qual cosa se li donarà més pes a les pràctiques i la programació. Bàsicament treballarem amb dos llenguatges, C per a la programació concurrent amb fils d'execució i Java per a introduir les Apis d'alt nivell per a la sincronització i els patrons concurrents. Cap d'aquests llenguatges s'explicarà des de zero, a causa que tant C com Java ja s'han vist en assignatures prèvies de la titulació. Per poder seguir l'assignatura és imprescindible que els estudiants tinguin uns bons fonaments de programació en C i Java. En l'assignatura es dóna per suposat que els estudiants són capaços de dissenyar, desenvolupar i depurar aplicacions seqüencials de dificultat mitjana sense problemes.

Objectius acadèmics de l'assignatura

- Entendre la importància de la programació concurrent en les aplicacions actuals.
- Identificar les principals característiques dels diferents tipus de sistemes concurrents que existeixen.
- Conèixer i entendre els problemes que planteja el desenvolupament de programes concurrents i que no apareixen en la programació seqüencial.
- Entendre els conceptes de sincronització i exclusió mútua entre processos.
- Identificar les propietats de seguretat i vivacitat que un sistema concurrent ha de complir i ser capaç de raonar si aquestes propietats es compleixen.
- Adquirir experiència i coneixement en els mecanismes de sincronització i comunicació que s'utilitzen en l'actualitat per desenvolupar programes concurrents tant per a sistemes de memòria compartida com per a sistemes distribuïts.
- Entendre el funcionament de semàfors i monitors com a mecanismes de sincronització per a memòria compartida i comprendre com es poden resoldre problemes de programació concurrent utilitzant monitors.
- Aplicar metodologies de l'enginyeria del programari en el desenvolupament d'aplicacions concurrents i paral·leles.

Competències

Competències transversals EPS:

- EPS7. Capacitat de treballar en situacions de falta d'informació i/o sota pressió.

Competències transversals globals:

- GII-CRI11. Coneixement i aplicació de les característiques, funcionalitats i estructura dels Sistemes Distribuïts, les Xarxes de Computadors i Internet i dissenyar i implementar aplicacions basades en elles.
- GII-CRI14. Coneixement i aplicació dels principis fonamentals i tècniques bàsiques de la programació paral·lela, concurrent, distribuïda i de temps real.
- GII-CRI16. Coneixement i aplicació dels principis, metodologies i cicles de vida de l'enginyeria de programari.

Continguts fonamentals de l'assignatura

1. Introducció a la concurrència

1. Definició concurrència
2. Necessitat i beneficis de la programació concurrent
3. Arquitectures maquinari concurrents
4. Característiques sistemes concurrents
5. Especificació programes Concurrents
 1. Condicions de *Bernstein*
 2. Taula de concurrència
 3. Grafs de precedència
6. Formes d'expressar la concurrència
 1. Cobegin/Coend
 2. Fork/Join
 3. Unix + C
 4. Exemples
7. Cas d'estudi: Fils d'execució en Linux i Java

1. Disseny aplicacions concurrents

1. Model de programació concurrent / paral·lela
2. Eficiència Programes Concurrents / Paral·lels
 1. Abast del paral·lisme
 2. Granularitat
 3. Localitat
3. Disseny programes concurrents
 1. Etapes de Disseny
 2. Tècniques de Descomposició de tasques
 3. Patrons de disseny paral·lels
 4. Distribució i comunicació de tasques
4. Casos d'estudi.

1. Sincronització de processos concurrents

1. Sincronització
2. El problema de l'exclusió mútua
 1. Solucions Programari
3. Solucions Hardware
4. Semàfors
5. Problemes clàssics de sincronització
6. Cas d'estudi:
 1. Pthread mutex i variables de condició
 2. Sincronització en Java

1. API d'alt nivell per a la concurrència i sincronització

1. Patrons concurrents
2. Monitors
3. Cas d'estudi:
 1. Llibreria Boost
 2. Java Concurrent API

Eixos metodològics de l'assignatura

Grups Grans: Classes Teoria (3 crèdits)

- Lliçó magistral: classes basades en anotacions i transparències on es presentaran els conceptes de l'assignatura.

- Problemes: Els conceptes de l'assignatura es treballaran mitjançant una sèrie de problemes que es resoldran de forma cooperativa i que ajudaran a assimilar els conceptes clau.
- Casos d'ús: S'aplicaran les tècniques vistes en classe a exemples reals i s'analitzarà el seu impacte sobre el rendiment de les aplicacions.

Grups Mitjans: Classes Laboratori (3 crèdits)

- Classes dirigides i seguiment personalitzat per grups de pràctiques
- Pràctiques de laboratori: es presentaran les tecnologies i APIs de programació concurrent i es treballin de forma pràctica mitjançant tutorials i exemples.
- Problemes: Realització i correcció de problemes relacionats tant amb la part teòrica com a pràctica de l'assignatura.

Treball Autònom (no presencial):

- Les pràctiques i els problemes es completaran fora de l'horari de classe.

Pla de desenvolupament de l'assignatura

S	Descripció:	Activitat Presencial GG	Activitat Presencial (GM) Dimarts PraLab1 i PraLab2	Activitat treball autònom
1	Presentació Introducció Concurrència	Presentació Assignatura T1: Introducció Concurrència	L1: Posix Threads	Estudiar bibliografia i programa
2	Introducció Concurrència	T1: Introducció Concurrència	L1: Posix Threads Pràctica 1: Presentació	T1: Introducció Concurrència
3	Introducció Concurrència	T1: Introducció Concurrència	Dimarts Festa (Lab1 i Lab2)	Pràctica 1 Problemes: Concurrència
4	Disseny aplicacions concurrents	T1: Introducció Concurrència T2: Disseny aplicacions concurrents	L2: Java Threads Pràctica 2: Presentació	Pràctiques 1 i 2 T2: Disseny
5	Disseny aplicacions concurrents	T2: Disseny aplicacions concurrents	Dimarts Festa (Lab1 i Lab2)	Pràctiques 1 i 2 Problemes: Disseny
6	Disseny aplicacions concurrents	T2: Disseny aplicacions concurrents	Problemes: Introducció Concurrència Problema P1 Eval. Cont.	Pràctiques 1 i 2 Problemes: Disseny
7	Disseny aplicacions concurrents	T2: Disseny aplicacions concurrents	Correcció P1 Eval. Cont. Problemes: Disseny	Pràctiques 1 i 2
8	Sincronització	T3: Sincronització processos concurrents	Problemes: Disseny Lliurament practiques 1 i 2	T3: Sincronització
9		1er Parcial		Estudiar
10	Sincronització	T3: Sincronització processos concurrents	L3: Sincronització Linux Pràctica 3: Presentació	T3: Sincronització
11	Sincronització	T3: Sincronització processos concurrents	L4: Sincronització Java Pràctica 4: Presentació	Pràctica 3 Problemes: Sincronització
12	Sincronització	T3: Sincronització processos concurrents	L4: Sincronització Java Problema P2 Eval. Cont.	Pràctiques 3 i 4 Problemes: Sincronització

13	Sincronització	T3: Sincronització processos concurrents	Dimarts Festa (Lab1 y Lab2)	Pràctiques 3 i 4 Problemes: Sincronització
14	APIs concurrència i sincronització	T4: API d'alt nivell per a la concurrència i sincronització	Correcció P2 Eval. Cont. Pràctica 5: Presentació	Pràctiques 3 i 4 T4: APIs alt nivell
15	APIs concurrència i sincronització	Dijous Festa (Nadal)	Lliurament practiques 3 i 4	Pràctica 5 T4: APIs alt nivell
16		2^{on} Parcial		Estudiar
17		2^{on} Parcial		Estudiar
18		TUTORIAS		
19		Recuperació		Estudiar

Sistema d'avaluació

Acr.	Activitats d'Avaluació	Ponderació	Nota Mínima	En grup	Obligatòria	Recuperable
P1	Examen 1 ^{er} Parcial	25%	NO	NO	NO	SI
P2	Examen 2 ^{on} Parcial	25%	NO	NO	NO	SI
PRA	Pràctiques	40%	4	SI (<=2)	SI	1 Prác.
PRB	Problemes	10%	NO	SI (<=2)	NO	NO
<i>S'han d'aprovar totes les pràctiques de forma individual. Una pràctica es considera suspesa si no arriba a un 4.</i>						
NotaFinal = 0,25*P1 + 0,25*P2 + 0,4*PRA + 0,1*PRB						

L'assignatura s'aprova amb una nota final igual o superior 5 i havent realitzat correctament les pràctiques de laboratori (totes elles amb almenys un 4 de nota).

La nota final de l'assignatura s'obté a partir de la suma ponderada de les notes dels dos exàmens i les pràctiques, més les notes de participació a classe i l'avaluació continuada.

L'assignatura té dos parcials, cada un d'ells amb un pes del 25% a la nota final. Aquestes proves eliminen matèria.

La realització i superació de les pràctiques de laboratori és obligatòria per aprovar l'assignatura. Les pràctiques s'avaluaran amb una nota que representarà el 40% de la nota final de l'assignatura. La còpia de qualsevol de les pràctiques de l'assignatura implicarà suspendre totes les pràctica i amb això l'assignatura.

Hi ha nota mínima (4) per a les pràctiques de laboratori. Una pràctica es considera suspesa si no arriba a un 4.

Bibliografia i recursos d'informació

Bibliografia Bàsica:

- José Tomás Palma Méndez, M. C. Garrido Carrera, F. Sanchez Figueroa, A. Quesada Arencibia, "Programación Concurrente", Thomson, 2003.
- Maurice Herlihy, Nir Shavit, "The Art of Multiprocessor Programming", Morgan Kaufmann, 2008.
- Douglas Lea, "Concurrent Programming in Java: Design Principles and Patterns", Addison-Wesley Professional, 2000.

Bibliografia Ampliada:

- Gadi Taubenfeld, "Synchronization Algorithms and Concurrent Programming", Pearson / Prentice Hall, 2006
- M. Ben-Ari, "Principles of Concurrent and Distributed Programming", Addison-Wesley, 2nd Edition, 2006
- KayA. Robbins, Steven Robbins, "UNIX Programación Práctica. Guía para la Concurrencia, la Comunicación y los Multihilos", Edt.Prentice-Hall, 1997.