



Universitat de Lleida

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

Any acadèmic 2013-14

Informació general de l'assignatura

Denominació	SISTEMES CONCURRENTS I PARAL·LELS
Codi	102022
Semestre d'impartició	1r Q Avaluació Continuada
Caràcter	Obligatòria
Nombre de crèdits ECTS	6
Crèdits teòrics	3
Crèdits pràctics	3
Departament/s	Informàtica i Enginyeria Industrial
Distribució càrrega docent entre la classe presencial i el treball autònom de l'estudiant	60 hores presencials 90 hores treball autònom
Informació important sobre tractament de dades	Consulteu aquest enllaç per a més informació.
Idioma/es d'impartició	Castellà
Distribució de crèdits	Fernando Cores 6.0
Horari de tutoria/lloc	Dilluns 16h-17h (s3/17) Dijous 12h-13h (s3/17)

Fernando Cores Prado

Informació complementària de l'assignatura

La assignatura es inminentement pràctica, por lo cual se le dará bastante peso a las prácticas y la programación. Básicamente trabajaremos con tres lenguajes, C para la programación concurrente con hilos, Java para introducir las Apis de alto nivel para la sincronización y los patrones concurrentes.

Para poder seguir la asignatura es imprescindible que los estudiantes tengan unos buenos fundamentos de programación en C y Java. Es mucho más difícil, aprender a desarrollar programas concurrentes y paralelos si no se domina previamente la programación secuencial. En la asignatura se da por supuesto que los estudiantes son capaces de diseñar, desarrollar y depurar aplicaciones secuenciales de dificultad media sin muchos problemas. La indústria informàtica està patint, si no una altra revolució, sens dubte una vigorosa agitació. Els principals fabricants de processadors han renunciat, de moment, a tractar de dissenyar / fabricar processadors que funcionin més ràpid. La Llei de Moore encara no ha estat derogada: cada any, més i més transistors caben en un mateix espai, però la seva velocitat de rellotge no es pot augmentar a causa del sobreescalfament i l'alt-consum associat amb les altes freqüències de rellotge. En lloc d'incrementar la velocitat dels processadors, els fabricants han optat per les arquitectures "multi-nucli", en les que múltiples processadors (nuclis) s'integren conjuntament dins d'un xip, els quals es poden comunicar directament a través de memòries cache compartides. Els xips multiprocessador realitzen els càlculs de forma més eficient mitjançant l'explotació del paral·lelisme: l'aprofitament de múltiples processadors per treballar en una única tasca.

La generalització de les arquitectures multiprocessador tindrà un impacte generalitzat sobre com es dissenya i desenvolupa de programari futur. Fins fa poc, els avenços en la tecnologia implicava també avenços en la velocitat de rellotge, de manera que el programari accelerava efectivament la seva velocitat d'execució a mesura que passava el temps. Ara, però, aquesta millora automàtica de les aplicacions ha acabat. Els avenços en la tecnologia significaran un major paral·lelisme i no augmentar la velocitat de rellotge, per tant, l'explotació d'aquest paral·lelisme és un dels majors reptes pendents de la informàtica moderna.

Aquesta assignatura se centra en el disseny d'aplicacions concurrents i paral·leles capaços aprofitar de tota la capacitat de processament dels sistemes multiprocessador per millorar el rendiment de les aplicacions. En desenvolupar aquestes aplicacions sorgeixen desafiaments de programació a tots els nivells del sistema multiprocessador - a una escala molt petita, dins dels nuclis d'un mateix xip, cal coordinar l'accés a una mateixa posició de memòria compartida, ja gran escala, es necessita coordinar l'enrutament de les dades entre els processadors d'un superordinador. La programació concurrent i paral·lela representa un desafiament major a causa que els moderns sistemes de computació són inherentment asíncrons: les activitats es pot aturar o retardar sense previ avís per les interrupcions, preferència, fallades de memòria cau, les excepcions, i altres esdeveniments. Aquests retards són inherentment impredecibles, i pot variar enormement en durada, provocant que el comportament d'una aplicació concurrent / paral·lela pugui no ser determinista (diferents execucions amb els mateixos paràmetres d'entrada poden generar diferents resultats) si no es programa correctament.

En aquest curs, estudiarem les tècniques de programació concurrent, especialment adequades per obtenir el màxim rendiment dels sistemes multiprocessador. Anem a centrar-nos en els diferents models d'expressar la concurrència, la utilització dels fils d'execució per al disseny d'aplicacions concurrents, els mecanismes de sincronització tant de baix nivell (semàfors) com d'alt nivell (monitors i variables de condició) per coordinar les tasques d'una aplicació concurrent i diferents APIs d'alt nivell que facilitin la programació d'aquestes aplicacions.

Objectius acadèmics de l'assignatura

Veure apartat de competències.

Competències

Competències específiques de la titulació

- Coneixement i aplicació dels principis, metodologies i cicles de vida de l'enginyeria de programari.

Objectius

- Aplicar metodologies de l'enginyeria del programari en el desenvolupament d'aplicacions concurrents i paral·leles.
- Coneixement i aplicació dels principis fonamentals i tècniques bàsiques de la programació paral·lela, concurrent, distribuïda i de temps real.

Objectius

- Conocer los principales modelos de programación concurrente, paralela y distribuida.
- Adquirir experiencia y conocimiento en los mecanismos de sincronización y comunicación que se utilizan en la actualidad para desarrollar programas concurrentes tanto para sistemas de memoria compartida como para sistemas distribuidos.
- Entender el funcionamiento de semáforos y monitores como mecanismos de sincronización para memoria compartida y comprender cómo se pueden resolver problemas de programación concurrente usando monitores.
- Entender la importancia de la programación concurrente en las aplicaciones actuales.
- Identificar las principales características de los distintos tipos de sistemas concurrentes que existen.
- Conocer y entender los problemas que plantea el desarrollo de programas concurrentes y que no aparecen en la programación secuencial.
- Entender los conceptos de sincronización y exclusión mutua entre procesos.

Competències transversals de la titulació

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

Objectius

- Identificar las propiedades de seguridad y vivacidad que un sistema concurrente debe cumplir y ser capaz de razonar si dichas propiedades se cumplen.

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. Cas d'estudi: Fils d'execució en Linux i Java

2. Disseny aplicacions concurrents

1. Models concurrents
 1. Tècniques de descomposició
 2. Balanceig de càrrega
 3. Exemples
2. Anàlisi de dependències
 1. Condicions de Bernstein
 2. Taules de precedència
 3. Grafs de precedència

3. Formes d'expressar la concurrència
 1. Cobegin/Coend
 2. Fork/Join
 3. Unix+C

3. 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

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

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

5. Optimització programes concurrents

1. Eines de depuració i monitorització
2. Principals problemes de rendiment.
 1. Balanceig de Carrega
 2. Colls d'ampolla
 3. Jerarquia de memòria
 4. Deadlock / Sincronització

Pla de desenvolupament de l'assignatura

Sem	Descripció:	Activitat Presencial GG	Activitat Presencial (GM)	HTP (6 Hrs)	Activitat treball autònom	HTNP (9 Hrs)
1	Introducció Concurrència	Presentació Assignatura T1: Introducció Concurrència	L1: Posix Threads	4	T1: Introducció Concurrència	6
2	Introducció Concurrència	T1: Introducció Concurrència	L2: Java Threads Pràctica1: Presentació	4	T1: Introducció Concurrència	6
3	Disseny aplicacions concurrents	T2: Disseny aplicacions concurrents	L2: Java Threads Problemes: Disseny	4	Pràctica1 T2: Disseny	5

SISTEMES CONCURRENTS I PARAL·LELS 2013-14

4	Disseny aplicacions concurrents	T2: Disseny aplicacions concurrents	Problemes: Disseny	4	Pràctica1 Problemes: Disseny	6
5	Disseny aplicacions concurrents	T2: Disseny aplicacions concurrents	Problemes: Disseny	4	Pràctica1 Problemes: Disseny	5
6	Sincronització	T3: Sincronització processos concurrents	Pràctica1: Entrega Correcció P1 Eval. Cont.	4	Problemes: Disseny	6
7	Sincronització	T3: Sincronització processos concurrents Repàs 1er Parcial	L3: Sincronització Linux Pràctica2: Presentació	4	T3: Sincronització	7
8		1er Parcial			Estudiar	6
9	Sincronització	Correcció Parcial 1 T3: Sincronització processos concurrents	L4: Sincronització Java	4	Pràctica2 T3: Sincronització	4
10	Sincronització	T3: Sincronització processos concurrents	Problemes: Sincronització	4	Pràctica2 Problemes: Sincronització	8
11	Sincronització	T3: Sincronització processos concurrents	Pràctica2: Entrega	4	Pràctica2 Problemes: Sincronització	7
12	APIs concurrència i sincronització	T4:API d'alt nivell per a la concurrència i sincronització	Pràctica3: Presentació Problemes: Sincronització	4	Problemes: Sincronització T4: APIs alt nivell	7
13	APIs concurrència i sincronització	T4:API d'alt nivell per a la concurrència i sincronització	Correcció P2 Eval. Cont.	4	T4: APIs alt nivell Pràctica3	7
14	Optimització programes concurrents	T5: Optimització programes concurrents	Pràctica3: Entrega Pràctica4: Presentació	4	Pràctica3 Pràctica4	7
15	Optimització programes concurrents	T5: Optimització programes concurrents Repàs 2nd Parcial	Pràctica4: Entrega	4	Pràctica4	7
16		2on Parcial		2	Estudiar	6
17						
18						
19		Recuperació		2		5

Sistema d'avaluació

Actividad de Evaluación	Ponderación	Nota Mínima	En grupo	Obligatoria
-------------------------	-------------	-------------	----------	-------------

<i>Examen 1^{er} Parcial</i>	25%	NO	NO	SI
<i>Examen 2^{on} Parcial</i>	25%	NO	NO	SI
<i>Pràcticas</i>	40%	5	SI (≤ 2)	SI
<i>Problemas</i>	10%	NO	SI (≤ 2)	NO
<i>Participaci3n Clase</i>	1 punto	NO	NO	NO

Bibliografia i recursos d'informaci3n

Bibliografia Bàsica:

- [Pal03] Jos3 Tomàs Palma M3ndez, M. C. Garrido Carrera, F. Sanchez Figueroa, A. Quesada Arencibia, "Programaci3n Concurrente ", Thomson, 2003.
- [Her08] Maurice Herlihy, Nir Shavit, "The Art of Multiprocessor Programming", Morgan Kaufmann, 2008.
- [Dou00] Douglas Lea, "Concurrent Programming in Java: Design Principles and Patterns", Addison-Wesley Professional, 2000.
- [Pac11] Peter Pacheco, "An Introduction to Parallel Programming", Morgan Kaufmann, 2011.

Bibliografia Ampliada:

- [Gad06] Gadi Taubenfeld, "Synchronization Algorithms and Concurrent Programming", Pearson / Prentice Hall, 2006
- [Ben06] M. Ben-Ari, "Principles of Concurrent and Distributed Programming", Addison-Wesley, 2nd Edition, 2006
- [Nic96] B. Nichols, D. Buttlar, "Pthreads Programming", O'Reilly, 1996
- [Kay97] Kay A. Robbins, Steven Robbins, "UNIX Programaci3n Pràctica. Guía para la Concurrencia, la Comunicaci3n y los Multihilos", Edt. Prentice-Hall, 1997.
- [Afz97] Afzal, Amir, "Introducci3n a UNIX un enfoque pràctico ", Edt. Prentice-Hall,