

## Arquitectura de computadores II

Codi	Tipus	Curs/Semestre	Crèdits
20369	Troncal Semestral	4art / 2on	5

## Objectius

---

### Competències específiques

#### Coneixements

- Definicions, classificacions i tendències sobre la arquitectura del computador.
- Model del rendiment i cost del computador y del seus elements.
- Efecte en las prestacions de les diferents alternatives de disseny del computador
- Mecanismes i algoritmes bàsics d'optimització de prestacions implementats en *hardware*: segmentació, duplicació de recursos, planificació estàtica i dinàmica de l'execució de les instruccions, execució simultània de *threads*, multi-processoradors, memòria cau, coherència de memòria.
- Algoritmes bàsics d'optimització de prestacions implementats en *software*: desenrotllat de bucles, segmentació software, augment de la localitat d'accés a memòria
- Model de rendiment de multi-processoradors.
- Algoritmes paral·lels: operacions bàsiques i la seva especificació amb llenguatges paral·lels actuals

#### Habilitats

- Avaluar el rendiment del computador y dels seus elements
- Dissenyar programes optimitzats per a rendiment en un computador mono-processorador
- Dissenyar programes optimitzats per a rendiment en un computador multi-processorador
- Anàlisi del coll d'ampolla en el rendiment d'un programa

### Competències genèriques

- Aprenentatge autònom i presa de decisions
- Resolució de problemes
- Comunicació oral i escrita
- Treball en equip
- Raonament crític

## Capacitats prèvies

---

Les derivades d'haver cursat Arquitectura de Computadors I, Estructura de Computadors I i II, i les assignatures relacionades.

## Continguts

---

<b>0. Presentació</b>	
Presentació del Pla Docent de l'assignatura	

<b>1. Les Prestacions dels Computadors</b>	
<p><b>1.1 Mesura de Prestacions</b> La mesura del Rendiment dels computadors. Principis quantitativs del disseny de computadors: Equació del Rendiment. Cost, Consum i Fiabilitat. Tendències tecnològiques i els efectes en les prestacions.</p> <p><b>1.2 Paral·lelisme i Rendiment</b> Execució segmentada de les instruccions. Execució de múltiples instruccions per cycle. El paral·lelisme i les dependències de dades: el camí crític. El paral·lelisme i les dependències de control. El paral·lelisme i els recursos de còmput.</p> <p><b>1.3 Localitat i Rendiment</b> Capacitat i latència: accés a dades locals i remotes. Rendiment del sistema de memòria. Amplada de banda d'accés a les dades. Rendiment del sistema de memòria. Latència d'accés i Paral·lelisme: solapament i anticipació. Rendiment del sistema de memòria.</p>	

<b>2. El Paral·lelisme a nivell de instruccions (ILP)</b>	
<p><b>2.1 Estratègies S/W per incrementar ILP</b> Repertoris amb ILP explícit (VLIW o EPIC). Renomenament de registres / variables Gestió estàtica de salts: predicació. Dependències en memòria: especulació. Desenrotllat de bucles i desenrotllat simbòlic.</p> <p><b>2.2 Estratègies H/W per incrementar ILP</b> Filosofia Superescalar: renomenament dinàmic de registres i planificació dinàmica de l'execució. Predicció dinàmica de salts i de dependències en memòria: execució especulativa.</p> <p><b>2.3 Jerarquia de Memòria</b> Estratègies H/W i S/W per reduir la latència a memòria i el seu efecte en el rendiment. Estratègies H/W i S/W per augmentar la amplada de banda de la memòria.</p>	

--	--

### 3. El Paral·lelisme a nivell de threads (TLP)

#### 3.1 TLP i Rendiment

Paral·lelisme de Threads i Tasques.  
Repartiment de la càrrega de còmput.  
Comunicació entre threads i repartiment de les comunicacions.  
Solapament de còmput i comunicacions.  
Anàlisi de Rendiment i els seus Límits.

#### 3.2 Algoritmes i Programació Paral·lela

Operacions primitives per a la creació de threads i per a compartir dades: model de memòria compartida i model de memòria distribuïda.  
Llenguatges de programació en paral·lel: MPI i OpenMP.  
Exemples d'algorismes Paral·lels.

#### 3.3 Multiprocessadors i Multicomputadors

Hyperthreading: TLP en processadors superescalars. Exemple: OpenSPARC.  
Multi-cores i nodes multiprocessador amb memòria compartida. Coherència de dades en caché.  
Exemple: OpenSPARC.  
Multicomputadors i xarxes d'interconnexió.

## Metodologia docent

#### Activitats a CLASSE de Teoria (2 hores els dilluns):

- Assistència obligatòria per obtenir crèdits extra d'anglès.
- 80 minuts/sessió exposar temari bàsic (model classe magistral): discussió de problemes, proposta de solucions, i resolució de dubtes.
- 20 minuts/sessió per a realitzar problemes en grup, no avaluable. Resolució guiada i moderada pel professor.

#### Activitats a CLASSE de Problemes (1 hora el dijous):

- 15 minuts/sessió per a resoldre dubtes de problemes proposats previament.
- 45 minuts/sessió per a resoldre PROBLEMES AVALUABLES individualment. Permis disposar en paper NOMÉS dels problemes exemple amb solució i de les transparències publicats al Campus Virtual. Solució publicada al Campus Virtual. Problema= 2 o més apartats, cada apartat avaluat com CORRECTE (tots els punts) o NO CORRECTE (0 punts)
- proves d'elecció múltiple AVALUABLE individualment (20 minuts i sense apunts)

#### Activitats fora de CLASSE:

- Estudiar apunts, transparències de classe, problemes i preguntes d'elecció múltiple amb i sense solució.
- Estudiar capítols seleccionats del llibre bàsic de referència.

#### Activitats de LABORATORI:

- 2 sessions d'activitats pràctiques guiades (2/3 alumnes per grup). Curt qüestionari previ per validar haver llegit l'enunciat de les pràctiques. Si es supera cal entregar al final de la sessió les respostes i conclusions de les activitats. Si no es supera, caldrà fer i presentar a la setmana següent un informe escrit i detallat de la pràctica (3-4 pàgines).
- Proposta de pràctica (entre les idees suggerides pel professor), disseny de pla de treball, defensa oral del pla de treball, realització i defensa oral dels resultats (grup de 2-3 alumnes).
  - \* **Entrevista de Planificació (10 minuts - Laboratori):** tots els membres del grup, presentació per escrit (una pàgina) del pla de treball, i discussió oral de la proposta.
  - \* 3 sessions x 2 hores al laboratori per a realitzar les activitats experimentals relacionades amb el treball i resoldre dubtes amb el professor.
  - \* **Entrevista de Presentació Final (15 minuts - Laboratori):** tots els membres del grup, informe final del treball (2-4 pàgines), indicant les incidències més notables, els resultats obtinguts, i les conclusions més rellevants.

## Avaluació

1a convocatòria (febrer/juny)		2a convocatòria (juliol/setembre)
Avaluació en grups	Avaluació individual	
Pràctiques al Laboratori	<p>No hi ha examen final.</p> <p>3 Proves d'elecció múltiple sobre coneixements.</p> <p>3x4 Problemes per a avaluar la capacitat d'aplicació dels coneixements teòrics (escollir la nota de 3x3 problemes).</p> <p><b>NO PRESENTAT= avaluat en menys del 20% del total evaluable</b></p>	<p>Haver assistit a sessions de Laboratori i entrevista de planificació de pràctica.</p> <p>Recuperar un únic Bloc temàtic (1,5 hores) o fer examen Complet (4,5 hores) amb problemes i prova d'elecció múltiple.</p> <p>Entregar i presentar pràctica de Laboratori.</p>

## Bibliografia bàsica

---

**Computer Architecture. A Quantitative Approach. Fourth Edition.** J.Hennesy & D. Patterson. Morgan Kaufmann (Elsevier), 2006.

## Bibliografia complementària

---

**Organización y Arquitectura de Computadores.** 5ª Edición. W. Stallings. Prentice Hall, 2000

**Computer Organization and Design. The Hardware/Software Interface, Third Edition.** D. Patterson & J. Hennesy. Morgan Kauffman (Elsevier), 2005.

**Embedded Computing: A VLIW approach to architecture compilers and tools.** Joseph A. Fisher, Paolo Faraboschi, Cliff Young. Morgan Kauffman (Elsevier), 2005.

**Parallel Programming in C with MPI and OpenMP.** Michael J. Quinn. McGraw Hill, 2004

## Enllaços

---