

## 30237 - Multiprocessors

### Syllabus Information

**Academic Year:** 2019/20

**Subject:** 30237 - Multiprocessors

**Faculty / School:** 110 - Escuela de Ingeniería y Arquitectura

**Degree:** 439 - Bachelor's Degree in Informatics Engineering

**ECTS:** 6.0

**Year:** 3

**Semester:** Second semester

**Subject Type:** ---

**Module:** ---

### 1.General information

#### 1.1.Aims of the course

#### 1.2.Context and importance of this course in the degree

#### 1.3.Recommendations to take this course

### 2.Learning goals

#### 2.1.Competences

#### 2.2.Learning goals

#### 2.3.Importance of learning goals

### 3.Assessment (1st and 2nd call)

#### 3.1.Assessment tasks (description of tasks, marking system and assessment criteria)

### 4.Methodology, learning tasks, syllabus and resources

#### 4.1.Methodological overview

**El proceso de aprendizaje que se ha diseñado para esta asignatura se basa en lo siguiente:**

Seguimiento de las actividades de aprendizaje programadas en la asignatura, mediante:

- corrección personalizada de ejercicios propuesto en clase
- tutorías
- seguimiento personalizado en las sesiones de laboratorio

#### 4.2.Learning tasks

The student will be able to achieve the expected results by doing the following activities:

- Lectures
- Problem-solving classes
- Laboratory practices assistance
- Practical non-presential work
- Personalized tutorials on specific aspects
- Study and personal work

### 4.3.Syllabus

#### Module I: Pipelined Vector Processors: Supercomputers

1. Introduction. parallelism
  - + Numerical scientific problems
  - + Performance of an addition of vectors by scalar processors
    - Pipelined, superpipelined, and superscalar
  - + Vector version of the vector addition.
2. Vector Extension of a ld/st architecture
  - + Architecture and Organization
  - + Basic instruction set (DLXV)
  - + Organizations and pipelining
    - Vector register file
    - Functional units (ALUs)
    - Multibank memory (synchronous and concurrent access)
  - + Five organizations of vector processor and basic pipelining
  - + Performance measures without strip mining:  $R_n$ ,  $R?$ ,  $N?$ ,  $N_v$
  - + ZV processor organization: a pipelined vector processor supporting DLXV
3. Two aspects of programming: vector length and vector stride
  - + Vector length and strip mining
  - + Two schemes for strip mining code generation. AXPY example
  - + Performance with strip mining:
    - Assembler example AXPY
    - $R_n$ ,  $R?$ ,  $N?$ ,  $N_v$  when processing noncontiguous elements of a vector (stride)
4. Conflicts in accessing memory banks
  - + Introduction. Storage scheme. Fundamental property.
  - + Tight Systems
  - + Loose Systems
5. DLXV architecture: full instruction set
6. Vector Compilation = automatic extraction of vector operations
  - + Introduction
  - + Previous transformations that simplify dependency analysis
  - + Analysis of dependencies. Dependency graph. Approximate tests
  - + Architecture independent optimizations: rename, scalar expansion, vector copy
  - + Vectorization
    - Basic Procedure. Full vs. partial vectorization: loop distribution and loop exchange. Reduction
7. Final Thoughts: Amdahl's Law
8. Commercial Vector Processors
  - + Introduction
  - + Table of Supercomputers
  - + Family NEC SX-4 and SX-9 ACE (may change)
    - Concept of partitioned data path
  - + Vector Extensions Intel: from SSE to AVX512 (may change)

#### Module II: Shared Memory Multiprocessors

1. Classification of parallel computers from M.J. Flynn
  - + SISD, SIMD and MIMD
2. Objectives and problems of the MIMD machines
3. Simple model of H.S. Stone to distribute processes in processors

- 
4. Shared-memory multiprocessors. Overview
    - + Architecture-Programming: communication, synchronization, process creation
    - + Organization: caches, interconnection network, main memory
  5. Interconnection Network
    - + Conflict, degradation, topology, cost, circuit switching or packet switching , performance, availability
    - + Dynamic Topologies (indirect networks): bus, multibus, crossbar, multi-stage networks
    - + Static Topologies (direct network): star, ring, mesh, tree, hypercube
  6. Synchronization Mechanisms
    - + Instruction set: Test & Set, Fetch & Op, Load Linked
    - + Implementation. Combination of requests
    - + Barriers
  7. Parallel Compilation
    - + Automatic extraction of parallel tasks
  8. The problem of consistency
    - + System, multiprocessor, multi-level cache, more examples
    - + Copy-back and write-through
  9. The memory model
    - + Sequential consistency, pros and cons
    - + A definition of consistency
  10. coherence protocols based on diffusion
    - + Invalidation. Diffusion vs. selective shipping
    - + Examples of invalidation + CB + Bus: MSI, EI, Write Once, MESI
    - + Snoopy protocols
  11. Hierarchy of multilevel caches
  12. coherence protocols based on directory
    - + Hw requirements and some sample transactions
    - + Simple protocol directory
  13. Examples of current chip with more than one processor (core)
    - + SUN, Intel, AMD, ARM, ...

#### 4.4.Course planning and calendar

##### Schedule of sessions and labs:

Please see the academic calendar published by EINA.

##### Expected distribution of student work:

Lectures: 30 hours  
 Problems: 15 hours  
 Labs: 15 hours  
 Personal practice: 12 hours  
 Personal study: 73 hours  
 Rating: 5 hours

#### 4.5.Bibliography and recommended resources

<http://psfunizar7.unizar.es/br13/egAsignaturas.php?codigo=30237&Identificador=14702>

[BB Bibliografía básica] BB Computer architecture : a quantitative approach / John Hennessy, David A. Patterson ; with contributions by Andrea C. Arpaci-Dusseau ... [et al.] . 4th ed. San Francisco : Morgan Kaufmann, 2007 BB Culler, David E.. Parallel Computer Architecture : A Hardware-Software Approach / David E. Culler, Jaswinder Pal Singh ; with Anoop Gupta . - [1st ed.] San Francisco : Morgan Kaufmann, cop. 1999 BB Dally, William James. Principles and practices of interconnection networks / William James Dally, Brian Towles San Francisco : Morgan Kaufmann, cop. 2004 BB Patterson, David A.. Computer organization and desing : the hardware, software interface / David A. Patterson, John L. Hennessy ; with contributions by Perry Alexander ... [et al.] . 5th ed. Amsterdam : Elsevier : Morgan Kaufmann, cop. 2014