

## 30237 - Multiprocesadores

### Información del Plan Docente

**Año académico:** 2020/21

**Asignatura:** 30237 - Multiprocesadores

**Centro académico:** 110 - Escuela de Ingeniería y Arquitectura

**Titulación:** 439 - Graduado en Ingeniería Informática

**Créditos:** 6.0

**Curso:** 3

**Periodo de impartición:** Segundo semestre

**Clase de asignatura:** ---

**Materia:** ---

## 1. Información Básica

### 1.1. Objetivos de la asignatura

**La asignatura y sus resultados previstos responden a los siguientes planteamientos y objetivos:**

Alcanzar los resultados de aprendizaje.

### 1.2. Contexto y sentido de la asignatura en la titulación

Esta asignatura completa los conocimientos en Ingeniería Informática relativos a la organización y arquitectura de computadores en el contexto de la ingeniería de computadores. La explotación del paralelismo, tanto vectorial como espacial (múltiples procesadores) es de hecho la palanca básica de todos los sistemas digitales complejos en la sociedad: desde los teléfonos móviles a los supercomputadores, pasando por todo tipo de sistemas empujados, tablets, laptops, ordenadores de sobremesa o servidores de centros de datos.

### 1.3. Recomendaciones para cursar la asignatura

Para cursar esta asignatura se recomienda haber cursado la asignatura Arquitectura y Organización de Computadores 2.

## 2. Competencias y resultados de aprendizaje

### 2.1. Competencias

**Al superar la asignatura, el estudiante será más competente para...**

Resolver problemas y tomar decisiones con iniciativa, creatividad y razonamiento crítico.

Usar las técnicas, habilidades y herramientas de la Ingeniería necesarias para la práctica de la misma.

Diseñar y construir sistemas digitales, incluyendo computadores, sistemas basados en microprocesador y sistemas de comunicaciones.

Desarrollar procesadores específicos y sistemas empujados, así como desarrollar y optimizar el software de dichos sistemas.

Analizar y evaluar arquitecturas de computadores, incluyendo plataformas paralelas y distribuidas, así como desarrollar y optimizar software para las mismas.

Analizar, evaluar y seleccionar las plataformas hardware y software más adecuadas para el soporte de aplicaciones empujadas y de tiempo real.

Analizar, evaluar, seleccionar y configurar plataformas hardware para el desarrollo y ejecución de aplicaciones y servicios informáticos.

### 2.2. Resultados de aprendizaje

**El estudiante, para superar esta asignatura, deberá demostrar los siguientes resultados...**

Conoce las familias de multiprocesadores, identifica los componentes principales de un multiprocesador y sus funciones. Entiende los problemas de coherencia y consistencia y sus soluciones básicas. Conoce la teoría y la práctica la extracción automática de paralelismo.

Conoce la organización de multiprocesadores comerciales, tanto los integrados en un chip como los formados por varios módulos o placas, especialmente en lo relativo a la memoria y la red de interconexión.

Programa algoritmos sencillos, pero conscientes de la organización del multiprocesador, mediante algún estándar de programación para multiprocesadores.

### 2.3.Importancia de los resultados de aprendizaje

Está claro que desde las primeras etapas de análisis y diseño de software va a ser necesario incorporar capacidades de ejecución paralela y/o distribuida. Para ello va a ser muy importante conocer con cierto grado de detalle las posibilidades y limitaciones de las tecnologías de base.

Esta asignatura permite comprender el panorama actual y futuro de las plataformas de procesado de información con ejecución paralela. Los multiprocesadores de memoria compartida son un ejemplo claro de esta tecnología. Su penetración en el mercado es y va a ser muy grande, ya que prácticamente todos los chips comerciales, para casi todos los segmentos, tienen versiones con múltiples procesadores en un solo chip. En muchos casos estos chips se pueden conectar entre sí para formar multiprocesadores de gran escala.

Programar de forma eficiente, comprar o vender multiprocesadores, por ejemplo, va a tener una importancia creciente en la actividad de un ingeniero informático, y esta asignatura va a ofrecer los conocimientos necesarios para desarrollar mejor estas actividades.

## 3.Evaluación

### 3.1.Tipo de pruebas y su valor sobre la nota final y criterios de evaluación para cada prueba

**El estudiante deberá demostrar que ha alcanzado los resultados de aprendizaje previstos mediante las siguientes actividades de evaluación**

La evaluación constará de tres partes:

1. Defensa de las prácticas de laboratorio (20 puntos).
2. Presentación de resultados sobre trabajo práctico no presencial (20 puntos).
3. Examen de teoría y problemas (60 puntos).

Para superar la asignatura el alumno deberá obtener al menos 50 puntos sobre el total y al menos 24 puntos sobre 60, es decir, un 4 sobre 10, en el examen. En el caso de no alcanzar un 4 sobre 10 en el examen, la nota del alumno en la convocatoria coincidirá con la obtenida en dicho examen.

La entrega de resultados de prácticas de laboratorio y trabajo práctico no presencial se realizará coincidiendo con las fechas programadas para examen en cada convocatoria.

## 4.Metodología, actividades de aprendizaje, programa y recursos

### 4.1.Presentación metodológica general

**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.Actividades de aprendizaje

**El programa que se ofrece al estudiante para ayudarle a lograr los resultados previstos comprende las siguientes actividades...**

- Clases magistrales.
- Clases de resolución de problemas.
- Prácticas de laboratorio asistidas.
- Trabajo práctico no presencial.
- Tutorías personalizadas sobre aspectos específicos
- Estudio y trabajo personal.

### 4.3.Programa

**Módulo I: Procesadores Vectoriales Segmentados: supercomputadores**

1. Introducción. El paralelismo
  - + Problemas científicos numéricos
  - + Rendimiento de una suma de vectores en procesadores escalares
    - Segmentado, supersegmentado y superescalar
  - + Versión vectorial de la suma de vectores.
2. Extensión vectorial de una arquitectura ld/st
  - + Arquitectura y Organización
  - + Repertorio básico de instrucciones vectoriales
  - + Organizaciones y Segmentación
    - Banco de registros vectoriales
    - Unidades funcionales (ALUs)
    - Memoria multibanco (acceso síncrono y concurrente)
  - + Cinco organizaciones de procesador vectorial y segmentación básica
  - + Medidas de rendimiento sin seccionar:  $R_n$ ,  $R?$ ,  $N?$ ,  $N_v$
  - + Organización ZV: un procesador vectorial segmentado sencillo
3. Dos aspectos de programación: vector length y vector Stride
  - + Vector length y seccionado (strip mining)
  - + Dos esquemas de generación de código seccionado. Ejemplo AXPY
  - + Rendimiento con seccionado:
    - ejemplo AXPY en ensamblador
    - $R_n$ ,  $R?$ ,  $N?$ ,  $N_v$
  - + Procesado de elementos no contiguos de un vector (stride)
4. Conflictos en el acceso a bancos de memoria
  - + Introducción. Esquema de almacenamiento. Propiedad fundamental.
  - + Sistemas Ajustados
  - + Sistemas Sobrados
5. Arquitectura DLXV: repertorio completo de instrucciones
6. Compilación vectorial = extracción automática de operaciones vectoriales
  - + Introducción
  - + Transformaciones previas que simplifican el análisis de dependencias
  - + Análisis de dependencias. Grafo de dependencias. Tests aproximados
  - + Optimizaciones independientes de la arquitectura: renombrar, expansión escalar, copia de vectores
  - + Vectorización. Procedimiento básico. Vectorización parcial vs. total: distribución e intercambio de bucles. Reducción
7. Consideraciones finales: Ley de Amdhal
8. Procesadores Vectoriales Comerciales
  - + Introducción
  - + Tabla de Supercomputadores
  - + Familia NEC: del SX-4 al SX-9 ACE (puede cambiar)
    - Concepto de ruta de datos particionada
  - + Extensiones vectoriales Intel: desde SSE hasta AVX512. Vigilancia Tecnológica.

## **Módulo II: Multiprocesadores Memoria Compartida**

1. Clasificación de computadores paralelos según M.J. Flynn
  - + SISD, SIMD y MIMD
2. Objetivos y problemas de las máquinas MIMD
3. Modelo sencillo de H.S. Stone para repartir procesos en procesadores
4. Multiprocesadores de Memoria compartida. Visión general
  - + Arquitectura-Programación: comunicación, sincronización, creación de procesos
  - + Organización: caches, red de interconexión, memoria principal
5. Red de interconexión
  - + Conflicto, degradación, topología, coste, conmutación circuitos o de paquetes, prestaciones, disponibilidad
  - + Topologías Dinámicas (redes indirectas): bus, multibus, crossbar, redes multietapa
  - + Topologías Estáticas (redes directas): estrella, anillo, malla, árbol, hipercubo
6. Mecanismos de Sincronización
  - + Repertorio de instrucciones: Test&Set, Fetch&Op, Linked Load
  - + Implementación. Combinación de peticiones
  - + Barreras
7. Compilación paralela
  - + extracción automática de tareas paralelas
8. El problema de la coherencia
  - + sistema, multiprocesador, cache multinivel, mas ejemplos
  - + escritura retardada e inmediata (animaciones)
9. El modelo de memoria

- + consistencia secuencial, pros y contras
- + una definición de coherencia
- 10. Protocolos de coherencia basados en difusión
  - + Invalidación. Difusión vs. envío selectivo
  - + ejemplos invalidación + CB + Bus: MSI, EI, Write Once, MESI
  - + Protocolos comerciales
- 11. Jerarquía de caches multinivel
- 12. Protocolos de coherencia basados en directorio
  - + requisitos hw y algunos transacciones de ejemplo
  - + protocolo sencillo de directorio
- 13. Ejemplos de chips actuales con más de un procesador (core)
  - + SUN, Intel, AMD, ARM, ...

#### 4.4. Planificación de las actividades de aprendizaje y calendario de fechas clave

##### Calendario de sesiones presenciales y presentación de trabajos

Se concretará cuando se apruebe el calendario académico.

##### Trabajo del estudiante

- clases magistrales: 30 horas
- clases de problemas: 15 horas
- clases de prácticas: 15 horas
- trabajo práctico no presencial: 12 horas
- estudio personal: 73 horas
- evaluación: 5 horas

La asignatura se compone de clases magistrales, clases de problemas, prácticas de laboratorio y trabajo práctico no presencial.

El calendario detallado de las diversas actividades a desarrollar se establecerá una vez que la Universidad haya aprobado el calendario académico del curso correspondiente. En cualquier caso, las fechas de las diferentes actividades serán anunciadas con suficiente antelación en la plataforma de apoyo a la enseñanza presencial Moodle 2 del Anillo Digital Docente (<http://add.unizar.es>) de la Universidad.

#### 4.5. Bibliografía y recursos recomendados

[BB: Bibliografía básica / BC: Bibliografía complementaria]

- [BB] John Hennessy, David A. Patterson. Computer architecture: a quantitative approach. 5th ed. San Francisco : Morgan Kaufmann, 2012.
- [BB] Culler, Singh, and Gupta. Parallel Computer Architecture : A Hardware-Software Approach, 1st ed. San Francisco : Morgan Kaufmann, cop. 1999
- [BB] Dally and Towles. Principles and practices of interconnection networks: Morgan Kaufmann, cop. 2004
- [BC] Patterson and Hennessy. Computer organization and desing: the hardware, software interface. 5th ed. Morgan Kaufmann. 2014
- [BC] Sorin, Hill, and Wood, A primer on memory consistency and cache coherence. Synthesis Lectures on Computer Architecture, Morgan & Claypool Publishers, 2011.
- [BB] Dubois, M. Annavaram, and Stenström. Parallel Computer Organization and Design, Cambridge University Press, 2012.