

PLAN DOCENTE DE LA ASIGNATURA

Curso académico: 2025/2026

Identificación y características de la asignatura					
Código	501287				
Denominación (español)	Arquitectura de Computadores				
Denominación (inglés)	Computer Architecture				
Titulaciones	Grado en Ingeniería Informática en Ingeniería de Computadores				
Centro	Escuela Politécnica				
Módulo	Módulo de Tecnología Específica en Ingeniería de Computadores				
Materia	Arquitecturas Paralelas y Distribuidas				
Carácter	Obligatorio	ECTS	6	Semestre	6
Profesorado					
Nombre		Despacho		Correo-e	
Miguel Ángel Vega Rodríguez		ARCO		mavega@unex.es	
Área de conocimiento	Arquitectura y Tecnología de Computadores				
Departamento	Tecnología de Computadores y Comunicaciones				
Profesor/a coordinador/a (si hay más de uno)	Miguel Ángel Vega Rodríguez				
Competencias					
COMPETENCIAS BÁSICAS					
CB1: Que los estudiantes hayan demostrado poseer y comprender conocimientos en un área de estudio que parte de la base de la educación secundaria general, y se suele encontrar a un nivel que, si bien se apoya en libros de texto avanzados, incluye también algunos aspectos que implican conocimientos procedentes de la vanguardia de su campo de estudio. CB2: Que los estudiantes sepan aplicar sus conocimientos a su trabajo o vocación de una forma profesional y posean las competencias que suelen demostrarse por medio de la elaboración y defensa de argumentos y la resolución de problemas dentro de su área de estudio. CB3: Que los estudiantes tengan la capacidad de reunir e interpretar datos relevantes (normalmente dentro de su área de estudio) para emitir juicios que incluyan una reflexión sobre temas relevantes de índole social, científica o ética. CB4: Que los estudiantes puedan transmitir información, ideas, problemas y soluciones a un público tanto especializado como no especializado. CB5: Que los estudiantes hayan desarrollado aquellas habilidades de aprendizaje necesarias para emprender estudios posteriores con un alto grado de autonomía.					

COMPETENCIAS GENERALES

CG04: Capacidad para definir, evaluar y seleccionar plataformas hardware y software para el desarrollo y la ejecución de sistemas, servicios y aplicaciones informáticas, de acuerdo con los conocimientos adquiridos para la tecnología específica de Ingeniería de Computadores.

CG06: Capacidad para concebir y desarrollar sistemas o arquitecturas informáticas centralizadas o distribuidas integrando hardware, software y redes, de acuerdo con los conocimientos adquiridos para la tecnología específica de Ingeniería de Computadores.

CG08: Conocimiento de las materias básicas y tecnologías, que capaciten para el aprendizaje y desarrollo de nuevos métodos y tecnologías, así como las que les doten de una gran versatilidad para adaptarse a nuevas situaciones.

COMPETENCIAS TRANSVERSALES

CT03: Capacidad para resolver problemas.

CT16: Capacidad para adaptarse a nuevas situaciones y cambios.

COMPETENCIAS TÉCNICAS (ESPECÍFICAS DE LA INGENIERÍA DE COMPUTADORES)

CIC03: Capacidad de analizar y evaluar arquitecturas de computadores, incluyendo plataformas paralelas y distribuidas, así como desarrollar y optimizar software para las mismas.

OBJETIVOS DE APRENDIZAJE (DOCENTES) DE LA ASIGNATURA

- Saber por qué son necesarias las arquitecturas paralelas de computadores, conocer la clasificación de las mismas y algunas de las métricas de rendimiento más habitualmente utilizadas en arquitecturas paralelas.
- Dominar los conceptos fundamentales sobre las arquitecturas paralelas multiprocesador y multinúcleo, así como otros tipos de arquitecturas multihilo. Conocer la jerarquía de la memoria en dichas arquitecturas paralelas, así como los principales conceptos sobre coherencia y consistencia de la memoria.
- Conocer los aspectos fundamentales sobre la programación paralela de multiprocesadores y procesadores multinúcleo, utilizando para ello OpenMP.
- Dominar los conceptos fundamentales sobre las arquitecturas paralelas multicomputador. Conocer las redes de interconexión usadas en las mismas, así como los principales conceptos sobre éstas: topologías, encaminamiento, etc.
- Conocer los aspectos fundamentales sobre la programación paralela de multicomputadores, utilizando para ello MPI.
- Saber los conceptos fundamentales sobre las arquitecturas superescalares, las etapas de segmentación típicas en ellas, las técnicas más habituales y algunos ejemplos de procesadores superescalares.
- Conocer las arquitecturas paralelas en datos más comunes (vectoriales, SIMD y GPGPUs), así como sus características y técnicas fundamentales.
- Conocer los aspectos fundamentales sobre la programación paralela en datos, utilizando para ello extensiones vectoriales/SIMD del repertorio de instrucciones, OpenMP y conceptos básicos de programación CUDA.

Contenidos
Descripción general del contenido: Arquitecturas multiprocesador. Arquitecturas multinúcleo y multihilo. Arquitecturas multicomputador. Arquitecturas superescalares. Arquitecturas vectoriales. La asignatura consta de dos partes: - Parte teórica. En ella se introducen los conceptos principales sobre diversas arquitecturas paralelas de computadores, así como las herramientas habituales para desarrollar y optimizar software para las mismas (aplicaciones paralelas). - Parte práctica. El estudiante deberá utilizar las herramientas/ideas introducidas en la parte teórica para desarrollar/diseñar distintas aplicaciones y arquitecturas paralelas, así como evaluar su rendimiento.
Temario
Denominación del tema 1: Introducción a las arquitecturas paralelas de computadores. Contenidos del tema 1: ¿Por qué son necesarias las arquitecturas paralelas? - Clasificación de las arquitecturas paralelas. - Métricas de rendimiento en arquitecturas paralelas. Descripción de las actividades prácticas del tema 1: Problemas de la Ley de Amdahl y la Ley de Gustafson. Denominación del tema 2: Multiprocesadores y procesadores multinúcleo. Contenidos del tema 2: - Arquitecturas paralelas multiprocesador y multinúcleo. - Conceptos básicos sobre multiprocesadores y procesadores multinúcleo. - Tipos de arquitecturas multihilo. - Jerarquías de memoria en multiprocesadores. - Jerarquías de memoria en procesadores multinúcleo. - Coherencia y consistencia de la memoria. - Protocolos de coherencia caché basados en vigilancia. - Protocolos de coherencia caché basados en directorio. - Programación paralela de multiprocesadores y procesadores multinúcleo. - Introducción a OpenMP. - Directivas OpenMP. - Cláusulas de alcance de datos en OpenMP. - Funciones de biblioteca en OpenMP. - Variables de entorno en OpenMP. Descripción de las actividades prácticas del tema 2: Simulador de memorias cachés en multiprocesadores y procesadores multinúcleo (SMPCache) y programación paralela con OpenMP.

Denominación del tema 3: Multicomputadores.

Contenidos del tema 3:

- 3.1 Arquitecturas paralelas multicomputador.
 - 3.1.1 Conceptos básicos sobre multicomputadores.
 - 3.1.2 Redes de interconexión en multicomputadores.
 - 3.1.3 Topologías de redes de interconexión.
 - 3.1.4 Técnicas de conmutación en redes de interconexión.
 - 3.1.5 Encaminamiento en redes de interconexión.
- 3.2 Programación paralela de multicomputadores.
 - 3.2.1 Introducción a MPI.
 - 3.2.2 Funciones de biblioteca básicas en MPI.
 - 3.2.3 Tipos de datos en MPI.
 - 3.2.4 Comunicaciones punto a punto en MPI.
 - 3.2.5 Comunicaciones colectivas en MPI.
 - 3.2.6 Programación híbrida MPI/OpenMP.

Descripción de las actividades prácticas del tema 3: Simulador de redes de interconexión en multicomputadores (SimuRed) y programación paralela con MPI.

Denominación del tema 4: Arquitecturas superescalares.

Contenidos del tema 4:

- 4.1 Conceptos importantes.
- 4.2 Arquitectura de un procesador superescalar.
- 4.3 Etapas de segmentación en un procesador superescalar.
- 4.4 Tipos de procesadores superescalares.
- 4.5 Técnicas usadas en procesadores superescalares.
- 4.6 Ejemplo de procesador superescalar.

Descripción de las actividades prácticas del tema 4: Simulador de procesadores superescalares (SimpleScalar).

Denominación del tema 5: Arquitecturas vectoriales, SIMD y GPGPU.

Contenidos del tema 5:

- 5.1 Arquitecturas paralelas en datos.
 - 5.1.1 Introducción a las arquitecturas paralelas en datos.
 - 5.1.2 Arquitecturas vectoriales.
 - 5.1.3 Arquitecturas GPGPU: Unidades de procesamiento gráfico para propósito general.
- 5.2 Programación paralela en datos.
 - 5.2.1 SIMD: Extensiones vectoriales del repertorio de instrucciones.
 - 5.2.2 SIMD utilizando OpenMP.
 - 5.2.3 CUDA: Programación paralela de GPGPUs.

Descripción de las actividades prácticas del tema 5: Programación paralela con extensiones vectoriales (SIMD) del repertorio de instrucciones.

Actividades formativas								
Horas de trabajo del alumno/a por tema		Horas Gran grupo	Actividades prácticas				Actividad de seguimiento	No presencial
Tema	Total	GG	CH	L	O	S	TP	EP
1	9	3	0	1	0	0	0	5
2	36	12	0	4	0	0	1	19
3	36	12	0	4	0	0	1	19
4	13	4	0	1	0	0	0	8
5	31	10	0	3	0	0	1	17
Evaluación	25	4	0	2	0	0	0	19
TOTAL	150	45	0	15	0	0	3	87

GG: Grupo Grande (85 estudiantes).

CH: Actividades de prácticas clínicas hospitalarias (7 estudiantes).

L: Actividades de laboratorio o prácticas de campo (15 estudiantes).

O: Actividades en sala de ordenadores o laboratorio de idiomas (20 estudiantes).

S: Actividades de seminario o de problemas en clase (40 estudiantes).

TP: Tutorías Programadas (seguimiento docente, tipo tutorías ECTS).

EP: Estudio personal, trabajos individuales o en grupo, y lectura de bibliografía.

Metodologías docentes

Clases teórico-prácticas en aula

Se emplearán distintas actividades en el aula, dirigidas al grupo completo o a pequeños grupos. Principalmente, se realizarán clases expositivas para el desarrollo de los contenidos fundamentales de la asignatura y, para conseguir la participación activa de los estudiantes, se llevarán a cabo actividades breves individuales o en grupo que permitan aplicar los conceptos expuestos y resolver problemas. Se propondrán actividades encaminadas a la aplicación de los conocimientos en la resolución de problemas propios del ámbito de la Arquitectura de Computadores.

Sesiones de laboratorio

Se realizarán actividades prácticas, sesiones de laboratorio guiadas, resolución de problemas, etc. en grupos, bajo la dirección del profesor. Se podrán incluir actividades previas y posteriores a las sesiones de laboratorio que ayuden a conseguir los objetivos propuestos. Las actividades propuestas se aproximarán, en la medida de lo posible, a las actividades reales a las que se enfrenta un Ingeniero de Computadores en su desarrollo profesional.

Tutorías programadas

En estas tutorías programadas individuales o en grupos pequeños se realizará un seguimiento más individualizado del estudiante, con actividades de formación y orientación. Estas tutorías servirán para guiar al estudiante en la toma de decisiones, evaluar los trabajos realizados y fomentar las actitudes propias de los profesionales del ámbito de la Informática.

Trabajo y estudio individual no presencial

Realización de actividades, trabajos y estudio por parte del estudiante, de manera autónoma, individualmente o en grupo. Se fomentarán las tareas no presenciales similares a las que realiza un Ingeniero en su ámbito profesional.

Resultados de aprendizaje

- Domina los conceptos fundamentales sobre diversas arquitecturas de computadores, como arquitecturas superescalares, multinúcleo, vectoriales, multihilo, multiprocesador y multicomputador, pudiendo evaluar de manera efectiva y eficiente los distintos sistemas hardware en términos de calidad, aportando posibles alternativas a un determinado problema.
- Reconoce la estructura de un problema, datos de entrada, incógnitas, magnitudes, condiciones iniciales, así como los pasos de su resolución.
- Extrae del problema las soluciones triviales, reconoce la multiplicidad de soluciones, etc.
- Sabe elegir con fundamento los métodos y medios más adecuados para resolver un problema.
- Identifica las situaciones de cambio.
- Elabora las estrategias para abordar la problemática implicada por la nueva situación.
- Aplica las estrategias para adaptarse a la nueva situación.

Sistemas de evaluación

La asignatura contempla 2 modalidades de evaluación: **evaluación continua** y **evaluación global**. Los estudiantes podrán elegir entre ambas evaluaciones (continua o global) de acuerdo a la Normativa de Evaluación de la UEx, en los plazos establecidos por la misma.

La asignatura está dividida en dos partes: teoría y prácticas; que deben aprobarse por separado. Ambas partes se califican de 0 a 10. Los estudiantes que aprueben una de las partes (teoría o prácticas) se les guarda durante el resto de convocatorias del mismo curso académico.

Criterios de evaluación:

- Demostrar la adquisición, comprensión y dominio de los principales conceptos de la asignatura.
- Resolver problemas aplicando conocimientos teóricos y basándose en resultados experimentales, demostrando que se saben elegir con fundamento los métodos y medios más adecuados para resolver cada problema.
- Desarrollar y comprender adecuadamente las prácticas de la asignatura, demostrando que se saben aplicar las estrategias necesarias para adaptarse a cambios o nuevas situaciones prácticas.
- La nota final de la asignatura (NF) se calculará realizando la suma ponderada siguiente, donde NT es la calificación obtenida en la parte teórica y NP es la calificación obtenida en la parte práctica de la asignatura:

$$NF = NT*0,75 + NP*0,25$$

- Para poder realizar dicha suma ponderada se ha de obtener en cada parte, por separado, una nota igual o superior a 4,5 puntos. Caso de no llegar a esa nota, la asignatura quedará suspensa en esa convocatoria.

- Para aprobar la asignatura la nota final (tras hacer la suma ponderada de ambas partes) debe ser igual o superior a 5 puntos.

Actividades e instrumentos de evaluación:

- *Prácticas (Laboratorio):*
 - **Evaluación continua:** para cada práctica semanal se evaluará la calidad de la documentación técnica entregada, además de la asistencia a dicha práctica y la realización de la misma.
 - **Evaluación global:** los **estudiantes que no hayan superado o no deseen seguir la evaluación continua** tendrán derecho a presentarse al **examen final de prácticas (prueba final alternativa de carácter global, evaluación global)** en las convocatorias oficiales correspondientes. En este caso, para aprobar las prácticas hay que realizar un examen práctico, demostrando el conocimiento y dominio de los conceptos tratados en las prácticas de la asignatura, así como la capacidad de adaptación a cambios o nuevas situaciones prácticas.
- *Teoría (Grupo Grande):*
 - **Evaluación continua:** de los 10 puntos de teoría, 2 puntos se evaluarán mediante preguntas de teoría semanales y los 8 puntos restantes mediante un examen de teoría (cuya nota mínima debe ser un 3,5), es decir, una prueba de desarrollo escrito y resolución de problemas.
 - **Evaluación global:** para aprobar la teoría hay que realizar un examen de teoría en las convocatorias oficiales correspondientes, demostrando la adquisición, comprensión y dominio de los distintos conceptos de la asignatura, así como la correcta aplicación de estos conceptos, de forma creativa y segura, a supuestos de carácter práctico (problemas).

Bibliografía (básica y complementaria)

- Apuntes, diapositivas y enunciados de prácticas y problemas facilitados por el profesor.
- Referencias bibliográficas:
 - Arquitectura de Computadores. Thomson-Paraninfo. Julio Ortega, Mancia Anguita y Alberto Prieto.
 - Parallel Computer Architecture: A Hardware/Software Approach. Morgan Kaufmann. David Culler, Jaswinder P. Singh, and Anoop Gupta.
 - Computer Architecture: A Quantitative Approach. 6th edition, Morgan Kaufmann. John L. Hennessy and David A. Patterson.
 - Computer Organization & Architecture: Designing for Performance. 11th edition, Prentice Hall. William Stallings.
 - Advanced Computer Architecture: A Design Space Approach. Addison-Wesley. Dezso Sima, Terence Fountain, and Peter Kacsuk.
 - Arquitectura de Computadores: Fundamentos de los Procesadores Superescalares. McGraw-Hill. John Paul Shen y Mikko H. Lipasti.
 - The OpenMP Common Core: Making OpenMP Simple Again. MIT Press. Timothy G. Mattson, Yun (Helen) He, and Alice E. Koniges.
 - Using MPI: Portable Parallel Programming with the Message Passing Interface. 3rd edition, MIT Press. William Gropp, Ewing Lusk, and Anthony Skjellum.
 - Programming Massively Parallel Processors: A Hands-on Approach. 4th edition, Morgan Kaufmann. Wen-Mei W. Hwu, David B. Kirk, and Izzat El Hajj.

Otros recursos y materiales docentes complementarios

- Aula virtual de la asignatura (descarga de materiales, foros, noticias, etc.): <https://campusvirtual.unex.es/zonaunex/avunex/course/view.php?id=17213>
- Sitio web del simulador SMPCache: <http://arco.unex.es/smpcache>
- Sitio web de OpenMP: <https://openmp.org>
- Sitio web del simulador SimuRed: <https://simured.uv.es>
- Sitio web de MPI: <https://www.mpi-forum.org>
- Sitio web de MPICH: <https://www.mpich.org>
- Sitio web de Open MPI: <https://www.open-mpi.org>
- Sitio web del simulador SimpleScalar: <https://github.com/onesuper/SimpleScalar>
- Sitio web de extensiones SIMD del repertorio de instrucciones: <https://software.intel.com/sites/landingpage/IntrinsicsGuide/>
- Sitio web de CUDA: <https://developer.nvidia.com/cuda-zone>
- TOP500 Supercomputer List: <https://www.top500.org>
- Green500 Energy-Efficient Supercomputer List: <https://www.top500.org/lists/green500/>
- Sitios web de los distintos libros recomendados en la asignatura.