

Curso Académico: (2024 / 2025)

Fecha de revisión: 08-05-2024

Departamento asignado a la asignatura: Departamento de Informática

Coordinador/a: GARCIA SANCHEZ, JOSE DANIEL

Tipo: Obligatoria Créditos ECTS : 6.0

Curso : 3 Cuatrimestre : 1

Rama de Conocimiento: Ingeniería y Arquitectura

REQUISITOS (ASIGNATURAS O MATERIAS CUYO CONOCIMIENTO SE PRESUPONE)

- + Programación (Curso 1 / Cuatrimestre 1)
- + Estructura de Computadores (Curso 2 / Cuatrimestre 1)
- + Sistemas Operativos (Curso 2 / Cuatrimestre 2)

COMPETENCIAS Y RESULTADOS DEL APRENDIZAJE

¿ Distinguir los distintos elementos de la organización y jerarquía de memoria de un computador y comprender cómo afectan al rendimiento de un procesador pudiendo aprovechar dicho conocimiento para optimizar algoritmos escritos en lenguajes de alto nivel

¿ Conocer el concepto de arquitectura multiprocesador y de procesador multi-núcleo y poder desarrollar y optimizar software para dichas arquitecturas.

¿ Conocer y aplicar los principios fundamentales y las técnicas básicas de la programación paralela y la programación concurrente.

OBJETIVOS

El objetivo principal de esta asignatura es que el estudiante conozca los conceptos básicos sobre la arquitectura de un computador y el impacto que estos tienen sobre el rendimiento de las aplicaciones y los sistemas informáticos.

Para alcanzar este objetivo:

- + Se profundiza en aspectos avanzados de la arquitectura del computador como la optimizaciones de la jerarquía de memoria, el paralelismo a nivel de instrucción, o el diseño de multiprocesadores.
- + Se refuerzan conceptos básicos de programación concurrente y su interrelación con la arquitectura del computador.
- + Se introducen conceptos de programación paralela.
- + Se fomenta una preocupación constante por el rendimiento y el consumo energético de las aplicaciones.

DESCRIPCIÓN DE CONTENIDOS: PROGRAMA

- Fundamentos del diseño de computadores.
 - 1.1. Introducción.
 - 1.2. Perspectiva histórica.
 - 1.3. Clasificación de computadores.
 - 1.4. Paralelismo.
 - 1.5. Arquitectura del Computador.
- Evaluación del rendimiento de sistemas informáticos.
 - 2.1. Clasificación de arquitecturas y evaluación.
 - 2.2. Fiabilidad de Sistemas Informáticos.
- Jerarquía de memoria.
 - 3.1. Optimizaciones de memoria caché.

- 3.2. Optimizaciones avanzadas de memoria caché.
- 4. Paralelismo a nivel de instrucción.
 - 4.1. Introducción al paralelismo a nivel de instrucción.
 - 4.2. Explotación del paralelismo a nivel de instrucción
- 5. Multiprocesadores.
 - 5.1. Arquitecturas de memoria compartida simétrica.
 - 5.2. Memoria compartida distribuida.
 - 5.3. Sincronización en memoria compartida.
 - 5.4. Modelos de consistencia de memoria.
- 6. Modelos de programación paralela y concurrente.
 - 6.1. Introducción a la programación paralela: OpenMP.
 - 6.2. Modelo de programación concurrente: Hilos en ISO C++.
 - 6.3. Modelos de consistencia de memoria en C++.

ACTIVIDADES FORMATIVAS, METODOLOGÍA A UTILIZAR Y RÉGIMEN DE TUTORÍAS

** ACTIVIDADES FORMATIVAS **

CLASES TEÓRICO-PRÁCTICAS [38 horas con 100% de presencialidad, 1.52 ECTS].

En ellas se presentarán los conocimientos que deben adquirir los alumnos. Estos recibirán las notas de clase y tendrán textos básicos de referencia para facilitar el seguimiento de las clases y el desarrollo del trabajo posterior. Se resolverán ejercicios por parte del alumno que le servirá de autoevaluación y para adquirir las capacidades necesarias. Clases de problemas, en las que se desarrollen y discutan los problemas que se proponen a los alumnos.

TALLERES Y/O PRÁCTICAS DE LABORATORIO [10 horas con 100% de presencialidad, 0.40 ECTS].

TUTORÍAS [28 horas con 25% de presencialidad, 1.12 ECTS].

Asistencia individualizada (tutorías individuales) o en grupo (tutorías colectivas) a los estudiantes por parte del profesor.

TRABAJO INDIVIDUAL O EN GRUPO DEL ESTUDIANTE [70 horas con 0% de presencialidad, 2.8 ECTS].

EXAMEN FINAL [4 horas 100% de presencialidad, 0.16 ECTS]

Se valorarán de forma global los conocimientos, destrezas y capacidades adquiridas a lo largo del curso.

** METODOLOGÍAS **

Seminarios y lecciones magistrales con apoyo de medios informáticos y audiovisuales.

Aprendizaje práctico basado en casos y problemas y resolución de ejercicios.

Trabajo individual y en grupo o cooperativo con opción a presentación oral o escrita.

Tutorías individuales y en grupo para resolución de dudas y consultas sobre la materia.

Prácticas y actividades dirigidas de laboratorios

SISTEMA DE EVALUACIÓN

Peso porcentual del Examen Final: 40

Peso porcentual del resto de la evaluación: 60

Evaluaciones de conocimientos durante el curso: 24%

Prácticas realizadas individualmente o en grupo: 36%

Examen final: 40%

CONVOCATORIA ORDINARIA

Nota mínima en el examen: 3.5 puntos sobre 10.

Peso porcentual del Examen Final:	40
Peso porcentual del resto de la evaluación:	60

Nota mínima en proyectos: 2 sobre 10.

La nota final se incrementará en 1 punto a aquellos alumnos que realicen todas las pruebas de evaluación continua, obtengan más un 7 en la evaluación continua y al menos 6 puntos en el examen final.

Para aquellos alumnos que no hayan completado el proceso de evaluación continua, o no hayan superado las notas mínimas de prácticas, el examen final tendrá un valor del 60% de la calificación total de la asignatura. Por tanto, para poder aprobar, el alumno deberá obtener una calificación superior a 8,33 sobre 10 en este examen.

BIBLIOGRAFÍA BÁSICA

- Hennessy, JL y Patterson, DA. Computer Architecture: A Quantitative Approach. 6th Edition., Morgan Kaufmann,, 2017

BIBLIOGRAFÍA COMPLEMENTARIA

- David A. Patterson, John Hennessy Computer Organization and Design MIPS Edition: The Hardware/Software Interface, Morgan Kaufmann, 2020

- Timothy G. Mattson, Yun (Helen) He and Alice E. Koniges The OpenMP Common Core: Making OpenMP Simple Again, MIT Press, 2019

- Williams, A. C++ Concurrency in Action. Practical Multithreading. 2nd Edition, Manning., 2018