

PROGRAMA ALGORITMIA Y COMPLEJIDAD

Código:	CS023
Clases:	2 semanales (lunes y miércoles)
Sección:	A
Catedrático:	Mario A. Castillo
Correo electrónico:	macastillom@ufm.edu
UMAs:	3

Descripción del curso:

Este curso es una introducción a la solución de problemas de programación, y al análisis y diseño de algoritmos. En él se pretende, no solamente explicar los algoritmos, sino también construir las herramientas teóricas necesarias para entender, analizar y diseñar.

El curso comprende diversos temas, entre ellos: recursión, estrategias *divide & conquer* y análisis de complejidad. Se estudian también algoritmos de ordenamiento, estructuras de datos y teoría de grafos. El curso finaliza con una introducción al estudio de optimización combinatoria.

Se fomenta también el uso de herramientas tecnológicas para la resolución de problemas. El desarrollo de las habilidades para resolver sistemas se lleva a cabo mediante lecturas y se complementa mediante la discusión de ejercicios en el aula.

Objetivo General:

- Introducir al estudiante a los principios fundamentales de los algoritmos, el análisis de su complejidad y sus aplicaciones en la solución de problemas.

Objetivos Específicos:

- Aprender a analizar algoritmos iterativos y recursivos.
- Desarrollar fluidez con notación *big-O*.
- Aprender a elegir e implementar algoritmos eficientes para problemas numéricos, combinatorios y geométricos.
- Aprender conceptos básicos y terminología en computación y complejidad computacional.

Metodología:

Se impartirá clase teórica 2 días por semana según la calendarización. Los exámenes parciales serán realizados en las fechas indicadas más adelante. El alumno deberá desarrollar ejercicios en clase, exámenes cortos, exámenes parciales y aplicaciones adicionales. Los exámenes cortos podrán ser programados o sorpresivos.

Habrán algunos temas que el catedrático deje que estudien por su cuenta, ese material también se evaluará. Los exámenes podrán contener preguntas de concepto y problemas de desarrollo.

Evaluación:

La integración de la nota final será de la siguiente manera:

PROCEDIMIENTO	PONDERACIÓN
Examen Parcial 1	15%
Examen Parcial 2	15%
Exámenes Cortos	15%
Aplicaciones	10%
Ejercicios en clase	5%
TOTAL ZONA	60%
Examen Final	40%
NOTA TOTAL	100%

Contenido (estimado) del curso:

1. Recursión	Inducción matemática Recursión Recursión con memoria <i>Divide & conquer</i> Búsqueda exhaustiva
2. Análisis de Complejidad	Técnicas de conteo y funciones Notación <i>Big-O</i> El conjunto de números naturales \mathbb{N} Definición de conjunto discreto
3. <i>Sorting Algorithms I</i>	<i>Selection sort</i> <i>Insertion sort</i> <i>Bubble sort</i> <i>Merge sort</i>
4. Estructuras de Datos	Pilas, colas y listas Árboles binarios Montículos (<i>Heaps</i>) Grafos
5. <i>Sorting Algorithms II</i>	Ordenamiento por árboles binarios de búsqueda Ordenamiento por <i>heaps</i> <i>Count sort</i> <i>Bucket sort</i>
6. Optimización combinatoria	Espacios de búsqueda Programación dinámica Programación dinámica en cortes

Fechas importantes:

- Examen Parcial 1 – semana del 14 al 18 de septiembre de 2020
- Examen Parcial 2 – semana del 02 al 06 de noviembre de 2020
- Examen Final – semana del 23 al 27 de noviembre de 2020

Nota: No habrá reposición de exámenes cortos ni exámenes parciales.

Bibliografía:

El curso no cuenta con un «*libro de texto*». Sin embargo, si el estudiante desea profundizar en alguno de los temas presentados, puede consultar las siguientes referencias bibliográficas.

Brassard, G. *Fundamentos de Algoritmia*.

Cormen, T. *Introduction to Algorithms*. 3rd Ed.

Dasgupta, S. *Algorithms*.

Levitin, A. *Introduction to the Design and Analysis of Algorithms*. 3rd Ed.