



Ciencias de la Computación (Curso 2021-2022)

Asignatura optativa de 4º curso del Grado en **Matemáticas**, adscrita al Departamento de Ciencias de la Computación e Inteligencia Artificial. Tiene 6 créditos. Se imparte en el primer cuatrimestre.

Profesor: F. Félix Lara Martín (fflara@us.es)

Despacho H1.45 (Escuela superior de Ingeniería Informática, modulo H, primera planta).

Página web: <http://www.cs.us.es/~felix>

Programa

TEMA I: Algoritmos y funciones computables.

- Concepto intuitivo de algoritmo.
- Modelos de computación: El lenguaje **GOTO**

TEMA II: Equivalencia entre modelos de computación.

- Máquinas de Turing y programas de Post–Turing.
- Procesos, gramáticas y máquinas de Turing.

TEMA III: Funciones recursivas.

- Funciones y predicados primitivos recursivos.
- La clase de las funciones recursivas. Tesis de Church–Turing.

TEMA IV: Programas universales y forma normal.

- Codificación de programas: Aritmetización.
- Programa universal y teorema de la forma normal.

TEMA V: Conjuntos recursivamente enumerables.

- El problema de la parada.
- Conjuntos recursivamente enumerables.
- Incompletitud e indecidibilidad de sistemas formales.

TEMA VI: El teorema de recursión.

- Transformaciones automáticas de programas: Teorema s - n - m .
- El teorema de recursión.

TEMA VII: Reducibilidad.

- El teorema de Rice. m -reducibilidad.
- Computabilidad relativa. Turing-reducibilidad.

TEMA VIII: Autómatas, lenguajes y gramáticas.

- Autómatas finitos y lenguajes regulares.
- Lenguajes independientes del contexto.

TEMA IX: Complejidad computacional.

- Computación con recursos acotados. Las clases **P** y **NP**.
- Reducibilidad polinomial. Problemas **NP**-completos.

Bibliografía

CUTLAND, N.: Computability: Introduction to Recursive Function Theory. Cambridge University Press (1980).

DAVIS, M.; SIGAL, R.; WEYUKER, E.: Computability, complexity and languages. Academic Press, 1994.

GOLDREICH, O.: P, NP, and NP-Completeness The Basics of Complexity Theory (2010). <http://www.wisdom.weizmann.ac.il/~oded/bc-drafts.html>

HOPCROFT, J.E.; MOTWANI, R.; ULLMAN, J.D.: Introducción a la Teoría de Autómatas, Lenguajes y Computación, 2ed. Pearson/Addison Wesley (2002).

JONES, N.: Computability and Complexity from a Programming Perspective. The MIT Press (1997). <http://www.diku.dk/hjemmesider/ansatte/neil/Comp2book.html>

LEWIS, H. R.; PAPADIMITRIOU, C.: Elements of the theory of computation. Prentice-Hall, 1981.

ODIFREDDI, P.: Classical Recursion Theory, Vol. I. North-Holland, 1990.

PRETORIUS L.; ZUCKER, J.I. Introduction to computability theory, South African Computer Journal, 9:3-30 (1990). <http://www.cas.mcmaster.ca/~zucker/Pubs/WOFACS/text.ps>

SIPSER, M.: Introduction to the Theory of Computation. Thomson, Course Technology. Boston, 2016.

Evaluación: La asignatura se evaluará de dos formas (no excluyentes entre sí):

- Realizando a lo largo del cuatrimestre dos pruebas escritas de carácter obligatorio (la primera al acabar el tema V y la segunda durante la última semana de clase). La superación de estas pruebas permitirá aprobar la asignatura sin necesidad de realizar el examen final.
 - Esta evaluación puede completarse de manera opcional con la realización de trabajos/ejercicios propuestos durante el desarrollo del curso. Podrán obtenerse así hasta 2 puntos, que se sumarán a la nota obtenida en las pruebas escritas, siempre que ésta no sea menor que 4 puntos
- Mediante la superación de un examen final en las fechas previstas.