

UNIVERSIDAD DE PUERTO RICO EN HUMACAO  
DEPARTAMENTO DE MATEMÁTICAS

A. Encabezado	Universidad de Puerto Rico en Humacao
B. Nombre del curso	Teoría de Computación
C. Codificación	COMP4099
D. Cantidad de horas/créditos	Tres (3) horas contacto <sup>1</sup> / Tres (3) créditos
E. Requisitos o correquisitos y otros requerimientos	MATE3050, MATE4031
F. Descripción del curso	

El curso es un estudio de modelos matemáticos de computación. Estos modelos incluyen los autómatos finitos, las expresiones regulares, las gramáticas no-contextuales, los autómatos de estibamiento (pushdown autómatos) y las máquinas de Turing. Para estos modelos se analiza su poder relativo y sus propiedades de clausura, decidibilidad y complejidad.

G. Objetivos de aprendizaje

Objetivos Generales:

Al finalizar el semestre el estudiante podrá:

1. Formular un modelo de computación que exhiba las características esenciales de una computadora digital. Por medio de este modelo se estudiarán las capacidades y limitaciones de las computadoras y de los algoritmos como métodos para resolver problemas.
2. Proveer las nociones teóricas necesarias para la construcción de herramientas usadas en la definición de lenguajes de programación y en la construcción de compiladores.

Objetivos Específicos:

Al finalizar el semestre el estudiante podrá:

1. Definir lenguajes formales y mostrar su relación con problemas típicos de la programación de computadoras.
2. Representar algunos sistemas mediante diagramas de transición.
3. Definir, mostrar y construir autómatos deterministas finitos.
4. Definir, mostrar y construir autómatos no-deterministas finitos.
5. Mostrar aplicaciones de los autómatos finitos a problemas de programación, de comunicación y de

---

<sup>1</sup> Una hora contacto equivale a cincuenta (50) minutos.

otras clases.

6. Definir, reconocer y construir expresiones regulares.
7. Establecer relación entre autómatos finitos y expresiones regulares.
8. Definir gramáticas no-contextuales.
9. Simplificar gramáticas no-contextuales.
10. Obtener formas normales de gramáticas no-contextuales.
11. Definir autómatos deterministas de estibamiento.
12. Definir autómatos no-deterministas de estibamiento.
13. Definir, mostrar y construir máquinas de Turing.
14. Programar máquinas de Turing.
15. Hacer extensiones a la máquina de Turing.
16. Definir la máquina universal de Turing.
17. Analizar el poder relativo de los lenguajes finitos, regulares y no-contextuales.
18. Estudiar problemas de decidibilidad para los autómatos finitos, las gramáticas no-contextuales y máquinas deterministas de Turing.
19. Determinar las relaciones de inclusión, equivalencia e intersección para los modelos computacionales estudiados anteriormente en el curso.
20. Definir la clase de problemas polinomial no-determinista (NP) y mostrar que algunos problemas importantes están en esa clase.

#### H. Bosquejo de contenido y distribución del tiempo

Unidad 0 - Introducción Qué es teoría de computación	(1 hora)
Unidad 1 - Lenguajes Alfabetos, palabras y lenguajes	(3 horas)
Unidad 2 - Autómatos Finitos y Lenguajes Regulares Autómatos finitos deterministas Autómatos no-deterministas Otros ejemplos de autómatos finitos Expresiones y lenguajes regulares Relación entre lenguajes regulares y autómatos finitos	(12 horas)

## Examen I

Unidad 3 - Gramáticas No-Contextuales y Automatos de Estiba (8 horas)  
Definición y propiedades de las gramáticas no contextuales  
Gramáticas regulares y lineales  
Automatos deterministas de estiba  
Automatos no-deterministas de estiba  
Lenguajes no-contextuales

Unidad 4 - Máquinas de Turing y Funciones Computables (12 horas)  
Definición  
Programación  
Versiones enriquecidas de la máquina de Turing  
Máquina de Turing universal  
Funciones computables y funciones recursivas  
Examen II

Unidad 5 - Decidibilidad y Completez NP (6 horas)  
Decidibilidad y pertenencia  
Problema de correspondencia de Post  
La clase NP  
Algunos problemas importantes de la clase NP  
Examen Final

La suma de horas sugeridas es 42. Las 3 horas restantes deben ser dedicadas a la administración de evaluaciones parciales.

### I. Estrategias Instruccionales

Para lograr los objetivos del curso se realizarán las siguientes actividades instruccionales: conferencias, demostraciones, trabajos en grupo, estudios independientes, discusión de asignaciones.

### J. Recursos mínimos disponibles o requeridos

Este curso no requiere el uso del laboratorio de computadoras.

### K. Técnicas de evaluación

Se darán un mínimo de tres evaluaciones entre las cuales se incluirán exámenes parciales, un examen final y asignaciones.

### L. Acomodo razonable

Los estudiantes que requieran acomodo razonable deben visitar la Oficina de Servicios para la Población con Impedimentos (SERPI) y comunicarse con el profesor al inicio del semestre para planificar el acomodo necesario conforme a las recomendaciones de SERPI.

## M. Integridad académica

El Artículo 6.2 del Reglamento General de Estudiantes de la UPR (Certificación Número. 13, 2009-2010 de la Junta de Síndicos) establece que *“la deshonestidad académica incluye, pero no se limita a: acciones fraudulentas, la obtención de notas o grados académicos valiéndose de falsas o fraudulentas simulaciones, copiar total o parcialmente la labor académica de otra persona, plagiar total o parcialmente el trabajo de otra persona, copiar total o parcialmente las respuestas de otra persona a las preguntas de un examen, haciendo o consiguiendo que otro tome en su nombre cualquier prueba o examen oral o escrito, así como la ayuda o facilitación para que otra persona incurra en la referida conducta”*.

Cualquiera de estas acciones estará sujeta a sanciones disciplinarias en conformidad con el procedimiento disciplinario establecido en dicho reglamento.

## N Sistema de calificación

Se adjudicará la nota A, B, C, D, ó F según el nivel de competencia demostrado en las evaluaciones.

**El profesor o profesora informará, las primeras semanas de clases (y en la guía de estudiante), la curva a usarse para asignar las calificaciones.**

## O. Bibliografía

1. Cutland, N., (1997, reprint), *Computability: An Introduction to Recursive Function Theory* Cambridge University Press.
2. Davis, M., (1994), *Computability, Complexity, and Languages: Fundamentals of Theoretical Computer Science*, Academic Press, (2da. Ed.).
3. Denning, P. J., (1978), *Machines, languages and Computation*, Prentice-Hall, Englewood Cliffs.
4. Gurari, E., (1999), *An Introduction to the Theory of Computation*, Computer Science Press.
5. Hopcroft, J.E., Motwani, R. & Ullman, Jeffrey D., (2000), *Introduction to Automata Theory, Languages and Computation*, Addison-Wesley Publishing, (2da. Ed.).
6. Hopcroft, J. E. & Ullman, J. D., (1979), *Introduction to Automata Theory, Languages, and Computation*, Addison-Wesley
7. Kain, R. Y., (1981), *Automata Theory: Machines and Languages*, Krieger.
8. Kohavi, Z., (1978), *Switching and Finite Automata Theory*, McGraw-Hill.
9. Kozen, D. C., (1999), *Automata and Computability*, Springer-Verlag.
10. Lewis, H. & Papadimitriou, C. H., (1997), *Elements of the Theory of Computation*, Prentice-Hall, Englewood Cliffs, (2da. Ed).
11. Linz, P., (1997), *Introduction to Formal Languages and Automata*, D. C. Heath & Company, (2da Ed.).

12. Martin, J.C., (2003), Introduction to Languages and the Theory of Computation, McGraw-Hill, (3ra. Ed.).
13. Sipser, M., (2005), Introduction to the Theory of Computation, Thomson Course Technology. (2da Ed.).
14. The Mathematical Association of America, (1982), Studies in Computer Science.
15. Wood, D., (1987), Theory of Computation, Harper & Row.

#### Responsables de las revisiones

Rev/Elio Ramos/Comité de Currículo, sep 2016  
Rev/MLV/jsm/PD/ProntuarioCOMP4099/25-jun.-07  
Rev/Bibliografía/Profa.B.Santiago-Figueroa/Oct.05  
Rev/ComitéCurrículo/Nov.05  
Aprob/DM/15-nov.05  
Rev/E.Pérez/Nov.02  
Rev/ComitéCurrículo/Dic.02