



UNIVERSIDAD NACIONAL DE LA PLATA  
FACULTAD DE INFORMÁTICA

---

## COMPUTABILIDAD Y COMPLEJIDAD

Año 2013

Carrera: *Licenciatura en Informática*  
*Plan 2003-2007*  
*Licenciatura en Sistemas*  
*Plan 2003-2007*

Año: 3°

Duración: *Semestral*

Profesor: *Fernando G. Tinetti*

---

### FUNDAMENTACION:

Se presentan las ideas básicas de tres temas fundamentales para el desarrollo profesional: 1) el modelo de procesamiento a partir del cual se construyen las computadoras, 2) La forma de analizar la dificultad de la resolución de clases de problemas, y 3) La metodología de análisis de algoritmos para comparación de comportamiento.

### OBJETIVOS GENERALES:

Introducir los conceptos de Computabilidad y Complejidad, aplicados al análisis de algoritmos. Resolver "casos" de análisis clásicos, relacionando la eficiencia de las soluciones. Introducir la formalización de las notaciones para estudiar tiempo y espacio en los algoritmos.

### CONTENIDOS MINIMOS:

- Análisis de algoritmos: análisis asintótico, comportamiento en el mejor caso, caso promedio, peor caso.
- Notación  $O()$ . Balance entre tiempo y espacio en los algoritmos.
- Problemas computables y no computables.
- Problemas de la detención.
- Problemas tratables e intratables.
- Funciones recursivas.

### Programa

---



Se trata de una introducción a las fronteras de la algoritmia. Comienza con elementos de teoría de la computación (modelos de computación y computabilidad). Continúa con el estudio de la complejidad de los problemas tratables. Termina con elementos de la clase intratable de los problemas de la complejidad computacional. Los trabajos prácticos incluyen problemas analíticos y de prueba de algoritmos.

1. **Computabilidad**
  - a. Programas y Problemas.
  - b. Funciones Recursivas y Parciales.
  - c. Autómatas y Máquinas de Turing.
  
2. **Complejidad Computacional**
  - a. Complejidad Temporal y Espacial.
  - b. Jerarquía Temporal.
  - c. Problemas NP-Completo.
  
3. **Análisis de Algoritmos**
  - a. Análisis Asintótico.
  - b. Algunos Algoritmos Particulares.

### **Metodología de enseñanza**

Las clases se organizan mediante:

- 1) Clases teóricas, normalmente guiadas a partir de diapositivas proyectadas y explicaciones de algunos detalles en pizarrón. También es usual presentar respuestas a los alumnos utilizando el pizarrón.
- 2) Clases prácticas, mayormente para: a) presentación de los temas de los trabajos prácticos, y b) consultas de los trabajos prácticos.
- 3) Se requiere a los alumnos la presentación de algunos ejercicios en grupo, cuya corrección se lleva a cabo de manera oral individual.

### **Propuesta de evaluación**

La evaluación del proceso se realiza mediante el seguimiento de los trabajos prácticos, que contienen uno de los temas fundamentales del contenido de la materia: la implementación de un programa donde se pueda especificar y ejecutar una máquina de Turing. Esta evaluación es oral.



**UNIVERSIDAD NACIONAL DE LA PLATA  
FACULTAD DE INFORMÁTICA**

---

La aprobación de la cursada consiste en un examen parcial escrito.  
La evaluación final se realiza mediante un examen final, el cuál adopta diferentes variaciones: puede ser escrito, oral y en algunos casos se ha determinado un coloquio de acuerdo al contenido de los exámenes parciales.

**Bibliografía Obligatoria**

Introduction to Algorithms, Second Edition, Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, The MIT Press, Cambridge, Massachusetts London, England, 2001.

Fundamentals of Algorithmics, Gilles Brassard, Paul Bratley, Prentice Hall, 1995.

Introducción a la Teoría de Autómatas, Lenguajes y Computación. Hopcroft y Ullman  
Teoría de la Computación, Lenguajes Formales, Autómatas y Complejidad. J. Glenn Brookshear

**Bibliografía Complementaria**

Teoría de Autómatas y Lenguajes Formales. Cubero, Moreno y Moriyón Salomón

**Cronograma de Clases y Evaluaciones**

Cada clase contiene su respectiva práctica con ejemplificación específica de la problemática conceptual del tema.

Clases 1 y 2: cardinalidades de conjuntos infinitos, su relación con los programas de computación y con los problemas computables y no computables.

Clase 3: modelo básico de máquina de Turing. Justificación y desarrollo formal.

Clases 4 y 5: otros modelos de máquinas de Turing. Equivalencia de modelos que impacta en la clasificación de los problemas computables y no computables. Idea de expresividad de modelos.

Clase 6: Máquinas de Turing, relación con lenguajes y relación de lenguajes con los problemas.

Clases 7 y 8: Reducibilidad, definición conceptual/formal y aplicación práctica en el contexto de los lenguajes asociados a problemas computables y no computables.



**UNIVERSIDAD NACIONAL DE LA PLATA  
FACULTAD DE INFORMÁTICA**

---

Clases 9 y 10: Complejidad temporal, utilización del modelo de Turing, de las reducciones y de terminología asociada a análisis de algoritmos. Distinción entre complejidad de problema y lo que se suele denominar como complejidad de algoritmos.

Clases 11 y 12: análisis de algoritmos, su utilización para comparación. Metodología de análisis y diferenciación tiempo, espacio y notación asintótica asociada al análisis. Métodos de análisis detallados y de asociación de notación asintótica.

Examen Parcial.

1er. Recuperatorio del Examen Parcial.

2do. Recuperatorio del Examen Parcial.

**Contacto de la Cátedra:** [cyc@ada.info.unlp.edu.ar](mailto:cyc@ada.info.unlp.edu.ar)

**Firma de los profesores responsables:**