



Resolución de Consejo Directivo **591 / 2026 - EXA -UNSa**  
EXP. 277 / 2026 - EXA -UNSa Dr. Diego Alejandro RODRÍGUEZ eleva  
Programa Analítico y Régimen de Regularidad y Promoción de la Asignatura  
ALGORITMOS Y ESTRUCTURAS DE DATOS de las carreras Licenciatura en  
Análisis de Sistemas, Plan 2010 y Tecnicatura Universitaria en Programación,  
Plan 2012

De: **EXACTAS-Secretaría Académica y de Investigación**



Salta,  
07/09/2026

VISTO: La nota mediante la cual el Dr. Diego Alejandro RODRÍGUEZ solicita la aprobación del Programa Analítico y Régimen de Regularidad y Promoción de la asignatura ALGORITMOS Y ESTRUCTURAS DE DATOS de las carreras Licenciatura en Análisis de Sistemas Plan 2010 y Tecnicatura Universitaria en Programación, Plan 2012, y

#### CONSIDERANDO:

Que las Comisiones de Carrera de Licenciatura en Análisis de Sistemas y de Tecnicatura Universitaria en Programación, y el Departamento de Informática emiten opinión favorable para la aprobación del Programa Analítico y del Régimen de Regularidad y Promoción.

Que la Comisión de Docencia e Investigación aconseja aprobar el Programa Analítico y el Régimen de Regularidad y Promoción.

Que la RCD 32/2026 EXA-UNSa dispone un formato único para la presentación de los programas de cada una de las asignaturas de las carreras que se dictan en esta Unidad Académica.

Que el Consejo Directivo, en su 14° Sesión Ordinaria realizada el 26 de agosto de 2026, aprobó el despacho de la Comisión de Docencia e Investigación.

Que, el Estatuto de la Universidad Nacional de Salta en el Artículo 117 inciso 8, establece que *"entre los deberes y atribuciones que le confiere al Consejo Directivo, incluye aprobar los programas Analíticos y la reglamentación sobre el régimen de regularidad y promoción propuesto por los módulos Académicos"*.

Por ello,

EL CONSEJO DIRECTIVO DE LA FACULTAD DE CIENCIAS EXACTAS

#### RESUELVE:

ARTÍCULO 1.- Aprobar el Programa Analítico, Régimen de Regularidad y Promoción de la asignatura ALGORITMOS Y ESTRUCTURAS DE DATOS de las carreras Licenciatura en Análisis de Sistemas, Plan 2010 y Tecnicatura Universitaria en Programación, Plan 2012, que como Anexo forma parte integrante de la presente Resolución.

ARTÍCULO 2.- Notifíquese al Dr. Diego Alejandro RODRÍGUEZ, docente responsable de la asignatura ALGORITMOS Y ESTRUCTURAS DE DATOS. Comuníquese a las Autoridades y Secretarías de esta Unidad Académica, a las Comisiones de Carrera de Licenciatura en Análisis de Sistemas y de Tecnicatura Universitaria en Programación, al Departamento de





Resolución de Consejo Directivo **591 / 2026 - EXA -UNSa**  
EXP. 277 / 2026 - EXA -UNSa Dr. Diego Alejandro RODRÍGUEZ eleva  
Programa Analítico y Régimen de Regularidad y Promoción de la Asignatura  
**ALGORITMOS Y ESTRUCTURAS DE DATOS** de las carreras Licenciatura en  
Análisis de Sistemas, Plan 2010 y Tecnicatura Universitaria en Programación,  
Plan 2012



**De: EXACTAS-Secretaría Académica y de Investigación**

Salta,  
07/09/2026

Informática, a la Dirección de Mesa de Entradas, Archivo y Digesto y a la Dirección de  
Alumnos para su toma de conocimiento, registro y demás efectos. Publíquese en el Boletín  
Oficial y en la Página WEB de la Facultad, Cumplido, archívese.

MFL



  
**LIC. MARCELA F. LÓPEZ**  
SECRETARÍA ACADÉMICA Y DE INVESTIGACIÓN  
FACULTAD DE CS. EXACTAS - UNSa



  
**Dr. JOSÉ RAMÓN MOLINA**  
DECANO  
FACULTAD DE CS. EXACTAS - UNSa

## 1. DEPARTAMENTO DOCENTE O DEPENDENCIA:

Departamento de Informática.

## 2. CÁTEDRA

Algoritmos y Estructuras de Datos.

## 3. ASIGNATURA

Algoritmos y Estructuras de Datos.

## 4. CARRERAS

Licenciatura en Análisis de Sistemas (Plan 2010)

Tecnicatura Universitaria en Programación (Plan 2012)

## 5. ÚLTIMO PROGRAMA VIGENTE APROBADO POR RESOLUCIÓN

Resolución D-EXA N° 604/2015 – Expte. EXA N° 8185/2011.

## 6. EQUIPO DOCENTE

Profesor Responsable

Dr. Diego Alejandro

Rodríguez

Docentes Auxiliares (JTP):

Lic. Carlos Nocera

Lic. Gerardo Cabero

Alumnos Auxiliares de Segunda Categoría: 1

## 7. UBICACIÓN EN EL PLAN DE ESTUDIOS

Año de cursado: 2º año

Cuatrimestre: 1º cuatrimestre

Carga horaria total: 120 horas

Teoría: 4 horas semanales

Práctica: 4 horas semanales

## 8. CONTENIDOS MÍNIMOS

| Contenido mínimo                     | Unidad/es |
|--------------------------------------|-----------|
| Teoría de las estructuras discretas  | Unidad 1  |
| Definiciones y pruebas estructurales | Unidad 1  |
| Teoría de números                    | Unidad 1  |
| Aritmética modular                   | Unidad 1  |
| Estructuras de control               | Unidad 2  |
| Recursividad                         | Unidad 2  |
| Eventos                              | Unidad 2  |



| Contenido mínimo                                                                | Unidad/es            |
|---------------------------------------------------------------------------------|----------------------|
| Excepciones                                                                     | Unidades 2, 5 y 7    |
| Concurrencia                                                                    | Unidad 2             |
| Tipos Abstractos de Datos: definiciones, especificación abstracta y operaciones | Unidad 3             |
| Isomorfismo                                                                     | Unidad 3             |
| Contenedores lineales                                                           | Unidad 4             |
| Estructuras de datos                                                            | Unidades 4, 5, 6 y 7 |
| Tipos de datos recursivos                                                       | Unidad 2             |
| Representación de datos en memoria                                              | Unidad 4             |
| Estrategias de implementación                                                   | Unidades 3 y 5       |
| Manejo de memoria en ejecución                                                  | Unidad 5             |
| Algoritmos en grafos                                                            | Unidad 6             |
| Algoritmos de análisis y manipulación de grafos                                 | Unidad 6             |
| Costos                                                                          | Unidad 6             |
| Aplicación                                                                      | Unidades 6 y 7       |
| Estructuras arbóreas: árboles generales y n-arios, binarios y balanceados       | Unidad 7             |
| Estrategias de diseño de algoritmos                                             | Unidades 6 y 7       |

## 9. CORRELATIVIDADES

### Para el cursado

#### LAS

- Regularizada: Análisis Matemático I, Programación
- Aprobada: Elementos de Programación

#### TUP

- Regularizada: Análisis Matemático I, Programación
- Aprobada: Elementos de Programación, Matemática para Informática

### Para el examen final

#### LAS

- Aprobada: Análisis Matemático I, Programación

#### TUP

- Aprobada: Análisis Matemático I, Programación

## 10. JUSTIFICACIÓN

### 10.1 Justificación de la asignatura

La asignatura Algoritmos y Estructuras de Datos constituye un eje central en la formación del estudiante de informática, al brindar fundamentos para la resolución de problemas mediante algoritmos y estructuras de datos.

Durante el cursado se abordan conceptos de estructuras discretas, recursividad y abstracción de datos, aplicados a la implementación de estructuras lineales y no



lineales, tales como listas, árboles y grafos. Estos contenidos favorecen el desarrollo del pensamiento lógico y algorítmico, así como la comprensión de técnicas fundamentales para la construcción de soluciones informáticas.

## 10.2 Articulación Académica

La asignatura articula horizontalmente con materias del área de programación y matemática, particularmente con Programación, Elementos de Programación y Matemática para Informática, recuperando y profundizando conocimientos relacionados con lógica, estructuras discretas y desarrollo de algoritmos.

Asimismo, articula verticalmente con asignaturas posteriores en las que se profundiza el desarrollo de programas y la implementación de soluciones computacionales que requieren el uso de estructuras de datos y algoritmos.

## 10.3 Aporte al perfil profesional

La asignatura contribuye a competencias relacionadas con:

- El análisis, diseño e implementación de soluciones informáticas.
- El diseño y adaptación de software.
- La aplicación de técnicas de algoritmos y estructuras de datos.
- La enseñanza de conocimientos básicos y técnicos del área informática.

## 10.4 Ejes Transversales Abordados

La asignatura contribuye al desarrollo de los siguientes ejes transversales definidos en la Res. ME N° 981/2025: ET01 – Identificación, formulación y resolución de problemas de informática (aporte ALTO); ET02 – Concepción, diseño y desarrollo de proyectos de informática (aporte MEDIO); ET04 – Utilización de técnicas y herramientas de aplicación en informática (aporte ALTO); ET06 – Fundamentos para el desempeño en equipos de trabajo (aporte MEDIO); ET07 – Fundamentos para la comunicación efectiva (aporte MEDIO); y ET10 – Fundamentos para el aprendizaje continuo (aporte MEDIO).

## 11. OBJETIVOS GENERALES

Al finalizar la asignatura, el estudiante será capaz de:

- Desarrollar pensamiento lógico y algorítmico para la resolución de problemas computacionales.
- Comprender y aplicar los principios de abstracción de datos mediante el uso de Tipos Abstractos de Datos.
- Analizar, seleccionar e implementar estructuras de datos lineales y no lineales utilizando estrategias adecuadas de representación y manejo de memoria.
- Resolver problemas computacionales mediante la selección e implementación de estructuras de datos y estrategias algorítmicas adecuadas.



- Representar y manipular estructuras de datos complejas, tales como listas enlazadas, árboles y grafos, utilizando un lenguaje de programación.
- Analizar la eficiencia de algoritmos y estructuras de datos considerando criterios de desempeño, claridad y mantenibilidad.
- Comunicar y defender decisiones de diseño e implementación utilizando terminología técnica apropiada.
- Trabajar de manera autónoma y colaborativa en el desarrollo de soluciones computacionales integradoras.

## 12. PROGRAMA ANALÍTICO

### Unidad 1: Fundamentos de estructuras discretas

#### Objetivo específico

Revisar conceptos de estructuras discretas necesarios para el desarrollo y análisis de algoritmos.

#### Contenidos

Teoría de las estructuras discretas. Definiciones formales y pruebas estructurales aplicadas a problemas computacionales. Teoría de números: divisibilidad, teorema de la división, algoritmo de Euclides, máximo común divisor y números primos. Aritmética modular y congruencias.

### Unidad 2: Recursividad

#### Objetivo específico

Comprender el paradigma recursivo y su relación con la construcción de algoritmos.

#### Contenidos

Estructuras de control en algoritmos: secuencia, selección e iteración. Recursividad: concepto, propiedades y aplicaciones. Tipos de recursividad. Tipos de datos recursivos. Comparación entre soluciones recursivas e iterativas. Aspectos relacionados con la ejecución de algoritmos: manejo de excepciones, eventos y concurrencia.

### Unidad 3: Tipos Abstractos de Datos

#### Objetivo específico

Comprender y aplicar el concepto de abstracción de datos en el diseño e implementación de estructuras de datos.

#### Contenidos

Tipos Abstractos de Datos: definiciones. Especificación abstracta. Operaciones sobre el tipo abstracto de datos. Isomorfismo entre tipos abstractos de datos. Estrategias de implementación.



## **Unidad 4: Contenedores lineales**

### **Objetivo específico**

Analizar estructuras de datos lineales y sus operaciones fundamentales.

### **Contenidos**

Contenedores lineales. Estructuras de datos secuenciales. Representación de datos en memoria. Listas. Pilas. Colas. Colas circulares. Colas con prioridad.

## **Unidad 5: Listas enlazadas**

### **Objetivo específico**

Comprender las estructuras dinámicas y su implementación en memoria.

### **Contenidos**

Listas enlazadas: listas simplemente enlazadas, doblemente enlazadas y circulares. Manejo de memoria en ejecución. Estrategias de implementación. Manejo de excepciones en operaciones sobre estructuras de datos.

## **Unidad 6: Grafos**

### **Objetivo específico**

Comprender e implementar representaciones y algoritmos sobre grafos para la resolución de problemas computacionales.

### **Contenidos**

Grafos: definiciones, propiedades y clasificación. Representación de grafos: matriz de adyacencia y listas de adyacencia. Algoritmos de análisis y manipulación de grafos. Recorridos en grafos: búsqueda en anchura (BFS) y búsqueda en profundidad (DFS). Problemas clásicos: camino mínimo y árbol generador mínimo. Aplicaciones. Estrategias de diseño de algoritmos aplicadas a grafos.

## **Unidad 7: Árboles**

### **Objetivo específico**

Comprender e implementar estructuras arbóreas y aplicar algoritmos asociados a su manipulación y recorrido.

### **Contenidos**

Estructuras arbóreas. Árboles generales y n-arios. Árboles binarios. Árboles binarios de búsqueda. Árboles balanceados. Recorridos de árboles. Aplicaciones. Estrategias de diseño de algoritmos aplicadas a estructuras arbóreas. Manejo de excepciones en operaciones sobre estructuras arbóreas.



**13. PROGRAMA DE TRABAJOS PRÁCTICOS**

| TP N° | Tema                        | Unidad     | Clases Estimadas |
|-------|-----------------------------|------------|------------------|
| 1     | Teoría de números           | 1          | 2 clases         |
| 2     | Aritmética modular          | 1          | 2 clases         |
| 3     | Recursividad                | 2          | 2 clases         |
| 4     | Tipos Abstractos de Datos   | 3          | 2 clases         |
| 5     | Pilas y colas               | 4          | 4 clases         |
| 6     | Listas enlazadas            | 5          | 3 clases         |
| 7     | Grafos                      | 6          | 4 clases         |
| 8     | Árboles                     | 7          | 4 clases         |
| 9     | Trabajo Práctico Integrador | Integrador | 4 clases         |

**14. LABORATORIOS**

La asignatura no contempla prácticas de laboratorio en el sentido experimental propio de disciplinas como las ciencias naturales o de la salud.

No obstante, se desarrollan actividades prácticas en aula y en entornos informáticos, en las cuales los estudiantes implementan algoritmos y estructuras de datos utilizando un lenguaje de programación, integrando los contenidos teóricos con la práctica.

**15. ESTRATEGIAS DE ENSEÑANZA**

El dictado de la asignatura se organiza bajo un enfoque de aprendizaje activo y centrado en el estudiante, articulando clases teóricas y prácticas.

En las clases teóricas se desarrollan los conceptos fundamentales mediante exposiciones, resolución guiada de problemas y análisis de ejemplos representativos. Las clases prácticas se orientan a la resolución de ejercicios y a la implementación de algoritmos y estructuras de datos utilizando un lenguaje de programación.

Durante el cursado, los estudiantes desarrollan un Trabajo Práctico Integrador en el que aplican los contenidos de la asignatura para resolver un problema computacional mediante el uso de distintas estructuras de datos.



La asignatura utiliza una plataforma virtual como repositorio de materiales y canal de comunicación con los estudiantes. Asimismo, en horarios de consulta suelen realizarse actividades prácticas complementarias orientadas a profundizar contenidos y atender dificultades específicas.

Las actividades propuestas promueven la resolución de problemas, el análisis de alternativas de implementación y la selección fundamentada de estructuras de datos. El Trabajo Práctico Integrador favorece además el desarrollo de habilidades de trabajo autónomo, comunicación técnica y exposición de soluciones computacionales.

## 16. ESTRATEGIAS DE EVALUACIÓN

De acuerdo con lo establecido en el plan de estudios vigente, la asignatura se aprueba mediante examen final. La evaluación del aprendizaje es continua durante el cursado.

Se consideran las siguientes instancias de evaluación:

- evaluaciones parciales teórico-prácticas;
- actividades prácticas desarrolladas durante el cursado; • desarrollo de un Trabajo Práctico Integrador (TPI).

Las evaluaciones parciales permiten valorar la comprensión de los conceptos fundamentales, la capacidad de análisis y la resolución de problemas vinculados con algoritmos y estructuras de datos.

El Trabajo Práctico Integrador consiste en el diseño e implementación de una solución computacional que integra los contenidos desarrollados durante la asignatura. En esta instancia se evalúa la correcta utilización de estructuras de datos, la adecuación de la solución al problema planteado y la calidad de la implementación.

Asimismo, la presentación y defensa del trabajo permiten valorar la comunicación técnica, la fundamentación de decisiones de implementación y la comprensión de los conceptos involucrados.

Las evaluaciones parciales contarán con instancias de recuperación de acuerdo con la normativa vigente.

## 17. RECURSOS MATERIALES

Para el normal desarrollo de la asignatura se precisa de un gabinete de computadoras con conexión a internet, proyector multimedia, pizarrón y un curso en la plataforma Moodle de la Facultad.

A estos recursos se suman diapositivas de los temas cubiertos, apuntes, libros específicos, código fuente y librerías para la implementación y prueba de los algoritmos y estructuras de datos.

## 18. CONDICIONES PARA REGULARIZAR

Para regularizar la asignatura el estudiante deberá cumplir simultáneamente con las siguientes condiciones:



- a) **Aprobación de evaluaciones parciales:** aprobar las evaluaciones parciales o sus correspondientes recuperatorios con una calificación mínima de 60 (60/100).
- b) **Aprobación de trabajos prácticos:** aprobar el Trabajo Práctico Integrador propuesto por la cátedra.
- c) **Actividades prácticas:** cumplir con las actividades prácticas previstas durante el cursado.

## 19. CONDICIONES PARA ACREDITAR

### Estudiante Regular

El examen final consiste en la presentación y defensa del Trabajo Práctico Integrador, en la cual el estudiante deberá exponer la solución desarrollada. A continuación se realizarán preguntas teórico-prácticas orientadas a valorar la comprensión conceptual y la fundamentación de las decisiones de implementación adoptadas. El trabajo deberá incorporar las mejoras o correcciones indicadas por la cátedra durante el cursado.

### Estudiante Libre

El estudiante deberá aprobar:

- Una evaluación teórica-práctica sobre los contenidos del programa.
- El desarrollo de un Trabajo Práctico Integrador previamente acordado con la cátedra.

## 20. BIBLIOGRAFÍA

### a) Bibliografía específica

- Aho, A. V., Hopcroft, J. E., & Ullman, J. D. (1988). *Data Structures and Algorithms*. Addison-Wesley.
- Drozdek, A. (2007). *Data structures and algorithms in Java (2nd ed.)*. Thomson.
- Goodrich, M. T., Tamassia, R., & Goldwasser, M. H. (2014). *Data structures and algorithms in Java (6th ed.)*. Wiley.
- Grimaldi, R. (1989). *Matemática discreta y combinatoria*. Addison-Wesley.
- Joyanes Aguilar, L. (1999). *Estructuras de datos*. McGraw-Hill.

### b) Bibliografía general o ampliatoria

- Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2009). *Introduction to algorithms (3rd ed.)*. MIT Press.
- Kleinberg, J., & Tardos, É. (2005). *Algorithm design*. Addison-Wesley.
- Knuth, D. E. (1997). *The art of computer programming*. Addison-Wesley.
- Mora, W. (2014). *Introducción a la teoría de números: ejemplos y algoritmos*. Editorial de la Universidad de Costa Rica.

Skiena, S. S. (2020). *The algorithm design manual (3rd ed.)*. Springer.

