

PROGRAMACIÓN 3

Carrera/ Plan:

Ingeniería en Computación Plan 2008 / 2011

Año: 2°

Régimen de Cursada: *Semestral*

Carácter (Obligatoria/Optativa): *Obligatoria*

Correlativas: *Programación II*

Profesor/es: *Alejandra Schiavoni –*

Laura Fava – Pablo Iuliano

Hs. Semanales: *6 hs.*

Año 2026

FUNDAMENTACIÓN

Esta materia es de gran importancia dentro de la carrera, ya que en ella se brindan los fundamentos de las estructuras de datos no lineales y del análisis de eficiencia de los algoritmos.

Los objetivos que se plantean en este curso consisten en lograr que los alumnos:

- a) adquieran un conocimiento exhaustivo de las principales estructuras de datos y aprendan a implementarlas en lenguaje Java, definiendo en forma eficiente sus clases y métodos;
- b) adquieran las herramientas necesarias para evaluar y aplicar las estructuras de datos adecuadas para la resolución de problemas concretos;
- c) aprendan a analizar algoritmos y evaluar su eficiencia, utilizando un formalismo matemático para estimar el tiempo de ejecución requerido en función de la entrada de los mismos.
- d) que logren implementar soluciones eficientes a problemas complejos, permitiendo la gestión eficiente de grandes cantidades de información y la optimización de procesos.

OBJETIVOS GENERALES

Que los alumnos adquieran un conocimiento exhaustivo de las principales estructuras de datos y aprendan a implementarlas en forma eficiente; aprendan a analizar diferentes algoritmos de acceso y manejo a tales estructuras de datos, utilizando un formalismo matemático para estimar la eficiencia de los algoritmos.

EJES TRANSVERSALES

- 3. Especificación, proyecto y Desarrollo de Software y Sistemas Conjuntos de Hardware y Software haciendo uso de conceptos, métodos y herramientas de gestión de proyectos, ingeniería de software, base de datos, experiencia del usuario, elicitación, análisis, especificación y validación de requerimientos. (2-Medio)
- 4. Desarrollo de Redes de Computadoras y de Redes de Computadoras de área amplia, locales, inalámbricas y móviles. (1-Bajo)
- 06. Proyecto, desarrollo, dirección, control, construcción, operación y mantenimiento de Sistemas de Procesamiento de Señales, Sistemas Embebidos y sus periféricos incluido en software de soporte, Sistemas Computarizados de automatización y control y Sistemas Conjuntos de Hardware y Software. (1- Bajo)
- 09. Identificación, formulación y resolución de problemas de ingeniería en computación. (1-Bajo)
- 12. Utilización de técnicas y herramientas de aplicación en la ingeniería en computación. (1-Bajo)
- 14. Fundamentos para el desempeño en equipos de trabajo. (1-Bajo)
- 15. Fundamentos para una comunicación efectiva. (1-Bajo)

CONTENIDOS MÍNIMOS (de acuerdo al Plan de Estudios)

- Estructuras de Datos no lineales
- Recursión
- Grafos
- Algorítmica
- Complejidad

PROGRAMA ANALÍTICO

1. Conceptos de la plataforma de ejecución y de desarrollo de JAVA. Conceptos básicos de Programación Orientada a Objetos: encapsulamiento, ocultamiento de información, clases e interfaces, objetos, herencia, polimorfismo, *binding* dinámico.

2.- Estructuras de datos recursivas: Listas, Árboles y Grafos. Distintas representaciones y estrategias de implementación de cada una. Resolución de problemas aplicando cada una de las estructuras. Repaso de Listas dinámicas, pilas y colas: representación, acceso y recorridos.

3.- Árboles binarios. Árboles de expresión. Recorridos ordenados (InOrden, PostOrden, PreOrden). Construcción de árboles. Búsquedas. Actualización: inserción y borrado. Análisis del tiempo de ejecución de estas operaciones.

4.- Árboles generales. Distintas implementaciones. Recorridos ordenados (InOrden, PostOrden, PreOrden). Búsquedas. Actualización: inserción y borrado. Análisis de la eficiencia de cada algoritmo. Aplicaciones.

5.- Cola de prioridades. Heap binaria. Implementaciones y operaciones. Operaciones de inserción, borrado y construcción. Aplicaciones: Selección y Ordenación (Heapsort). Análisis de la eficiencia.

6.- Análisis de algoritmos. Análisis asintótico, comportamiento en el mejor caso, caso promedio y peor caso. Modelo computacional. Concepto de tiempo de ejecución. Notación $O()$, Ω , Θ . Reglas generales para el cálculo del tiempo de ejecución. Cálculo de tiempo y orden de ejecución en algoritmos iterativos y recursivos. Comparación de distintas estrategias de diseño de algoritmos.

7.- Grafos orientados y no orientados. Grafos pesados. Distintas representaciones: Listas de Adyacencia y Matriz de Adyacencia. Definiciones básicas y conceptos fundamentales. Grafos acíclicos. Grafos conexos y dígrafos fuertemente conexos.

8.- Algoritmos de recorrido DFS y BFS. Árbol generador DFS: en grafos dirigidos y no dirigidos. Determinación de componentes conexas y fuertemente conexas. Análisis del tiempo de ejecución de los algoritmos mencionados.

9.- Ordenamiento topológico. Ejemplos de aplicación. Distintas implementaciones. Análisis de la eficiencia de cada uno.

10.- Problema del camino mínimo: estudio de distintos casos. Su desarrollo para grafos pesados y no pesados; y grafos dirigidos y acíclicos. Algoritmos de Dijkstra y Floyd. Árbol generador mínimo. Algoritmos de Prim y Kruskal. Análisis del tiempo de ejecución de los algoritmos vistos.

BIBLIOGRAFÍA

Título	Autor(es)	Editorial	Año de edición
<i>Data Structures And Algorithm Analysis in Java; 3rd Edition</i>	Mark Allen Weiss	Addison-Wesley	2012
<i>Data Structures and Algorithms</i>	A. Aho, J. Hopcroft, J. D. Ullman	Addison-Wesley	1983
<i>Thinking in Java, fourth edition.</i>	Bruce Eckel,	Prentice Hall,	2006

BIBLIOGRAFÍA COMPLEMENTARIA

Título	Autor(es)	Editorial	Año de edición
<i>Data Structures in Java; 1st Edition</i>	Thomas A. Standish	Addison-Wesley	1997
<i>Data Structures and Problem Solving using Java; 4th Edition.</i>	Mark Allen Weiss	Addison-Wesley	2009
<i>Introduction to algorithm; third edition</i>	Thomas H. Cormen	The MIT Press	2009

METODOLOGÍA DE ENSEÑANZA

Las clases teóricas son dos veces por semana de una hora y media cada una, en las que se dicta el contenido referido a las estructuras de datos mencionadas, análisis de complejidad de los algoritmos y conceptos referidos a Programación Orientada a Objetos y al lenguaje Java, articulados con las estructuras de datos y los algoritmos vistos.

Durante el desarrollo de la materia, se estudian en detalle estructuras de datos avanzadas, analizando distintas formas de implementación y algoritmos de acceso a dichas estructuras. Respecto de cada una de ellas, se ven tanto aspectos teóricos de modelización como prácticos de implementación usando distintas estrategias y evaluando la eficiencia de cada una. La implementación de las estructuras de datos se realiza usando el lenguaje Java, que les permite tomar contacto con tecnologías y paradigmas de programación actuales.

El tema de análisis de la complejidad de los algoritmos, es transversal a toda la materia y tiene como objetivo que los alumnos aprendan a evaluar la eficiencia de todos los algoritmos. Para lograr este objetivo, se les explica el formalismo matemático subyacente, basado en el análisis de crecimiento de funciones trigonométricas y mecanismos para encontrar cotas asintóticas.

Se estimula el análisis, diseño e implementación de soluciones a problemas del mundo real, como paso inicial al desarrollo de proyectos. Se pone énfasis en el estudio de las estructuras de datos y su aplicabilidad en problemas reales concretos resolubles desde la informática, evaluando ventajas y desventajas del uso de cada una de ellas. Se destaca la importancia de las estructuras de datos en su utilización para modelar y simular sistemas complejos. Las mismas se estudian como base en el desarrollo de software utilizado en redes de computadoras, en sistemas computarizados tanto de hardware como de software. A partir del análisis realizado, se implementan posibles soluciones, evaluando la eficiencia de cada una. El objetivo del almacenamiento y procesamiento de datos es aumentar el valor de la información y facilitar la toma de decisiones. La asignatura aporta la capacidad para desarrollar ideas lógicas e identificar el proceso de desarrollo de aplicaciones enfocado en la resolución de problemas. Los sistemas computarizados de automatización y control utilizan sensores, instrumentos, computadoras y procesamiento de datos para lograr operaciones automáticas y eficientes. Las estructuras de datos son una herramienta fundamental para cualquier ingeniero que trabaje con datos, ya que permiten organizar, manipular y acceder a la información de manera eficiente, lo que a su vez mejora el rendimiento y la eficiencia de sus aplicaciones y sistemas.

Las clases teóricas son de tipo expositivas, se utilizan presentaciones electrónicas que comprenden explicaciones detalladas de cada tema realizadas por el profesor responsable y los alumnos intervienen realizando consultas y preguntas. Para reforzar algunos temas, se dan ejercicios que se resuelven en clase en conjunto entre el docente y los alumnos, utilizando la pizarra.

Las clases prácticas se llevan a cabo en aulas equipadas con computadoras, con duración de dos horas, donde los ejercicios de implementación se resuelven en lenguaje Java usando un ambiente de desarrollo apropiado. El objetivo es que los alumnos aprendan técnicas y herramientas de aplicación en Informática, siguiendo las tendencias marcadas por los cambios tecnológicos y la aplicación efectiva de las mismas.

Los auxiliares docentes, jefes de trabajos prácticos y ayudantes, responden las consultas de los alumnos y realizan explicaciones generales en la pizarra en caso de ser necesario.

Durante el desarrollo de las clases prácticas, se organizan actividades por equipos de trabajo, con un número variable de alumnos entre 2 y 4, que constan de la resolución de diferentes problemas planteados por el docente a cargo. En la evaluación del trabajo en equipo, hay presentaciones donde se exponen las explicaciones y fundamentaciones correspondientes que permiten calificar diferentes aptitudes de los miembros del equipo, tales como conocimientos adquiridos, interés en el tema y predisposición al trabajo colaborativo.

Los contenidos teóricos y prácticos están estrechamente vinculados, siendo éstos últimos una aplicación directa de los temas teóricos impartidos. A lo largo del desarrollo de la materia, los docentes proponen la búsqueda y consulta de bibliografía actualizada sobre los temas y el planteo y discusión de diferentes soluciones y alternativas que permiten resolver cada uno de los problemas.

Las clases son presenciales, sin embargo se podrán hacer virtual un porcentaje menor de algunas actividades, si es pertinente con la temática abordada.

Con anterioridad a cada una de las instancias de evaluación se ofrecen clases de consulta en función de las problemáticas más relevantes.

Se utiliza Moodle (plataforma de aprendizaje virtual), que ofrece una funcionalidad muy útil para la organización del curso. A través de ella, se publican las clases teóricas, los trabajos prácticos y las explicaciones de los mismos, material adicional de consulta y la bibliografía. Además, se usan los Foros para realizar consultas, anuncios, discusiones, etc, las Wikis para que los alumnos trabajen en grupo, las Tareas para que los alumnos realicen entregas, las Encuestas cuando es necesario consultar a los alumnos, por ejemplo para que opten por un turno de práctica. También cuenta con la herramienta de video conferencia Big Blue Button, para la realización de actividades virtuales.

EVALUACIÓN

La asignatura cuenta con un régimen que permite a los alumnos promocionar la materia durante el semestre de cursada sin rendir examen final.

Durante el desarrollo de la cursada se tomará asistencia en las clases prácticas, lo que se considerará como nota de concepto para acceder al régimen de promoción. Además, se implementarán entregas y presentaciones de ejercicios en fechas preestablecidas. El docente responsable de la práctica registra los resultados de las presentaciones para ser tenidas en cuenta en las evaluaciones de los alumnos. Estas entregas permiten realizar un seguimiento de los estudiantes y brindarle una devolución de su progreso. También se considerarán aspectos relacionados con los aportes realizados por los alumnos, en base a búsquedas e investigaciones llevadas a cabo sobre los temas vistos.

La evaluación de la asignatura consiste en un parcial teórico-práctico que incluye módulos (temas) bien diferenciados, los cuales se evalúan en forma independiente. El parcial tiene dos instancias para recuperar el o los módulos desaprobados. Cada uno de los módulos se aprueba con nota 4 o superior, requiriendo en algunos casos un puntaje mínimo en algunas de las secciones. Si el promedio de las notas de los módulos es 6 o superior, el alumno obtiene la promoción de la materia y la obtención de una nota entre 4 y menor que 6 puntos, le otorga la aprobación de los trabajos prácticos, debiendo el alumno rendir un examen final.

En las evaluaciones se plantean ejercicios a resolver similares a los incluidos en los trabajos prácticos. Los ejercicios planteados en las evaluaciones comprenden el uso de las estructuras de datos y el análisis de complejidad de los algoritmos. Consisten en aplicar las estructuras de datos vistas en la materia e implementar algoritmos para plantear una solución a un problema concreto.

En la evaluación se tiene en cuenta el análisis del problema y el diseño de una solución adecuada y eficiente como punto inicial para la futura implementación de un posible proyecto global. Se evalúa no sólo la correctitud de la solución, lo que involucra también la elección de las estructuras de datos adecuadas, sino también la utilización de las técnicas y lenguaje de programación como herramienta de aplicación. Además, se plantean problemas en los que el alumno debe utilizar los métodos matemáticos formales vistos en las clases para estimar el tiempo de ejecución de los algoritmos en función del tamaño de la entrada. En las evaluaciones también se incluyen preguntas sobre conceptos teóricos relativos a los temas de la currícula.

El examen final es teórico-práctico, una parte con enunciados de opción múltiple y otra con preguntas de respuesta breve. La parte de las preguntas de opción múltiple permite evaluar conceptos y principios

teóricos vistos en la asignatura y la sección de preguntas de respuesta breve permite la evaluación de problemas teórico-prácticos donde el estudiante aplica los conocimientos teóricos adquiridos en la resolución de problemas concretos.

CRONOGRAMA DE CLASES Y EVALUACIONES

Clase	Fecha	Contenidos/Actividades
Semana 1	09/03/2026	• Introducción. Clases e instancias en Java. Pasaje de parámetros. • Herencia. Clases abstractas • Herencia con Pilas y Colas y List de JAVA. Tipos Genéricos. • Resolución del TP 1 – El lenguaje Java, TAD pilas y colas, List y Tipos Genéricos.
Semana 2	16/03/2026	• Árboles binarios: representaciones, recorridos. • Constructores en java. Constructores y herencia.
Semana 3	23/03/2026	• Aplicaciones de árboles binarios: árboles de expresión. • Árboles generales: ejemplos y terminología. • Interfaces en java, la interface comparable. Ejemplos de uso de esta interface en estructuras ordenadas. • Resolución del TP2 - Árboles Binarios
Semana 4	30/03/2026	• Distintas representaciones e implementaciones de Árboles Generales. Recorridos. Aplicaciones. • Interfaces. Interface Comparable. • Ejemplos. Interfaces. La interfaz Comparable. Ejemplo de uso en ABB
Semana 5	06/04/2026	• Cola de prioridades: concepto y características. Heap binaria: propiedades e implementación. Operaciones de acceso y construcción de una Heap. • Aplicación de Interfaces. Ejemplos con Heap. • Resolución del TP 3 – Árboles Generales
Semana 6	13/04/2026	• Aplicaciones de Heap: Selección y Ordenación (HeapSort). Análisis de la eficiencia.
Semana 7	20/04/2026	• Análisis de algoritmos. Modelo computacional. Concepto de tiempo de ejecución. • Paquetes y especificadores de acceso. Ejemplos de especificadores de acceso con árboles generales.
Semana 8	27/04/2026	• Notación "Big-Oh". Reglas generales para el cálculo del tiempo de ejecución.

		• Cálculo en algoritmos iterativos y recursivos. Ejemplo: Subsecuencia de suma máxima • Resolución del TP 4 – Análisis de algoritmos
Semana 9	04/05/2026	• Grafos: ejemplos y terminología. Grafos orientados y no orientados. Grafos pesados. Definiciones básicas y conceptos fundamentales. Distintas representaciones: Listas y Matriz de adyacencia. Grafos acíclicos. Grafos conexos y dígrafos fuertemente conexos.
Semana 10	11/05/2026	• Definición en java de Grafos con listas y con Matriz de adyacencia. Operaciones básicas. • Algoritmos de recorrido DFS y BFS. Árbol generador DFS: en grafos dirigidos y no dirigidos. Determinación de componentes conexas y fuertemente conexas. Análisis del tiempo de ejecución de los algoritmos mencionados. • Recorridos de grafos básicos, modificaciones a dichos algoritmos en java.
Semana 11	18/05/2026	• Ordenamiento topológico. Definición. Ejemplos. Distintas implementaciones. • Análisis del tiempo de ejecución de los algoritmos mencionados. • Resolución del TP 5 – Grafos
Semana 12	25/05/2026	• Problema del camino mínimo: introducción y estudio de distintos casos. Aplicaciones. Algoritmos para cada caso. Análisis del tiempo de ejecución. Caminos mínimos desde un origen para: grafos no pesados y grafos con pesos positivos - o Algoritmo de Dijkstra (versión original)
Semana 13	01/06/2026	• Problema del camino mínimo (continuación): - o Algoritmo de Dijkstra (implementado con heap) - o Caminos mínimos desde un origen para: grafos con pesos positivos y negativos y grafos dirigidos y acíclicos - o Caminos mínimos entre cada par de vértices. Algoritmo de Floyd.
Semana 14	08/06/2026	• Árbol generador mínimo. Definición. Algoritmos de Prim y Kruskal. Análisis del tiempo de ejecución de los algoritmos vistos.
Semana 15	15/06/2026	• Consultas

Semana 16	22/06/2026	Consultas
Semana 17	29/06/2026	Muestra y Consultas
Semana 18	06/07/2026	Muestra y Consultas
Semana 19	13/07/2026	Muestra y Consultas
20/07/2026 al 02/08/2026 Receso de invierno A confirmar		
Semana 20	03/08/2026	Muestra

Evaluaciones previstas	Fecha
Evaluación - 1era fecha - Tema Árboles	Semana 20/04/2026
Evaluación - 1era fecha - Temas Análisis de Algoritmos y Grafos	Semana 08/06/2026
1er Recuperatorio de la Evaluación	Semana 29/06/2026
2do Recuperatorio de la Evaluación	Semana 13/07/2026

Contacto de la cátedra (mail, sitio WEB, plataforma virtual de gestión de cursos):

Mail de contacto: ales@info.unlp.edu.ar

Plataforma virtual: <https://catedras.info.unlp.edu.ar/>

Firma del/los profesor/es

María Alejandra Schiavoni
Prof. Titular