



UNIVERSIDAD NACIONAL DE LA PLATA  
FACULTAD DE INFORMÁTICA

---

**ALGORITMOS Y ESTRUCTURAS DE  
DATOS**

**Año 2018**

**Carrera/ Plan:**

*Licenciatura en Informática*  
Plan 2003-07/2012/2015  
*Licenciatura en Sistemas*  
Plan 2003-07/2012/2015  
*Analista Programador Universitario*  
Plan 2007/2015  
*Analista en TIC Plan 2017*

**Año:** 2°

**Régimen de Cursada:** Semestral

**Carácter:** Obligatoria

**Correlativas:** Matemática 2 – Taller de Programación

**Coordinador:** Javier Díaz

**Profesores:** Alejandra Schiavoni –  
Catalina Mostaccio -  
Laura Fava –  
Pablo Iuliano

**Hs. semanales:** 6 hs.

---

**FUNDAMENTACIÓN**

Esta materia es de gran importancia dentro de las carreras, 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:

- 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;
- 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.

**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.

**CONTENIDOS MINIMOS:**

- Estructuras de Datos no lineales



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

- 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, Arboles 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. Árboles binarios de búsqueda. Árboles binarios de búsqueda balanceados: Árboles AVL. Recorridos ordenados (InOrden, PostOrden, PreOrden). Búsquedas. Actualización: inserción y borrado. Análisis de 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  $O()$ ,  $\Omega$ ,  $\Theta$ . 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.



UNIVERSIDAD NACIONAL DE LA PLATA  
FACULTAD DE INFORMÁTICA

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 OBLIGATORIA**

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, en las que se dictan el contenido referido a las estructuras de datos mencionadas y el 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.

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 donde los ejercicios se resuelven en lenguaje Java usando un ambiente de desarrollo apropiado. 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.



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

Los contenidos teóricos y prácticos están estrechamente vinculados, siendo éstos últimos una aplicación directa de los temas teóricos impartidos.

Las clases teóricas y las clases prácticas se dictan en horarios de mañana y de tarde. Los alumnos eligen libremente al comienzo de la cursada el horario de teoría y de práctica al que van a asistir, sin embargo, tienen la posibilidad de asistir eventualmente a otro turno de práctica ante la imposibilidad de hacerlo en su horario.

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.

### **EVALUACIÓN**

La asignatura cuenta con un régimen de promoción 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, y el alumno debe contar con un porcentaje establecido por la cátedra para acceder al régimen de promoción. Además, se implementarán entregas de ejercicios en fechas preestablecidas.

La evaluación de la asignatura consiste en dos parciales teórico-prácticos con puntaje ponderado, donde se plantean ejercicios a resolver similares a los trabajados en los prácticos y preguntas sobre conceptos teóricos relativos a los temas de la currícula. La aprobación de ambos parciales con un promedio de 6 o superior, le otorga al alumno la promoción de la materia, y la obtención de una nota entre 4 y 6 puntos, le otorga la aprobación de los trabajos prácticos, debiendo el alumno rendir un examen final.

El examen final es teórico-práctico, una parte con enunciados de opción múltiple y otra con preguntas de respuesta breve.

### **CRONOGRAMA DE CLASES Y EVALUACIONES**

<b>Clase</b>	<b>Contenidos/Actividades</b>	<b>Evaluaciones previstas</b>
Semana 1 Comienzo Semana 12/03/2018	<ul style="list-style-type: none"><li>• La Plataforma Java.</li><li>• Revisión de Fundamentos de Programación Orientada a Objetos.</li><li>• Aplicación: Ejecución de programas java con y sin IDE (<i>Integrated</i>)</li></ul>	



UNIVERSIDAD NACIONAL DE LA PLATA  
FACULTAD DE INFORMÁTICA

	<i>Development Environment</i>	
Semana 2	<ul style="list-style-type: none"><li>• Clases y objetos en Java</li><li>• Herencia y clases abstractas. Ejemplo de listas con Herencia en java.</li><li>• Tipos de datos genéricos. Ejemplos.</li><li>• Resolución del TP 1 – JDK y uso de Eclipse. Introducción al lenguaje Java</li></ul>	
Semana 3	<ul style="list-style-type: none"><li>• Repaso de conceptos de Listas, definición de la estructura y operaciones. Pilas y Colas. Listas con objetos.</li><li>• Noción intuitiva de análisis de algoritmos y complejidad.</li><li>• Árboles binarios: representaciones, recorridos.</li><li>• Constructores en java. Constructores y herencia.</li></ul>	
Semana 4	<ul style="list-style-type: none"><li>• Aplicaciones de árboles binarios: árboles de expresión.</li><li>• Árboles generales: ejemplos y terminología. Distintas representaciones e implementaciones. Recorridos. Aplicaciones.</li><li>• Interfaces en java, la interface comparable. Ejemplos de uso de esta interface en estructuras ordenadas.</li><li>• Resolución del TP 2 – Encapsulamiento y Abstracción</li></ul>	
Semana 5	<ul style="list-style-type: none"><li>• Cola de prioridades: concepto y características. Heap binaria: propiedades e implementación. Operaciones de acceso y construcción.</li><li>• Aplicaciones de Heap: Selección y Ordenación (HeapSort). Análisis de la eficiencia.</li><li>• Repaso de Interfaces. Ejemplos con Heap.</li><li>• Resolución del TP 3 – Árboles Binarios</li></ul>	
Semana 6	<ul style="list-style-type: none"><li>• Árboles binarios de búsqueda. Repaso de las operaciones. Árboles AVL: definición y representación. Concepto de Balanceo.</li><li>• Árboles AVL: implementación de las</li></ul>	



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

	<p>operaciones. Rotaciones simples y dobles.</p> <ul style="list-style-type: none"> <li>• Paquetes y especificadores de acceso. Ejemplos de especificadores de acceso con árboles generales y árboles binarios de búsqueda.</li> </ul>	
Semana 7	<ul style="list-style-type: none"> <li>• Análisis de algoritmos. Modelo computacional. Concepto de tiempo de ejecución. Notación "Big-Oh". Reglas generales para el cálculo del tiempo de ejecución.</li> <li>• Polimorfismo y binding dinámico en java.</li> <li>• Resolución del TP 4 – Árboles Generales</li> </ul>	
Semana 8	<ul style="list-style-type: none"> <li>• Cálculo en algoritmos iterativos y recursivos. Ejemplo: Subsecuencia de suma máxima</li> </ul>	
Semana 9		Primera Evaluación Parcial – Semana 07/05/2018
Semana 10	<ul style="list-style-type: none"> <li>• 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.</li> <li>• Definición en java de Grafos con listas y con Matriz de adyacencia. Operaciones básicas.</li> <li>• Resolución del TP 5 – Análisis de algoritmos</li> </ul>	
Semana 11	<ul style="list-style-type: none"> <li>• 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.</li> <li>• Recorridos de grafos básicos, modificaciones a dichos algoritmos en java.</li> </ul>	
Semana 12		Recuperatorio de la Primera Evaluación Parcial – Semana



UNIVERSIDAD NACIONAL DE LA PLATA  
FACULTAD DE INFORMÁTICA

		28/05/2017
Semana 13	<ul style="list-style-type: none"><li>• Ordenamiento topológico. Definición. Ejemplos. Distintas implementaciones.</li><li>• Análisis del tiempo de ejecución de los algoritmos mencionados.</li><li>• Resolución del TP 6 – Grafos 1era Parte</li></ul>	
Semana 14	<ul style="list-style-type: none"><li>• 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<ul style="list-style-type: none"><li>- Algoritmo de Dijkstra (versión original)</li></ul></li></ul>	
Semana 15	<ul style="list-style-type: none"><li>• Problema del camino mínimo (continuación):<ul style="list-style-type: none"><li>- Algoritmo de Dijkstra (implementado con heap)</li><li>- Caminos mínimos desde un origen para: grafos con pesos positivos y negativos y grafos dirigidos y acíclicos</li><li>- Caminos mínimos entre cada par de vértices. Algoritmo de Floyd.</li></ul></li><li>• Resolución del TP 7 – Grafos 2da Parte</li></ul>	Segunda Evaluación Parcial – Semana 18/06/2018
Semana 16	<ul style="list-style-type: none"><li>• Árbol generador mínimo. Definición. Algoritmos de Prim y Kruskal. Análisis del tiempo de ejecución de los algoritmos vistos.</li></ul>	
Semana 17	<ul style="list-style-type: none"><li>• Consultas</li></ul>	Recuperatorio de la segunda evaluación parcial – Semana 02/07/2018
Semana 18	<ul style="list-style-type: none"><li>• Muestra y Consultas</li></ul>	
Receso Invernal		
Semana 19	<ul style="list-style-type: none"><li>• Consultas</li></ul>	
Semana 20	<ul style="list-style-type: none"><li>• Consultas</li></ul>	Recuperatorio flotante – Semana 06/08/2018



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

**Contacto de la cátedra (mail, página, plataforma virtual de gestión de cursos):**

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

Firmas del/los profesores responsables:



UNIVERSIDAD NACIONAL DE LA PLATA  
FACULTAD DE INFORMÁTICA

---

**ALGORITMOS Y ESTRUCTURAS DE  
DATOS**  
**Redictado**

**Año 2018**

**Carrera/ Plan:**

*Licenciatura en Informática*  
Plan 2003-07/2012/2015  
*Licenciatura en Sistemas*  
Plan 2003-07/2012/2015  
*Analista Programador Universitario*  
Plan 2007/2015  
*Analista en TIC Plan 2017*

**Año:** 2°

**Régimen de Cursada:** *Semestral*

**Carácter:** *Obligatoria*

**Correlativas:** *Matemática 2 –  
Algoritmos, Datos y Programas*

**Coordinador:** *Javier Díaz*

**Profesores:** *Alejandra Schiavoni –  
Catalina Mostaccio -  
Claudia Queiruga –  
Pablo Iuliano*

**Hs. semanales:** 6 hs.

---

**FUNDAMENTACIÓN**

Esta materia es de gran importancia dentro de las carreras, 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:

- 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;
- 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.

**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.

**CONTENIDOS MINIMOS:**

- Estructuras de Datos no lineales



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

- 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, Arboles 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. Árboles binarios de búsqueda. Árboles binarios de búsqueda balanceados: Árboles AVL. Recorridos ordenados (InOrden, PostOrden, PreOrden). Búsquedas. Actualización: inserción y borrado. Análisis de 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  $O()$ ,  $\Omega$ ,  $\Theta$ . 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.



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

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 OBLIGATORIA**

Título	Autor(es)	Editorial	Año de edición
1. <i>Data Structures And Algorithm Analysis in Java; 3rd Edition</i>	Mark Allen Weiss	Addison-Wesley	2012
2. <i>Data Structures and Algorithms</i>	A. Aho, J. Hopcroft, J. D. Ullman	Addison-Wesley	1983
3. <i>Thinking in Java, fourth edition.</i>	Bruce Eckel,	Prentice Hall,	2006
4.			

**BIBLIOGRAFÍA COMPLEMENTARIA**

Título	Autor(es)	Editorial	Año de edición
5. <i>Data Structures in Java; 1st Edition</i>	Thomas A. Standish	Addison-Wesley	1997
6.			
7. <i>Data Structures and Problem Solving using Java; 4th Edition.</i>	Mark Allen Weiss	Addison-Wesley	2009
8. <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, en las que se dictan el contenido referido a las estructuras de datos mencionadas y el 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.

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 donde los ejercicios se resuelven en lenguaje Java usando un ambiente de desarrollo apropiado. Los



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

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.

Los contenidos teóricos y prácticos están estrechamente vinculados, siendo éstos últimos una aplicación directa de los temas teóricos impartidos.

Las clases prácticas se dictan en diferentes turnos. Los alumnos eligen libremente al comienzo de la cursada el turno de práctica al que van a asistir, sin embargo, tienen la posibilidad de asistir eventualmente a otro turno de práctica ante la imposibilidad de hacerlo en su horario.

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.

## **EVALUACIÓN**

La asignatura cuenta con un régimen de promoción 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, y el alumno debe contar con un porcentaje establecido por la cátedra para acceder al régimen de promoción. Además, se implementarán entregas de ejercicios en fechas preestablecidas.

La evaluación de la asignatura consiste en dos parciales teórico-prácticos con puntaje ponderado, donde se plantean ejercicios a resolver similares a los trabajados en los prácticos y preguntas sobre conceptos teóricos relativos a los temas de la currícula. La aprobación de ambos parciales con un promedio de 6 o superior, le otorga al alumno la promoción de la materia, y la obtención de una nota entre 4 y 6 puntos, le otorga la aprobación de los trabajos prácticos, debiendo el alumno rendir un examen final.

El examen final es teórico-práctico, una parte con enunciados de opción múltiple y otra con preguntas de respuesta breve.

## **CRONOGRAMA DE CLASES Y EVALUACIONES**

<b>Clase</b>	<b>Contenidos/Actividades</b>	<b>Evaluaciones previstas</b>
Semana 1 Comienzo Semana	<ul style="list-style-type: none"><li>• La Plataforma Java.</li><li>• Revisión de Fundamentos de Programación Orientada a Objetos.</li></ul>	



UNIVERSIDAD NACIONAL DE LA PLATA  
FACULTAD DE INFORMÁTICA

20/08/2018	<ul style="list-style-type: none"><li>• Aplicación: Ejecución de programas java con y sin IDE (<i>Integrated Development Environment</i>).</li></ul>	
Semana 2	<ul style="list-style-type: none"><li>• Clases y objetos en Java</li><li>• Herencia y clases abstractas. Ejemplo de listas con Herencia en java.</li><li>• Tipos de datos genéricos. Ejemplos.</li><li>• Resolución del TP 1 – JDK y uso de Eclipse. Introducción al lenguaje Java</li></ul>	
Semana 3	<ul style="list-style-type: none"><li>• Repaso de conceptos de Listas, definición de la estructura y operaciones. Pilas y Colas. Listas con objetos.</li><li>• Noción intuitiva de análisis de algoritmos y complejidad.</li><li>• Árboles binarios: representaciones, recorridos.</li><li>• Constructores en java. Constructores y herencia.</li></ul>	
Semana 4	<ul style="list-style-type: none"><li>• Aplicaciones de árboles binarios: árboles de expresión.</li><li>• Árboles generales: ejemplos y terminología. Distintas representaciones e implementaciones. Recorridos. Aplicaciones.</li><li>• Interfaces en java, la interface comparable. Ejemplos de uso de esta interface en estructuras ordenadas.</li><li>• Resolución del TP 2 – Encapsulamiento y Abstracción</li></ul>	
Semana 5	<ul style="list-style-type: none"><li>• Cola de prioridades: concepto y características. Heap binaria: propiedades e implementación. Operaciones de acceso y construcción.</li><li>• Aplicaciones de Heap: Selección y Ordenación (HeapSort). Análisis de la eficiencia.</li><li>• Repaso de Interfaces. Ejemplos con Heap.</li><li>• Resolución del TP 3 – Árboles Binarios</li></ul>	
Semana 6	<ul style="list-style-type: none"><li>• Árboles binarios de búsqueda. Repaso de las operaciones. Árboles</li></ul>	



UNIVERSIDAD NACIONAL DE LA PLATA  
FACULTAD DE INFORMÁTICA

	<p>AVL: definición y representación. Concepto de Balanceo.</p> <ul style="list-style-type: none"><li>• Árboles AVL: implementación de las operaciones. Rotaciones simples y dobles.</li><li>• Paquetes y especificadores de acceso. Ejemplos de especificadores de acceso con árboles generales y árboles binarios de búsqueda.</li></ul>	
Semana 7	<ul style="list-style-type: none"><li>• Análisis de algoritmos. Modelo computacional. Concepto de tiempo de ejecución. Notación "Big-Oh". Reglas generales para el cálculo del tiempo de ejecución.</li><li>• Polimorfismo y binding dinámico en java.</li><li>• Resolución del TP 4 – Árboles Generales</li></ul>	
Semana 8	<ul style="list-style-type: none"><li>• Cálculo en algoritmos iterativos y recursivos. Ejemplo: Subsecuencia de suma máxima</li></ul>	
Semana 9		Primera Evaluación Parcial – Semana 15/10/2018
Semana 10	<ul style="list-style-type: none"><li>• 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.</li><li>• Definición en java de Grafos con listas y con Matriz de adyacencia. Operaciones básicas.</li><li>• Resolución del TP 5 – Análisis de algoritmos</li></ul>	
Semana 11		Recuperatorio de la Primera Evaluación Parcial – Semana 29/10/2018
Semana 12	<ul style="list-style-type: none"><li>• Algoritmos de recorrido DFS y BFS. Árbol generador DFS: en grafos dirigidos y no dirigidos. Determinación de componentes conexas y fuertemente conexas. Análisis del</li></ul>	



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

	<p>tiempo de ejecución de los algoritmos mencionados.</p> <ul style="list-style-type: none"> <li>• Recorridos de grafos básicos, modificaciones a dichos algoritmos en java.</li> </ul>	
Semana 13	<ul style="list-style-type: none"> <li>• Ordenamiento topológico. Definición. Ejemplos. Distintas implementaciones.</li> <li>• Análisis del tiempo de ejecución de los algoritmos mencionados.</li> <li>• Resolución del TP 6 – Grafos 1era Parte</li> </ul>	
Semana 14	<ul style="list-style-type: none"> <li>• 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             <ul style="list-style-type: none"> <li>- Algoritmo de Dijkstra (versión original)</li> </ul> </li> </ul>	
Semana 15	<ul style="list-style-type: none"> <li>• Problema del camino mínimo (continuación):             <ul style="list-style-type: none"> <li>- Algoritmo de Dijkstra (implementado con heap)</li> <li>- Caminos mínimos desde un origen para: grafos con pesos positivos y negativos y grafos dirigidos y acíclicos</li> <li>- Caminos mínimos entre cada par de vértices. Algoritmo de Floyd.</li> </ul> </li> <li>• Resolución del TP 7 – Grafos 2da Parte</li> </ul>	Segunda Evaluación Parcial – Semana 26/11/2018
Semana 16	<ul style="list-style-type: none"> <li>• Árbol generador mínimo. Definición. Algoritmos de Prim y Kruskal. Análisis del tiempo de ejecución de los algoritmos vistos.</li> </ul>	
Semana 17	<ul style="list-style-type: none"> <li>• Consultas</li> </ul>	Recuperatorio de la segunda evaluación parcial – Semana 10/12/2018
Semana 18	<ul style="list-style-type: none"> <li>• Muestra y Consultas</li> </ul>	
Semana 19	<ul style="list-style-type: none"> <li>• Consultas</li> </ul>	
Receso		



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

Semana 20	• Consultas	Recuperatorio flotante – Semana 04/02/2019
-----------	-------------	---

**Contacto de la cátedra (mail, página, plataforma virtual de gestión de cursos):**

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

Firmas del/los profesores responsables: