



Programa de la Asignatura:

# Algoritmos y Programación II



Código: 06

Carrera: **Ingeniería en Computación** Plan: **2013** Carácter: **Obligatoria**  
Unidad Académica: **Secretaría Académica** Curso: Primer Año – Segundo Cuatrimestre  
Departamento: **Ingeniería** Carga horaria total: **128 hs.** Carga horaria semanal: **8 hs.**  
Resolución de Problemas de Ingeniería: **40 %** Formación teórica: **50 %**  
Actividades de Proyecto y Diseño: **00 %** Formación experimental: **10 %**  
Práctica Supervisada: **00 %**

## Materias Correlativas Obligatorias

- 01 Algoritmos y Programación I

## Cuerpo Docente

Lic. Rosita Wachenchauzer  
Ing. Lucas Videla

## Índice

- Fundamentación pág. 2
- Encuadre y articulación de la asignatura pág. 2
  - Encuadre dentro del Plan de Estudios pág. 2
  - Articulación Horizontal pág. 2
  - Articulación Vertical pág. 2
- Objetivos pág. 2
  - Objetivo General pág. 2
  - Objetivos Específicos pág. 2
- Contenidos mínimos pág. 3
- Programa analítico pág. 3
- Bibliografía básica pág. 4
- Bibliografía de consulta pág. 4
- Metodología del aprendizaje pág. 4
  - Desarrollo de la asignatura pág. 4
  - Dinámica del dictado de las clases pág. 4
  - Trabajos prácticos pág. 5
- Metodología de evaluación pág. 5
- Planificación pág. 5
- Información de versiones pág. 6

AÑO ACADÉMICO 2013

ÚLTIMA REVISIÓN 02/07/2013

Firma Docente

Firma Coordinador

## 1. FUNDAMENTACION

La asignatura está incluida entre las materias que brindará a los alumnos los conocimientos necesarios para poder desarrollar soluciones programáticas, utilizando un lenguaje de programación de alto nivel para delinear la forma de la resolución de los problemas planteados.

El avance del mercado y del estado del arte ha hecho indispensable el conocimiento por parte del alumno de métodos para programar computadoras, comunicándole instrucciones que luego se convertirán en algoritmos, y más tarde en completos sistemas que solucionarán los problemas humanos que tienen que ver con el manejo de la información, su obtención, tratamiento y distribución, entre otros. Es por ello que deben proporcionarse conocimientos amplios sobre el uso, el análisis y la construcción de las estructuras de datos complejas y los criterios para su evaluación y aplicación, de tal modo que se puedan verter conceptos no solamente académicos sino también presentes en la industria. Así se ha optado por tomar el enfoque que la materia presenta, articulado con las materias anteriores y posteriores.

Siguiendo con el planteo de Algoritmos y Programación I, se aplica el Currículum Invertido de Meyer, en este caso al comenzar con el uso, continuar con el análisis y finalizar con la construcción de las estructuras dinámicas de datos.

## 2. ENCUADRE Y ARTICULACIÓN DE LA ASIGNATURA

La asignatura está ubicada en la currícula de la carrera en la parte inicial de la misma que corresponde al "Ciclo Profesional".

En esa etapa, los alumnos comenzarán a adquirir experiencia en el área de la programación.

Requieren nociones de los conceptos de lógica.

Esta materia tiene como correlativa a Algoritmos y Programación I.

## 3. Articulación Horizontal

En cuanto a la articulación horizontal se ha planeado que esta asignatura se curse en simultáneo con Álgebra I, Análisis Matemático I y Física I, de modo que el alumno pueda integrar una perspectiva de usos inmediatos y aplicaciones de los conocimientos que va desarrollando, pudiendo programar sus propias soluciones a problemas de las otras asignaturas.

## 4. Articulación Vertical

Esta asignatura brinda las bases para comenzar a plasmar soluciones de programación en forma de software, por lo que verticalmente no se articula con ninguna asignatura previa.

A esta materia le seguirá en un curso superior la asignatura Algoritmos y Programación III que proveerá al alumno manejo de lenguajes y soluciones de bajo nivel, incluyendo manejo de punteros.

## 5. OBJETIVOS

### Objetivo General

La cátedra se ha fijado como "objetivo cognoscitivo" de esta materia:

***"Lograr que los alumnos conozcan, usen y construyan múltiples estructuras dinámicas de datos, tanto simples como complejas; adicionalmente se espera que desarrollen criterios para analizar comparativamente y evaluar la eficiencia temporal y espacial de diferentes alternativas de implementación de las estructuras y los algoritmos que las manipulan".***

### Objetivos Específicos

Luego de cursar esta asignatura el alumno podrá:

- Usar estructuras dinámicas de datos.
- Implementar estructuras dinámicas de datos.
- Analizar las ventajas y desventajas, fortalezas y debilidades de las múltiples estructuras dinámicas de datos y sus implementaciones.
- Analizar y desarrollar algoritmos recursivos.
- Analizar la complejidad de algoritmos y estructuras de datos

## 6. CONTENIDOS MÍNIMOS

Uso de Estructuras de datos dinámicas y estáticas existentes. Uso de Listas, Pilas, Colas, Diccionarios. Iteradores. Construcción de estructuras de datos estáticas y dinámicas. Introducción al concepto de Orden de Complejidad Computacional. Notación  $O()$ . Algoritmos de Ordenamiento. Algoritmos de Búsqueda. Búsqueda lineal y binaria. Recursividad. Algoritmos recursivos. Estructuras Recursivas, Grafos y árboles. Construcción de estructura de datos. Diseño de interfaces de estructuras. Implementación de estructuras. Tablas de hashing. Concepto de herencia. Herencia de implementación. Polimorfismo. Clases abstractas. Redefinición. Ligadura tardía. Herencia de interfaces: Interfaces. Conceptos de reutilización. Métodos de reutilización. Patrones de diseño y expresiones del lenguaje (idioms). Manejo avanzado de excepciones. Recolección de basura.

## 7. PROGRAMA ANALÍTICO

### Unidad 1: Referencias

Nociones básicas sobre objetos y referencias. Recolección de basura. Manejo de excepciones. Entrada y salida. Archivos binarios.

### Unidad 2: Programación Orientada a Objetos

Objetos y clases. Herencia de comportamiento y de tipos. Clases abstractas, interfaces, métodos abstractos. Redefinición y sobrecarga.

### Unidad 3: Análisis de algoritmos

Orden de complejidad de los algoritmos. Notación  $O()$ . Problema de la búsqueda estática: búsqueda secuencial, binaria, interpolada.

### Unidad 4: Uso de estructuras de datos dinámicas y estáticas existentes

Uso de listas, pilas, colas, diccionarios. Iteradores.

### Unidad 5: Recursividad

Recursividad. Resolución de problemas por división y conquista. Algoritmos de ordenamiento: concepto de orden, clave, características; clasificación: externos e internos, simples y avanzados.

### **Unidad 6: Construcción de estructuras dinámicas de datos**

Polimorfismo de clases y métodos. Ligadura tardía (dynamic binding). Herencia múltiple. Aplicaciones avanzadas de TDA. Polimorfismo en implementaciones. Construcción de implementaciones de estructuras dinámicas de datos: pilas, colas, listas, árboles.

### **Unidad 7: Representaciones de conjuntos y diccionarios**

Elección de la representación según el contexto de uso. Representación de conjuntos mediante mapas de bits, tablas de hashing y árboles binarios de búsqueda. Representación de colas de prioridad mediante parvas (heaps). Una aplicación avanzada de heaps: Heapsort.

### **Unidad 8: Grafos**

Grafos. Representación. Recorridos. Problemas de camino mínimo y de árbol de tendido mínimo en grafos.

## **8. BIBLIOGRAFÍA BÁSICA**

- WEISS, MARK ALLEN, ESTRUCTURAS DE DATOS EN JAVA, ADDISON-WESLEY, 2000.
- MEYER , BERTRAND, Construcción de Software Orientada a Objetos, Prentice-Hall, 1985, 2da. Edición 1997.
- AHO, HOPCROFT, ULLMAN, Estructura de datos y algoritmos, Addison-Wesley, 1988
- GUEREQUETA, VALLECITO, Técnicas de diseño de algoritmo, Servicio de Publicaciones de la Universidad de Málaga, 1998

## **9. BIBLIOGRAFÍA DE CONSULTA**

- ESAKOV, JEFFREY, WEISS, TOM, Data structures, An advanced approach using C, Prentice-Hall, 1989
- ARNOLD, GOSLING, HOLMES, Holmes, El lenguaje de programación Java, Addison-Wesley, 3ra. Edición 2002.
- KNUTH, DONALD, The Art of Computer Programming, volume 3, Sorting and Searching, 1973

## **10. METODOLOGÍA DEL APRENDIZAJE**

### **8.a DESARROLLO DE LA ASIGNATURA**

Inicialmente, se tratará de familiarizar al alumno con el uso de diferentes estructuras dinámicas de datos en la resolución de problemas. El objetivo en esta instancia es facilitar la comprensión de las características intrínsecas de cada una de las estructuras de datos presentadas, a partir de su aplicación ante situaciones problemática con creciente nivel de complejidad.

En segundo lugar, se desarrollarán los conceptos que permitan comprender, analizar y construir algoritmos y estructuras recursivas.

Complementariamente, se presentarán las herramientas y técnicas de análisis de eficiencia computacional provistas por la complejidad algorítmica.

Posteriormente, se desarrollarán las múltiples implementaciones de las diferentes estructuras de datos, a partir de los conceptos teóricos presentados y experimentalmente desarrollados en la primera parte del curso. En esta instancia, se hará hincapié en el análisis comparativo de implementaciones alternativas, para posibilitar la formación de criterios que permitan al alumno seleccionar una implementación en función al contexto de aplicación, en el marco de la definición de un Tipo de Dato Abstracto (TDA). La separación estricta entre interfaz (provista formalmente por un TDA) e implementación conducirá a la utilización polimórfica de las diferentes estructuras de datos presentadas a lo largo del curso.

Finalmente, se estudiarán estructuras de datos más complejas, como árboles y grafos; sus características, aplicaciones, algoritmos e implementaciones.

### **8.b DINÁMICA DEL DICTADO DE LAS CLASES**

Para poder llevar adelante los objetivos de la cátedra, se hace un gran énfasis hacia la carga horaria de las aproximaciones a problemas, dando tanto peso a éstas como a la teoría.

El Profesor a cargo del curso se ocupará en forma personal y semanal del dictado de aquellos temas con un fuerte contenido teórico y que significan conceptos básicos y poco volátiles en la especialidad. Procederá a describir técnicas, características y pondrá ejemplos. Éste generará un ámbito de reflexión y discusión de los temas presentados, para que mediante la intervención de los alumnos, se puedan aclarar aquellos aspectos que el docente puede captar a través de las consultas recibidas, como los que han resultado de más compleja comprensión. También deberá discutir las distintas soluciones que se presentan en muchos casos, y mostrar ventajas y desventajas.

Habrà una parte de la clase dedicada a la aplicación de los conceptos vistos durante la parte teórica. Se desarrollarán problemas con creciente nivel de dificultad.

### **8.c TRABAJOS PRÁCTICOS**

Habrà dos trabajos prácticos durante el cuatrimestre, los cuales tendrán como objetivo resolver un problema algorítmico novedoso, para el cual los alumnos deberán profundizar sobre los conceptos vistos.

Los mismos serán de consigna variante según el año en curso, lo cual ayuda a darle flexibilidad a la cátedra.

## **11. METODOLOGÍA DE EVALUACIÓN**

### **9.a NORMAS DE EVALUACIÓN.**

- El criterio es que la evaluación del alumno es permanente.
- Se tomarán dos exámenes parciales teórico/prácticos pudiendo acceder a un recuperatorio.
- Las notas de los parciales representan los resultados de la evaluación teórico/práctica.
- Los exámenes parciales y sus recuperatorios serán escritos.

### **9.b RÉGIMEN DE APROBACIÓN DE LA MATERIA.**

- Para la aprobación de la materia los alumnos deberán tener los dos parciales aprobados, teniendo la posibilidad de recuperar cada uno de ellos en una única oportunidad adicional, en la fecha acordada con los docentes.
- Además los alumnos deberán aprobar los trabajos prácticos, como condición para la aprobación de la materia.
- Los alumnos que obtengan una nota inferior a cuatro puntos se les asignará la nota insuficiente y deberán recursar la materia.

## 12. PLANIFICACIÓN

CALENDARIO DE CLASES Y EVALUACIONES	
Semana 1	Unidad 1
Semana 2	Unidad 2
Semana 3	Unidad 3
Semana 4	Unidad 4
Semana 5	Unidad 5
Semana 6	Unidad 6
Semana 7	Unidad 6
Semana 8	Primer Parcial
Semana 9	Unidad 7
Semana 10	Unidad 7
Semana 11	Unidad 7
Semana 12	Unidad 8
Semana 13	Unidad 8
Semana 14	Segundo Parcial
Semana 15	Repaso
Semana 16	Recuperatorios
Del al de	FINAL

Información de Versiones	
Nombre del Documento:	Ficha Académica de la asignatura
Nombre del Archivo	Plan 2013 – Algoritmos y Programación II.docx
Documento origen:	
Elaborado por:	Diego Fontdevila y Rosita Wachenchauzer

Revisado por:	Alejandro Oliveros
Aprobado por:	Alejandro Oliveros
Fecha de Elaboración:	02/07/2013
Fecha de Revisión:	03/07/2013
Fecha de aprobación	03/07/2013
Versión:	1.2