



## TALLER DE PROGRAMACION

Año 2017

### Carrera/ Plan:

*Licenciatura en Informática*

Plan 2015

*Licenciatura en Sistemas*

Plan 2015

*Analista Programador Universitario*

Plan 2007

Año: 1°

Régimen de Cursada: *Cuatrimestral*

Carácter: Obligatoria

Correlativas: Conceptos de Algoritmos, Datos y Programas

Coordinador: *Armando De Giusti*

Profesores: *Cristina Madoz, Gladys*

*Gorga, Laura De Giusti, Virginia*

*Ainchil, Waldo Hasperué, Victoria*

*Sanz, Ismael Rodriguez*

Hs. semanales: 6 (teórico-práctico)

---

### FUNDAMENTACIÓN

Se trabaja a partir de diferentes situaciones problemáticas de la vida real y se abordan las estrategias de solución bajo criterios de calidad, eficiencia y corrección, para finalmente llegar a una implementación acorde a las especificaciones planteadas. Todos los aspectos mencionados son abordados a partir del trabajo con los diferentes paradigmas de programación.

Los conceptos abordados en la asignatura permitirán al alumno familiarizarse con todos estos aspectos vinculados a la tarea de la programación básica ya que se constituyen en uno de los conceptos básicos de la disciplina.

### Objetivos Generales:

- Realizar desarrollo de programas simples en el paradigma imperativo.
- Extender el manejo de datos a datos no lineales (Arboles).
- Introducción de los conceptos básicos de un segundo paradigma de programación (orientación a objetos) con énfasis en la noción de reusabilidad.
- Desarrollo de programas simples en un lenguaje orientado a objetos.
- Introducción de los conceptos básicos de la Programación Concurrente
- Desarrollo de programas simples con un lenguaje de programación concurrente que permita interpretar los conceptos de comunicación y sincronización entre procesos.
- Combinar los elementos estudiados previamente en Conceptos de Algoritmos, Datos y programas con las tareas experimentales en diferentes lenguajes de programación, a fin de que el alumno complete el ciclo del problema a su solución con computadora.

### Contenidos Mínimos

- Estructuras de datos no lineales. Arboles.
- Conceptos de programación imperativa.



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

---

- Desarrollo de programas en un lenguaje imperativo.
- Implementación de algoritmos fundamentales (búsqueda, ordenación).
- Conceptos de programación orientada a objetos.
- Análisis de las características fundamentales de un lenguaje orientado a objetos.
- Desarrollo de programas en un lenguaje orientado a objetos.
- Conceptos básicos de concurrencia y paralelismo.
- Procesos. Comunicación y sincronización entre procesos.
- Desarrollo de programas concurrentes/paralelos.

### Programa

#### **A. Programación Estructurada**

Estructuras de datos no lineales.

Tipo de dato Arboles. Definición y terminología asociada. Características. Operaciones.

Implementación de algoritmos fundamentales sobre estructuras de datos estáticas y dinámicas: búsquedas, ordenación, merge.

Desarrollo de programas en un lenguaje imperativo (Pascal).

#### **B. Programación orientada a objetos**

Introducción a la POO.

Concepto de Objeto (estado y comportamiento), Clase e Instancia, Constructores.

Concepto de Herencia.

Desarrollo de programas simples en un lenguaje orientado a objetos (Java)

#### **C. Programación Concurrente**

Conceptos básicos de concurrencia y paralelismo.

Procesos. Comunicación y sincronización entre procesos.

Desarrollo de programas concurrentes/paralelos utilizando el ambiente del multirobot (r-info).

### Bibliografía Básica

#### **Algoritmos, datos y programas con aplicaciones en Pascal, Delphi y Visual Da Vinci.**

De Giusti, Armando et al. 1er edición. Prentice Hall 2001.

#### **Estructuras de Datos y Algoritmos.**

Hernández R., Dormido R., Lazaro J. Ros S. Pearson Education. 2000.

#### **Introduction to algorithms**

Comen, Leiserson. MIT Press 2001.

#### **Estructuras de Datos y Algoritmos.**

Aho Alfred, Hopcroft John y Ullman Jeffrey. Addison Wesley Publishing Company. EUA. 1998.

#### **Programación en Pascal**

Joyanes Aguilar, Luis. Mc Graw Hill. 2006

#### **Fundamentos de Programación. Libro de Problemas.**

Joyanes Aguilar L., Fernandez M., Rodríguez L. Mc Graw Hill. 2003.

#### **Data structures, algorithms and software principles.**

Standish, T. A. Addison Wesley Publishing Company. 1994.

#### **Estructuras de Datos y Algoritmos**

Weiss, M.A. Addison Wesley. 1995.



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

---

**Fundamentos de Programación.**

Joyanes Aguilar L., Fernandez M., Rodríguez L. Mc Graw Hill. 1999.

**Algoritmos y estructuras de datos y programación orientada a objetos.**

Flórez Rueda. Ecoe Ediciones. Bogotá. 2005. ISBN 958648394/0

**Programación En C Metodología, Algoritmos Y Estructura De Datos.**

Joyanes Aguilar Luis – Zahonero Martínez. Segunda Edición –Editorial Mc Graw Hill. España - Edición 2007

**An introduction to object-oriented programming**

Timothy Budd. Addison Wesley. 2002.

**Programación orientada a objetos con Java**

David J. Barnes, Michael Kolling. Tercera Edición. Pearson Educación, 2007 Bruce Eckel: Thinking in Java. Cuarta Edición. Prentice Hall, 2006.

**Thinking in C++.**

Bruce Eckel. Segunda Edición. Prentice Hall, 2000.

**Data Structures and Algorithm Analysis in Java**

Mark Allen Weiss. Tercera Edición. Addison-Wesley. 2012

**Bibliografía Adicional**

**Introduction to Computer Science with applications in Pascal.**

Garland, S.J. Addison Wesley Publishing Company. 1986.

**Estructuras de Datos.**

Franch Gutierrez, Xavier. Alfaomega Grupo Editor Argentino.2002

**Estructura de Datos.**

Joyanes Aguilar C., Zahonero Martinez I. Mc Graw Hill. 1998.

**Estructuras de Datos. Libro de Problemas.**

Joyanes Aguilar L., Fernandez M., Rodríguez L. Mc Graw Hill. 1999.

**Estructuras de Datos.**

Lipschutz, S. Mc Graw Hill. 1997.

**Programación estructurada en Turbo Pascal 7.**

Lopez Roman, L. Alfaomega Grupo Editor Argentino. 1998.

**Estructuras de Datos.**

Martinez Román, Quiroga Elda. Thomson International. 2002

**Estructura de Datos y Algoritmos.**

Sisa, Alberto Jaime. Editorial Prentice. 2002.

**Pascal Estructurado.**

Tremblay, Jean Paul. Mc Graw Hill.1980.

**Data structures, algorithms and performance.**

Wood, D. Addison Wesley Publishing Company. 1993.



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

---

**Structures and Algorithm Analysis in Java**

Weiss, M.A. Data, 3rd Edition, Pearson/Addison Wesley, 2011

**Data Structures and Algorithms using C#.**

M. McMillan. Cambridge University Press, 2006

**Sitios de interés:**

<http://csunplugged.org>

<http://www.eduteka.org>



## Metodología de Enseñanza

El curso Taller de Programación se divide en 3 módulos: Programación Imperativa, Programación Orientada a Objetos y Programación Concurrente.

Cada módulo del Taller tiene una duración estimada de 8 clases con una carga semanal de 2 clases de 3 hs. cada una. Cada clase consta de contenidos teórico-prácticos con actividades en máquina para resolver en el aula y también fuera del horario de clase.

El taller se divide en diferentes turnos dependiendo de la cantidad de alumnos inscriptos, cubriendo horarios de mañana y tarde. Cada turno tiene asignado un aula y un horario determinados.

Cada turno está a cargo de un docente responsable del dictado de los contenidos teóricos-prácticos y cuenta con auxiliares docentes para las consultas de las actividades en máquina.

En cada turno, se conformarán equipos integrados por 2 alumnos para el trabajo en máquina. Cada equipo será responsable de una notebook que utilizará para el desarrollo de los ejercicios prácticos.

El material teórico y la ejercitación práctica utilizados en el curso están disponibles en el EVEA IDEAS.

### Asistencia a Clases

- La asistencia a las clases teórico-prácticas es obligatoria.
- En cada clase de Taller los alumnos tendrán presente, ausente, o ausente justificado. Los ausentes justificados no pasan a ser presentes.
- La asistencia a cada clase será tomada una única vez durante el horario de clase. Si un alumno no se encuentra en el aula por cualquier motivo, tendrá ausente.
- Pueden justificarse ausentes solamente por razones de salud, presentando certificado otorgado por Hospital Público.
- El certificado, para ser tenido en cuenta, debe ser entregado al docente encargado del aula INDEFECTIBLEMENTE la semana posterior a la reincorporación a clase, debiéndose respetar esta condición para que el certificado sea aceptado.

### Evaluaciones Breves en la clase

- Durante la clase se propone la realización de ejercicios prácticos que los alumnos deben resolver y enviar, por el entorno IDEAS, al auxiliar a su cargo. La evaluación de estas actividades servirá de información para los docentes y de orientación para el alumno. El rendimiento satisfactorio de los alumnos en estas pruebas será considerado, a favor del alumno, durante la instancia de evaluación final del Taller.

## Evaluación

### Aprobación de un módulo

- El alumno debe cumplir con el 80% de asistencia a las clases.
- Asistir los días correspondientes a la resolución del trabajo final (Partes I y II)
- Resolver un trabajo final (a definir) en equipo que se desarrolla durante el horario del Taller.
- Aprobación del trabajo final (en equipo)
- Aprobación del Coloquio (en forma individual)

Se otorgará la cursada del Taller a aquellos alumnos que hayan cumplido con el 80% de asistencia a cada uno de módulos y hayan aprobado al menos dos de los tres módulos del Taller.

En caso que el alumno tenga solo uno de los tres módulos aprobados y haya cumplido con la asistencia y la presentación del trabajo final en los otros dos módulos (módulos desaprobados), tendrá una UNICA instancia recuperatoria en la que se evaluarán temas correspondientes a esos módulos.

### Condiciones de Aprobación del Examen Final

- Aquellos alumnos que hayan aprobado los tres módulos en la primera instancia deberán inscribirse y presentarse en una mesa de examen final para dar por cumplida la última etapa del Taller. Para cumplir con este paso, tienen tiempo hasta el mes de julio de 2018 inclusive. Para acceder a esta última etapa, el alumno debe tener aprobados los exámenes finales de las materias correlativas.



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

**CRONOGRAMA DE CLASES Y EVALUACIONES**

Cada semana del curso incluye dos días de clases teórico-prácticas que se organizan de acuerdo al siguiente cronograma:

**Módulo Programación Imperativa (Pascal)**

<b>Clase</b>	<b>Contenidos</b>	<b>Actividades</b>
Clase 1 (Imperativo)	Presentación del Taller. Ordenación en Arreglos. Método de ordenación por intercambio. Operaciones con Matrices.	Implementación de la operación de Ordenación por intercambio en Pascal. Resolución de ejercicios de algorítmica básica con matrices en Pascal.
Clase 2 (Imperativo)	Operación de Merge en Listas y Vectores.	Implementación de la operación de Merge de dos listas en Pascal. Implementación de la operación de Merge de dos vectores.
Clase 3 (Imperativo)	Recursión. Concepto. Características. Ejercitación.	Resolución de ejercicios básicos utilizando recursión.
Clase 4 (Imperativo)	Arboles Binarios Ordenados. Concepto. Operaciones. Ejercitación.	Implementación de las operaciones básicas de árboles binarios ordenados en Pascal.
Clase 5 (Imperativo)	Arboles Binarios Ordenados. Ejercitación	
Clase 6 (Imperativo)	Resolución del Trabajo Final (Parte I)	
Clase 7 (Imperativo)	Resolución del Trabajo Final (Parte II)	
Clase 8 (Imperativo)	Coloquios	

**Módulo Programación Orientada a Objetos (Pascal)**

<b>Clase</b>	<b>Contenidos</b>	<b>Actividades</b>
Clase 1 (Objetos)	Conceptos básicos del lenguaje Java	Implementación de programas simples imperativos en Java para ejercitar la sintaxis.
Clase 2 (Objetos)	Introducción a la POO	Ejercitación que comprende instanciación de objetos y envío de mensajes.
Clase 3 (Objetos)	Conceptos básicos de POO utilizando Java.	Ejercitación que comprende programación de nuevas clases, instanciación de objetos de dichas clases, envío de mensajes a dichos objetos.
Clase 4 (Objetos)	Constructores.	Ejercitación que comprende la incorporación de constructores a las clases implementadas con anterioridad.
Clase 5 (Objetos)	El concepto de herencia.	Ejercitación con herencia.
Clase 6 (Objetos)	Resolución del Trabajo Final (Parte I)	
Clase 7 (Objetos)	Resolución del Trabajo Final (Parte II)	
Clase 8 (Objetos)	Coloquios	



**Módulo Programación Concurrente**

<b>Clase</b>	<b>Contenidos</b>	<b>Actividades</b>
Clase 1 (Concurrencia)	Conceptos básicos de Concurrencia	Ejercicios que muestran los problemas de concurrencia.
Clase 2 (Concurrencia)	Entorno R-info	Trabajo con el entorno R-info.
Clase 3 (Concurrencia)	Memoria distribuida	Trabajo con el entorno R-info aplicando los conceptos vistos a ejercicios con memoria distribuida.
Clase 4 (Concurrencia)	Memoria compartida	Trabajo con el entorno R-info aplicando los conceptos vistos a ejercicios con memoria compartida.
Clase 5 (Concurrencia)	Memoria distribuida y compartida	Trabajo con el entorno R-info aplicando los conceptos vistos a ejercicios con memoria distribuida y compartida.
Clase 6 (Concurrencia)	Resolución del Trabajo Final (Parte I)	
Clase 7 (Concurrencia)	Resolución del Trabajo Final (Parte II)	
Clase 8 (Concurrencia)	Coloquios	

FECHA RECUPERATORIA: miércoles 15 de noviembre de 2017 – 9:30 Hs. Aula 4 (A CONFIRMAR).

Contacto de la cátedra (plataforma virtual de gestión de cursos):

Plataforma virtual: [http:// ideas.info.unlp.edu.ar/login](http://ideas.info.unlp.edu.ar/login)