



Ministerio de Cultura y Educación  
 Universidad Nacional de San Luis  
 Facultad de Ciencias Físico Matemáticas y Naturales  
 Departamento: Informatica  
 Area: Area IV: Pr. y Met. de Des. del Soft.

(Programa del año 2015)  
 (Programa en trámite de aprobación)  
 (Presentado el 22/09/2015 23:56:39)

### I - Oferta Académica

Materia	Carrera	Plan	Año	Período
PROGRAMACION III	ING. INFORM.	026/1 2- 08/15	2015	2° cuatrimestre

### II - Equipo Docente

Docente	Función	Cargo	Dedicación
NECCO, CLAUDIA MONICA	Prof. Responsable	P.Adj Exc	40 Hs
SANCHEZ, ALEJANDRO	Responsable de Práctico	JTP Semi	20 Hs

### III - Características del Curso

Credito Horario Semanal				
Teórico/Práctico	Teóricas	Prácticas de Aula	Práct. de lab/ camp/ Resid/ PIP, etc.	Total
3 Hs	Hs	Hs	3 Hs	6 Hs

Tipificación	Periodo
B - Teoria con prácticas de aula y laboratorio	2° Cuatrimestre

Duración			
Desde	Hasta	Cantidad de Semanas	Cantidad de Horas
10/08/2015	20/11/2015	15	90

### IV - Fundamentación

Quando se construye la solución a un problema a través de un programa, es necesario utilizar el mejor paradigma de programación para la solución planteada. El conocimiento de distintos paradigmas de programación por parte del alumno es fundamental para encontrar la mejor solución a un problema dado. Conocer otros paradigmas de programación hasta los ahora vistos, es de vital importancia para el ingeniero en informática, entre los cuales están el paradigma de programación lógico y el paradigma de programación funcional. Conocer lenguajes que integran distintos paradigmas y programar en entornos Web son también conceptos necesarios para un ingeniero.

### V - Objetivos / Resultados de Aprendizaje

Utilizar correctamente las singularidades del Paradigma de Programación Lógica y desarrollar aplicaciones de hasta mediana complejidad en dicho enfoque. Aprovechar razonablemente las posibilidades del Paradigma de Programación Funcional y construir programas de mediano nivel de dificultades en este contexto. Reconocer las oportunidades de utilización de la Programación Multiparadigma utilizando correctamente sus dificultades. Desarrollar aplicaciones en entorno Web distinguiendo y aprovechando correctamente los conceptos y herramientas asociados.

### VI - Contenidos

Unidad I: Paradigma de Programación Lógica 1

Introducción. Variables, constantes y estructuras: Desarrollo de ejemplos. Átomos, literales y hechos: Ejemplos y ejercicios en clase. Reglas. Cláusulas y predicados: Ejemplos y ejercicios en clase.

Programas, Preguntas y Ejecución: Desarrollo de ejemplos. Semántica operacional y árboles lógicos: Ejemplos y ejercicios en clase. Unificación: Concepto, ejemplos y ejercicios en clase.

### **Unidad II: Paradigma de Programación Lógica 2**

Ejemplos de notación estándar en Prolog: Ejercicios en clase. Uso del interfaz de Prolog: Desarrollo mediante ejemplos en laboratorio. Aritmética: Ejemplos y ejercicios en clase. Factorial. Estructuras internas: functor y arg. Ejemplos de su utilización. Listas explotando Prolog: Desarrollo de ejercicios en laboratorio. Uso del 'cut'. Metapredicados: Ejemplos y ejercicios.

### **Unidad III: Paradigma de Programación Funcional 1**

Paradigma de Programación Funcional. Historia y evolución de los lenguajes funcionales. Descripción del modelo de programación funcional. Características principales de los lenguajes funcionales: transparencia referencial, alto orden y curriificación, y sistemas de tipos. Reducción y evaluación en programas funcionales. Ordenes de reducción: reducción aplicativa y reducción normal.

### **Unidad IV: Paradigma de Programación Funcional 2**

Introducción a Haskell. Expresiones. Declaraciones. Estructuras de datos. Tipos simples y tipos numéricos. Tipos funcionales. Definición de funciones. Implementación de soluciones de problemas en lenguaje Haskell en el laboratorio. Listas por comprensión. Definición y ejemplos. Listas como tipo inductivo. Funciones básicas sobre listas. Funciones de alto orden sobre listas. Patrón de recorrido: map. Patrón de selección: filter. Patrones de recursión: foldr y foldl. Polimorfismo. Tipos de Polimorfismo. Definición de Tipos de datos abstractos. Mecanismos de definición de tipos nuevos y de funciones sobre ellos. Tipos algebraicos.

### **Unidad V: Programación Multiparadigma 1**

Concepto de Programación Multiparadigma: Ejemplos. Características y propiedades. Biblioteca estándar de lenguajes multiparadigma. Tipos de datos, Cadenas de texto. Tuplas y listas. Conjuntos. Controles de flujo. Funciones. Clases. Módulos. Archivos. Lenguajes multiparadigma en la Web. Lenguajes multiparadigmas y persistencia de datos.

### **Unidad VI: Programación Multiparadigma 2**

Lenguajes que permiten combinar paradigmas que parecen irreconciliables.

Programación imperativa en lenguaje Python.

Programación funcional en lenguaje Python.

Programación orientada a objetos en lenguaje Python

### **Unidad VII: Programación en entorno Web 1**

Programación en entorno Web: Programación en la Web con lenguajes del tipo PHP. Uso de lenguajes del tipo PHP y Programación Orientada a Objetos. AJAX: Asynchronous JavaScript and XML. Lenguajes del tipo PHP y Web Services.

### **Unidad VIII: Programación en entorno Web 2**

Introducción a la programación en Internet con lenguajes multiparadigmas. Network Scripting en Python. Socket. Manejo de multiples clientes. Script del lado del cliente. Protocolos. Transferencia de Archivos. Distribución de información. Envío de mensajes básicos. Procesamiento de correo electrónico. Acceso a grupos de noticias. Acceso a sitios Web. Scripts del lado del Servidor. CGI Script. Ejecución de scripts del lado del servidor. Utilidades.

## **VII - Plan de Trabajos Prácticos**

Práctico 1: Programación Lógica en lenguaje Prolog (con laboratorio).

Práctico 2: Programación funcional en lenguaje Haskell (con laboratorio).

Práctico 3: Programación multiparadigma en Python (con laboratorio).

## VIII - Regimen de Aprobación

- Para regularizar la asignatura:

El alumno debe aprobar dos parciales o sus recuperaciones, y presentar en forma y tiempo los laboratorios solicitados por la cátedra.

El alumno debe asistir al 70% de las clases correspondientes al crédito horario.

- Para promocionar la asignatura:

El alumno debe cumplir con las condiciones de regularización con un nivel superior o igual al 70% del total y aprobar la sección especial de integración de cada parcial o su recuperación.

Se tomarán las recuperaciones que indica la OCS 32/14.

### EXAMEN REGULAR

Se tomará un examen escrito u oral en las mesas habilitadas del calendario académico

### EXAMEN LIBRE:

Según lo dispuesto por Art. 27 de Ord. 13/03 CS.

## IX - Bibliografía Básica

[1] Sterling, L., Shapiro, E., "The Art of Prolog", MIT Press, 1994 (2a edición).

[2] Apt, K, "From Logic Programming to Prolog", Prentice-Hall, 1997.

[3] Bratko, I., "Prolog Programming for Artificial Intelligence", Addison-Wesley, 1990.

[4] Lloyd, J., "Foundations of Logic Programming", Springer-Verlag, 1991 (2da Edición).

[5] [http://www.comptechdoc.org/os/linux/programming/script/linux\\_pgscriptintro.html](http://www.comptechdoc.org/os/linux/programming/script/linux_pgscriptintro.html)

[6] Python Programming Language - Official Website <http://www.python.org/doc/>

[7] Haskell 98 Language and Libraries. The Revised Report December 2002. <http://www.haskell.org/onlinereport/>

[8] Gentle Introduction To Haskell, version 98. Paul Hudak, John Peterson, Joseph Fasel, Revised June, 2000 by Reuben Thomas. <http://www.haskell.org/tutorial/>

[9] Razonando con Haskell. Un curso sobre programación funcional Blas C. Ruiz; Francisco Gutierrez; Pablo Guerrero; José E. Gallardo, Paraninfo. ISBN: 8497322770 ISBN-13: 9788497322775

[10] Sockets in Python: Into the World of Python Network Programming

<http://www.devshed.com/c/a/Python/Sockets-in-Python-Into-the-World-of-Python-Network-Programming/>

[11] Sockets in Python. <http://www.devshed.com/c/a/Python/Sockets-in-Python/>

[12] Thompson, S., "Haskell-The Craft of Functional Programming", Addison-Wesley, ISBN 0-201-34275-8, 2da Edición

[13] PHP Reference Book Blog <http://www.phpreferencebook.com/>

[14] Larry E. Ullman, PHP for the World Wide Web, Third Edition, Peachpit Press; 3 edition, 2008

## X - Bibliografía Complementaria

[1] Mark Lutz, Programming Python, O'Reilly Media, Inc.; 3 edition, 2006.

[2] Peter Norton, Alex Samuel, David Aitel, et.al., Beginning Python - Wiley Publishing, Inc.2005.

[3] Python para todos - Raúl González Duque licencia Creative Commons 2.5 <http://mundogeek.net/tutorial-python/>

[4] Harm, Daryl, The Quick Python Book Daryl D. Harm, Editor : Manning Publications

[5] Perry, George, The Life of Python Perry, Editor : Pavilion Books, 2007

[6] Learning Python - David Ascher, Mark Lutz O'Reilly - Second Edition. 2004.

[7] Python Cookbook - Alex Martelli and David Ascher O'Reilly - Second Edition 2005

[8] Burtch, Linux Shell Scripting with Bash, Editorial: Sams, 2004-01-29 | ISBN: 0672326426

[9] Randal Michael, Mastering UNIX Shell Scripting, Editorial Wiley, 2003

[10] Todd Meadors, "Linux Shell Script Programming", Course Technology; 1 edition, 2003, ISBN-10: 0619159200

[11] Haskell-The Craft of Functional Programming. Second Edition. Simon Thompson. Addison-Wesley, ISBN 0-201-34275-8

- [12] Linux Network Administrator's Guide, Third Edition O'Reilly
- [13] Internetworking with TCP/IP. Douglas Comer.
- [14] TCP/IP Illustrated Vol1 – Richard Stevens - Prentice Hall
- [15] <http://www.haskell.org/onlinereport/>
- [16] Nixon, Robin, "Learning PHP, MySQL, and JavaScript: A Step-By Step Guide to Creating Dynamic Websites (Animal Guide)", Barnes & Noble, 2009
- [17] David Powers, PHP Solutions: Dynamic Web Design Made Easy friends of ED; 3rd edition 2010
- [18] Practical PHP Programming <http://www.tuxradar.com/practicalphp>

## XI - Resumen de Objetivos

Al finalizar el curso se espera que el alumno sea capaz de:  
 Desarrollar aplicaciones en el Paradigma de Programación Lógica de programas de mediana complejidad. Aprovechar razonablemente las posibilidades del Paradigma de Programación Funcional y construir programas de mediano nivel de dificultades en este contexto. Reconocer las oportunidades de utilización de la Programación Multiparadigma utilizando correctamente sus dificultades. Desarrollar aplicaciones en entorno Web distinguiendo y aprovechando correctamente los conceptos y herramientas asociados.

## XII - Resumen del Programa

Paradigma de Programación Lógica.  
 Paradigma de Programación Funcional.  
 Programación Multiparadigma.  
 Programación en entorno Web.

## XIII - Imprevistos

--

## XIV - Otros

--

<b>ELEVACIÓN y APROBACIÓN DE ESTE PROGRAMA</b>	
	<b>Profesor Responsable</b>
Firma:	
Aclaración:	
Fecha:	