

Nombre de la Asignatura	Programación Funcional Avanzada
Créditos	12 Créditos
Objetivo de la Asignatura	El objetivo de este curso es introducir al estudiante en algunos de los últimos avances en la práctica de la programación funcional. Se presentarán varias técnicas avanzadas de programación funcional, patrones, librerías y herramientas.
Metodología de enseñanza	Se darán 40hs. de clases teórico-prácticas divididas en veinte clases de dos horas cada una, a razón de dos clases por semana.
Temario	<ol style="list-style-type: none"> 1. Desarrollo <ul style="list-style-type: none"> - Herramientas - Testeo - <i>Debugging</i> - <i>Profiling</i> 2. Librerías <ul style="list-style-type: none"> - Estructuras de Datos - Interfaz a Otros Lenguajes de Programación - Mónadas y Estructuras Relacionadas - Lenguajes de Dominio Específico Embebidos 3. Características y Extensiones del Lenguaje <ul style="list-style-type: none"> - <i>Type Classes Avanzadas</i> - <i>Kinds</i> - Tipos de Datos Algebraicos Generalizados - Tipos Existenciales - Polimorfismo de Alto Rango
Bibliografía	<ul style="list-style-type: none"> • Real World Haskell. Code You Can Believe In. Bryan O'Sullivan, Donald Bruce Stewart y John Goerzen. O'Reilly Media. ISBN-13: 978-0596514983. Noviembre 2008 • Introduction to Functional Programming using Haskell. Richard Bird. Prentice Hall. ISBN-13: 978-0134843469. Enero 1998 • Haskell 98 Language and Libraries: The Revised Report. Simon Peyton Jones (editor). Cambridge University Press. ISBN-13: 978-0521826143. Diciembre 2002 • Types and Programming Languages. Benjamin Pierce. MIT Press. ISBN-13: 978-0262162098. Febrero 2002 • Advanced Topics in Types and Programming Languages. Benjamin Pierce. MIT Press. ISBN-13: 978-0262162289. Noviembre 2004 • Fun of Programming. Jeremy Gibbons y Oege de Moor (editores). Palgrave Macmillan. ISBN-13: 978-0333992852. Marzo 2003 <p>En el curso se va a hacer referencia a artículos de investigación.</p>
Conocimientos previos exigidos y recomendados	Introducción a la Programación Funcional Lógica

Anexo:

1) Cronograma tentativo.

Tentativamente, el cronograma de actividades propuesto es:

- Semana 1: Introducción, Ambiente de Desarrollo
- Semana 2: Testeo, *Debugging*, Cálculo lambda y evaluación perezosa
- Semana 3: *Profiling*, etestructuras de datos
- Semana 4: Concurrencia, redes, FFI, GUI
- Semana 5: Mónadas y estructuras relacionadas
- Semana 6: Módulos y registros
- Semana 7: Presentaciones de estudiantes 1, EDSL
- Semana 8: *Type classes*, *GADTs*
- Semana 9: Polimorfismo de alto orden, *System F*
- Semana 10: Presentaciones de estudiantes 2, Temas avanzados

2) Modalidad del curso y procedimiento de evaluación.

Semanalmente se dictarán 4 horas de clase teórico-prácticas en las cuales se presentarán los conceptos y técnicas fundamentales. El estudiante deberá profundizar los temas haciendo uso de la bibliografía sugerida.

A lo largo del curso se marcarán ejercicios prácticos para realizar, algunos de los cuales serán obligatorios (y realizados en forma personal) con el objetivo de luego ser corregidos.

Se introducirá al estudiante en la actividad de investigación por medio de la lectura y presentación de artículos técnicos de revistas y conferencias. De esta forma se espera que el estudiante no sólo adquiera conocimientos en el área específica del curso sino también tenga una actitud crítica acerca de los desarrollos existentes en el tema.

Se recomienda al estudiante dedicar, en promedio, 10 horas semanales complementarias de estudio, incluyendo la realización de ejercicios prácticos y lectura de artículos.

En el curso habrá práctica de laboratorio. El laboratorio consistirá en la implementación de un sistema de mediano porte en Haskell haciendo uso de las herramientas y técnicas vistas en el curso. Se estima que el laboratorio requiera una dedicación aproximada de 30 horas.

Al finalizar el curso se realizará un examen de teórico.

3) Materia.

Programación.

4) Previaturas.

Para cursar esta asignatura es necesario tener aprobada la asignatura Introducción a la Programación Funcional.

5) Cupo

La cantidad y dedicación horaria de los docentes asignados al curso limitan el número de estudiantes que es posible atender para llevar adelante el curso adecuadamente. Teniendo en cuenta que participarán además estudiantes de postgrado se limita la cantidad de estudiantes de grado a un máximo de 30.

El cupo se define mediante una evaluación los méritos académicos, basados en escolaridad y avance en la carrera. Concretamente se definirá un orden de selección en función de una

suma ponderada del promedio de calificaciones y el porcentaje de créditos obtenidos con respecto al total requerido para la carrera.

6) Esta asignatura no adhiere a resolución del consejo sobre condición de libre

APROB. RES. CONSEJO DE FAC. ING.

de fecha 20.8.2009 Exp. 060120-002125-09