



Facultad de Ingeniería Comisión Académica de Posgrado

Formulario de Aprobación Curso de Posgrado 2014

Asignatura: Construcción Formal de Programas en Teoría de Tipos (CFPTT)

(Si el nombre contiene siglas deberán ser aclaradas)

Profesor de la asignatura ¹ :Dr. Ing. Gustavo Betarte, Profesor Titular (Gr. 5) InCo

Msc. Carlos Luna, Profesor Adjunto (Gr. 3), InCo

(título, nombre, grado o cargo, Instituto o Institución)

Profesor Responsable Local ¹ :

(título, nombre, grado, Instituto)

Otros docentes de la Facultad:

(título, nombre, grado, Instituto)

Docentes fuera de Facultad:

(título, nombre, cargo, Institución, país)

Instituto ó Unidad: Instituto de Computación

Departamento ó Area: Departamento de Programación

¹ Agregar CV si el curso se dicta por primera vez.

(Si el profesor de la asignatura no es docente de la Facultad se deberá designar un responsable local)

Fecha de inicio y finalización: segundo semestre

Horario y Salón: A determinar

Horas Presenciales: 66

(se deberán discriminar las mismas en el ítem Metodología de enseñanza)

Nº de Créditos: 10

(de acuerdo a la definición de la UdelaR, un crédito equivale a 15 horas de dedicación del estudiante según se detalla en el ítem metodología de la enseñanza)

Público objetivo y Cupos:

Estudiantes avanzados de Ingeniería en Computación. No hay cupos para estudiantes de posgrado.

Objetivos: Iniciar al estudiante al uso de métodos formales para la producción de software correcto por construcción

Conocimientos previos exigidos: Cálculo Proposicional y de Predicados. Dedución natural. Conjuntos y pruebas inductivas. Funciones de alto orden

Conocimientos previos recomendados: Cálculos funcionales. Sistemas de Tipos. Programación Funcional.

Programación Lógica, procedimientos de unificación de términos

Metodología de enseñanza:

(comprende una descripción de las horas dedicadas por el estudiante a la asignatura y su distribución en horas presenciales -de clase práctica, teórico, laboratorio, consulta, etc.- y no presenciales de trabajo personal del estudiante)

- Horas clase (teórico): 30
- Horas clase (práctico):
- Horas clase (laboratorio):15
- Horas consulta:15
- Horas evaluación:6
 - Subtotal horas presenciales: 66
- Horas estudio:30
- Horas resolución ejercicios/prácticos:60
- Horas proyecto final/monografía:
 - Total de horas de dedicación del estudiante: 156

Forma de evaluación:

La asignatura se evaluará por medio de dos parciales y trabajos de laboratorio. El nivel mínimo de suficiencia en los trabajos de laboratorio es eliminatorio, ya que esta parte del trabajo del curso no puede ser evaluada mediante exámenes. Por otra parte, dependiendo de las condiciones de dictado del curso, el trabajo de laboratorio se evalúa según las opciones aprobado/ no aprobado, o con puntaje diferenciado en el caso de aprobación. En este último caso, el puntaje del laboratorio se integraría al puntaje total del curso, prorrateándose en los de las pruebas parciales.

En todos los casos de los resultados obtenidos surgen dos posibilidades:

- Exoneración del curso
- Insuficiencia en el curso; el estudiante reprueba el curso

Se presenta a continuación el esquema de evaluación del curso

Exoneración. El estudiante debe cumplir los siguientes requisitos:

1. llegar al nivel mínimo en los trabajos de laboratorio
2. reunir al menos el 60 % del puntaje de parciales
3. obtener al menos el 25 % en cada prueba parcial

Insuficiencia. El estudiante no obtiene los puntajes de ninguna de las franjas anteriores

Temario:

1. Asistentes de pruebas para lógicos y matemáticos: Una presentación formal de la lógica proposicional y de primer orden.
2. Asistentes de pruebas para programadores: El calculo lambda como lenguaje de programación funcional.
3. Identificación de pruebas y programas: Isomorfismo de Curry Howard



Facultad de Ingeniería Comisión Académica de Posgrado

4. Recursión: Definiciones Inductivas, Principios de Inducción, Esquemas de Recursión
 5. Extracción de programas a partir de pruebas. Construcción de Pruebas a partir de programas
 6. Construcción de programas certificados usando Coq.
-

Bibliografía:

Interactive Theorem Proving and Program Development. Y. Bertot, P. Casteran, Springer- Verlag. 2004

Computation and Reasoning: A Type Theory for Computer Science. Volume 11 of International Series of Monographs on Computer Science. Oxford Science Publications, 1994, ISBN: 0198538359

The Coq Proof Assistant. Reference Manual. B. Barras et. al. Rapport de Recherche INRIA. Disponible en: <http://pauillac.inria.fr/coq/doc-eng.html>

A Tutorial on Recursive Types in Coq. Eduardo Giménez, Rapport de Recherche INRIA, Disponible en: <http://pauillac.inria.fr/coq/doc-eng.html>
