

## Teoría de Tipos

Objetivos. La Teoría de Tipos es una familia de formalismos ideados para servir de fundamento a la Matemática que ha demostrado tener aplicaciones muy significativas en Ciencia de la Computación. Entre ellas se cuentan: los sistemas de tipos de los lenguajes de programación, las lógicas de programación --especialmente las relativas a lenguajes funcionales-- y los modelos o estructuras de sistemas lógicos. Asimismo, asociados a estos formalismos se desarrollan e investigan asistentes de desarrollo o demostración implementados en máquina que permiten su utilización práctica. Este curso aspira a desarrollar teoría y práctica de las teorías de tipos, estudiando su formulación, principales propiedades matemáticas, herramientas relacionadas y aplicaciones.

Programa. El curso tiene una duración de 16 semanas, con 4hs. semanales de clases teórico-prácticas.

1. *Cálculo Lambda puro*. Reducción. Teorema de Church-Rosser. Estrategias de reducción – estandarización. Expresividad. Indecibilidad.
2. *Cálculo Lambda con tipos simples*. Tipos explícitos (á la Church). Asignación de tipos (á la Curry) – Esquema de tipo principal, algoritmo de asignación. Normalización. Lenguaje funcional mínimo: propiedades de las computaciones impuestas por el sistema de tipos.
3. *Lógica Intuicionista*. Lógica minimal – el isomorfismo de Curry-Howard. Interpretación intuicionista de la noción de proposición. Lógica proposicional intuicionista. Lógica de primer orden. Aritmética.
4. *Teoría Constructiva de Tipos*. Sistema T de Gödel. Tipos dependientes. Teoría de Tipos: estructura lógica. Teoría de conjuntos, definiciones inductivas.
5. *Lógica Proposicional de Segundo Orden*. El sistema F de Girard. Expresividad.
6. *Cálculo de Construcciones*. Formulación y propiedades. El cubo de Barendregt.
7. *Asistentes de demostración*. Coq, Agda. Práctica.

Evaluación. Se evaluará por medio de trabajos domiciliarios consistentes en la resolución de problemas. Las soluciones serán además expuestas y discutidas en clase.

### Bibliografía.

Bertot, Y.; Castéran, P.; Huet, G.; Paulin-Mohring, C. *Interactive Theorem Proving and Program Development*. Springer, 2004. ISBN-10: 3540208542, ISBN-13: 978-3540208549.

Girard, J.-Y.; Lafont, Y.; Taylor, P. *Proofs and Types*. Cambridge University Press, 1989. ISBN-10: 0521371813, ISBN-13: 978-0521371810.

Heine Sørensen, M.; Urzyczyn, P. *Lectures on the Curry-Howard Isomorphism*. Elsevier Science, Volumen 149 de Studies in Logic and the Foundations of Mathematics, 2006. ISBN-10: 0444520775, ISBN-13: 978-0444520777

Hindley, J. Roger. *Basic simple type theory*. Cambridge: Cambridge University Press, 1997. (Cambridge tracts in theoretical computer science; 42). ISBN: 0-521-46518-4

Hindley, J. Roger; Seldin, Jonathan P. *Lambda-Calculus and Combinators: an Introduction*. Cambridge University Press, 2008. ISBN 978-0-521-89885-0.

Luo, Zhaohui. *Computation and reasoning: a type theory for computer science*. New York: Clarendon, 1994. ISBN: 0-19-853835-9.

Martin-Löf, Per. *Intuitionistic type theory*. Padova: Bibliopolis, 1980.

Nordström, Bengt; Petersson, Kent; Smith, Jan M. *Programming in Martin-Löf's type theory : an introduction*. Oxford: Clarendon Press, 1990. ISBN: 0-19-853814-6.

Thompson, Simon. *Type theory and functional programming*. Wokingham: Addison-Wesley, 1991. ISBN: 0-201-41667-0