

Fecha de inicio y finalización: 02/05/2017 - 30/04/2019

Director: Bonelli, Eduardo.

Co-Director: Martínez López, Pablo.

Integrantes: Lombardi, Carlos; Barenbaum, Pablo; Arévalo, Gabriela Beatriz; Passerini, Pablo Nicolás; Ríos, Alejandro; Ciolek, Daniel; Viso, Andrés; Repond, Federico.

Título: TIPADO PARA PATRONES Y REDUCCIÓN EFICIENTE EN PROGRAMACIÓN FUNCIONAL.

Resumen: El proyecto propone desarrollar dos líneas de trabajo en el marco de Lenguajes de Programación Funcional, a saber, reducción eficiente de términos y tipado para patrones generalizados. La programación funcional es un paradigma de programación, así como lo es la programación imperativa (ej. C) y la programación orientada a objetos (ej. Java). Un paradigma es esencialmente un conjunto de abstracciones que permiten resolver problemas algorítmicos. Ejemplos de lenguajes de programación pertenecientes a este paradigma son Haskell, OCaml y Erlang. Estos lenguajes se destacan por su alto nivel de abstracción y sus robustos fundamentos matemáticos. Hay un uso cada vez más extenso de estos lenguajes en la industria. Lo mismo sucede con muchas de las abstracciones que se originaron en el paradigma funcional y que por su potencia y versatilidad fueron migrando a lenguajes de otros paradigmas: garbage collection (Java, Smalltalk, etc.), generics (Java, Scala), continuaciones (C#, Perl, Ruby, etc.), funciones anónimas (C#, Perl, Ruby, etc.), etc. Este proyecto propone dos extensiones a los lenguajes de programación funcional.

La ejecución un programa en un lenguaje funcional consiste en tomar el programa e ir simplificándolo hasta llegar a un resultado final. Este modelo de ejecución se conoce como ejecución por evaluación. Este proceso de evaluación sigue reglas previamente establecidas para cada una de las construcciones del lenguaje. Por ejemplo, si un programa hace uso de la suma, hay reglas que indican cómo se comporta la suma. Es importante que la evaluación sea lo más eficiente posible. La primera línea de trabajo de este proyecto es considerar técnicas de evaluación eficientes. Particularmente en el caso de los lenguajes funcionales que forman parte de asistentes de prueba como Coq. El objetivo es definir nociones de reducción call-by-need que computen programas a forma normal.

Los programas en un lenguaje funcional consisten de una serie de definiciones de funciones que interactúan entre sí para llevar a cabo alguna tarea. Estas definiciones de funciones suelen hacer un uso intensivo de pattern matching, una técnica que permite describir cómo operar sobre estructuras de datos de una manera concisa y descriptiva. La segunda línea de este proyecto propone la adopción de patrones más generales y, por tanto, más expresivos que aquellos existentes. El objetivo es poder definir funciones que operen, indistintamente, sobre listas, árboles y otras estructuras de datos recursivas.

