

Práctica 5: Implementación de MiniC

Lenguajes de Programación I

López Soto Ramses Antonio
Quintero Villeda Erik

November 7, 2019

Introducción

Objetivo

Extender *BAE* para que se comporte como un *lenguaje de programación imperativo*.

Desarrollo

Los módulos implementados anteriormente que fueron extendidos con las funciones adaptadas:

- Syntax
 - `frVars :: Expr -> [Identifier]`
 - `subst :: Expr -> Substitution -> Expr`
- Dynamic
 - `eval1 :: (Memory, Expr) -> (Memory, Expr)`
 - `evals :: (Memory, Expr) -> (Memory, Expr)`
 - `evale :: Expr -> Expr`

También se implementó un nuevo módulo *Memory* con las siguientes funciones:

- `newAddress :: Memory -> Expr`
- `access :: Address -> Memory -> Maybe Value`
- `update :: Cell -> Memory -> Maybe Value`

Para mayor detalle, véase la implementación en el directorio `BAE/src/BAE/`.

Conclusión

La práctica nos tomó más tiempo de lo esperado pues, como en prácticas pasadas, el módulo *Dynamic* nos causó muchos problemas. Al extender *BAE* haciendo uso del módulo *Memory*, no se reconocía éste, mandaba errores que no entendíamos, la extensión nos costó más trabajo de implementar, pero todas las funciones se pudieron hacer casi en su totalidad. A diferencia de *Dynamic*, los módulos *Syntax* y *Memory* fueron muy sencillos, pues sólo bastaba con gregar algunas cosas y/o implementar cosas muy rápido.

En conclusión, El objetivo inicial de la práctica fue alcanzado, pues se implementó una simulación (pequeña) de lo que es un *Lenguaje de programación imperativo*