

Генерация зависимых языков по спецификации пользователя

Выступающий: Гарифуллин Шамиль Раифович
Научный руководитель: Исаев Валерий Иванович

СПбАУ

31 мая 2017 г.

Языки с зависимыми типами — типы могут зависеть от термов. Одна из частых ошибок при программировании на языке Haskell — взятие первого элемента пустого списка.

```
head :: [a] -> a
head (x:_) = x
head [] = error "No head!"
```

В языке с зависимыми типами мы можем усилить ограничения на входные данные функции.

```
head :: {n : N} -> Vec a (suc n) -> a
head (x:_) = x
```

Одно из основных применений языков с зависимыми это доказательство утверждений.

Реализовать генерацию алгоритма проверки типов и вычислителя зависимых языков по спецификации

- Дизайн языка спецификации
- Выбор внутреннего представления и генерация структур данных конструкций языка
- Генерация кода функций проверки типов и нормализации

Язык программирования состоит из

- Конструкций
- Правил построения конструкций (правил вывода)
- Правил вычисления (редукций)

Гаекр

Наше представление

Ограничения накладываемые языком?

- Топосорт
- Проверка типов и проч
- Идея в том что этим мы отличаемся от всяких LF

Индексы де Брейна?

Индексы де Брейна?

Пример кода?

Реализована генерация алгоритма проверки типов и вычислителя зависимых языков по спецификации.

- Спроектирован типизированный язык спецификации
- Генерация структур данных конструкций языка с использованием индексов де Брейна на уровне типов с использованием полиморфной рекурсии
- Генерация кода функций проверки типов и нормализации

Репозиторий проекта: github.com/esengie/fpl-exploration-tool/