Программа

Введение

- Особенности функционального программирования(ФП)
- Отличия ФП от императивного программирования
- Сильные и слабые стороны ФП

Лямбда-исчисление

- Формализации понятия алгоритма
- Чистое λ-исчисление
- Свободные и связанные переменные
- Комбинаторы
- Каррирование
- Подстановка
- Правила преобразования
- Эквивалентность λ-термов

Рекурсия и редукция

- Теорема о неподвижной точке, Y-комбинатор
- Совместимые (с λ-исчислением) отношения, отношения редукции, отношения равенства(конгруэнтности)
- Одношаговая β-редукция, β-редукция, β-равенство
- Редукционные графы
- Нормальная форма(nf)
- Ненормализуемые, слабо нормализуемые и сильно нормализуемые термы
- Теорема Чёрча-Россера, теорема об общем редукте, теорема о единственности nf
- Стратегии редукции
- Головная нормальная форма, слабая головная нормальная форма
- Теорема о нормализации

Просто(е) типизированное лямбда-исчисление

- Понятие типа (предназначение типов)
- Просто(е) типизированное λ-исчисление (STLC)
- Типизация по Чёрчу и Карри
- Множество типов, предтермы, утверждение о типизации, контекст
- Правила типизации
- Леммы генерации, о контекстах
- Теорема о редукции субъекта
- Теорема о единственности типа для типизации по Чёрчу
- Теорема о нормализации

Введение в язык Haskell

- Базовые типы
- Связывание
- Определение функций
- Охранные выражения
- Конструкции where и let…in
- Операторы и сечения
- Модули
- Реализации языка Haskell и инфраструктура (cabal, stack, hackage, hoogle)

Основы программирования в Haskell

- Ленивые и строгие вычисления
- Алгебраические типы данных: декартово произведение и размеченное объединение
- Сопоставление с образцом
- Работа со списками
- Генераторы списков
- Функции высших порядков над списками

Классы типов

- Виды полиморфизма
- Классы типов
- Сравнение с другими языками программирования
- Обзор стандартных классов типов(Eq, Ord, Enum, Bounded, Show, Read, Num, Fractional, Integral)
- Особенности внутренней реализации классов типов (реализация с помощью словарей)

Свертки и моноиды

- Левая и правая свертки (foldr, foldl, foldl', foldr1, foldl1)
- Родственные сверткам функции (scanl, scanr, unfold)
- Semigroup a, закон для полугруппы
- Semigroup a ⇒ Monoid a, законы для моноида
- Стандартные моноиды ([a], Any, All, Product a, Sum a, Endo a, Dual a, Last a, First a)
- Foldable

Функторы

- Функторы
- Законы для функторов
- Аппликативные функторы
- Законы для аппликативных функторов

- Функция как функтор, аппликативный функтор
- Список как аппликативный функтор, ZipList

Монады

- Стрелка Клейсли
- Понятие монады
- Класс типов Monad
- Монады Identity, Maybe
- Список как монада
- Отличие монад от аппликативных функторов

Стандартные монады

- Reader
- Writer
- State
- IO
- Функции ввода-вывода

Остальное не успели

Использование аппликативных функторов

- Аппликативные парсеры
- Класс типов Alternative
- Законы класса Alternative
- Класс типов Traversable
- Законы класса Traversable

Трансформеры монад

- Класс типов MonadPlus
- Законы класса MonadPlus
- Монада Except
- Мультипараметрические классы типов
- Трансформеры монад
- Законы для класса типов MonadTrans
- Стандартные трансформеры библиотеки mtl

Вывод типов*

Программирование с зависимыми типами*

Чисто функциональные структуры данных*

ФП в mainstream-языках*

GHC Core*

From:
https://se.moevm.info/ - **МОЭВМ Вики [se.moevm.info]**

Permanent link:
https://se.moevm.info/doku.php/courses:functional_programming:lectures

Last update: