

Программа курса

«Теория языков программирования»

Введение. Теория типов. Системы типов в проектировании языков. Математический аппарат. Множества, отношения и функции. Упорядоченные множества. Последовательности. Индукция.

Раздел 1: Основы теории типов

Бестиповые системы. Бестиповые арифметические выражения. Синтаксис, семантика, вычисление. Индукция на термах. Реализация арифметических выражений на функциональном языке.

Бестиповое лямбда-исчисление. Программирование на языке лямбда-исчисления. Представление термов без использования имен. Термы и контексты. Сдвиг и подстановка. Реализация лямбда-исчисления на функциональном языке.

Простые типы. Типизированные арифметические выражения. Отношение типизации. Простое типизированное лямбда-исчисление. Типы функций. Свойства типизации. Соответствие Карри-Говарда. Стирание типов и типизируемость. Реализация простых типов на функциональном языке.

Простые расширения. Базовые типы. Единичный тип. Производные формы. Приписывание типа. Связывание let. Пары. Кортежи. Записи. Типы-суммы. Варианты. Рекурсия общего вида. Списки.

Нормализация. Ссылки. Типизация содержимого памяти. Исключения: порождение и обработка. Исключения, сопровождаемые значениями.

Раздел 2: Подтипы и рекурсивные типы

Подтипы. Включение. Отношения вложения типов. Свойства подтипов и типизации. Типы Top и Bottom. Подтипы и другие элементы языка. Семантика, основанная на приведении типов. Типы-пересечения и типы-объединения.

Метатеория подтипов. Алгоритмическое вложения типов. Алгоритмическое отношение типизации. Пересечения и объединения. Алгоритмическая типизация и тип Bot. Реализация подтипов на функциональном языке.

Императивные объекты. Объектно-ориентированное программирование. Объекты и их генераторы. Группировка переменных экземпляра. Простые классы. Добавление новых переменных экземпляра. Вызов методов надклассом. Классы с переменной Self. Открытая рекурсия через Self. Открытая рекурсия и порядок вычислений.

Расширенный пример: облегченная Java. Именные и структурные системы типов. Определения. Свойства.

Рекурсивные типы. Формальные определения. Метатеория рекурсивных типов. Индукция и коиндукция. Конечные и бесконечные типы. Проверка принадлежности. Регулярные деревья. μ -типы. Подсчет выражений. Вложения типов для изорекурсивных типов.

Раздел 3: Полиморфизм и системы высших порядков

Полиморфизм. Реконструкция типов. Типовые переменные и подстановки. Типизация на основе ограничений. Унификация. Главные типы. Неявные аннотации типов. Полиморфизм через `let`.

Универсальные типы. Разновидности полиморфизма. Система F . Стирание, типизируемость и реконструкция типов. Параметричность. Импредикативность.

Экзистенциальные типы. Абстракция данных при помощи экзистенциальных типов. Кодирование экзистенциальных типов. Реализация системы F на функциональном языке. Ограниченная квантификация. Ограниченные экзистенциальные типы.

Метатеория ограниченной квантификации. Выявление. Минимальная типизация. Подтипы в ядерной $F_{<}$. Подтипы в полной $F_{<}$. Неразрешимость в полной $F_{<}$. Ограниченные кванторы существования. Ограниченная квантификация и тип `Bot`.

Системы высших порядков. Операторы над типами и виды. Полиморфизм высших порядков. Свойства исчисления. Варианты F_{ω} . Зависимые типы. Подтипы высших порядков.

Чисто функциональные объекты (пример). Типы интерфейсов. Отправка сообщений объектам. Простые классы. Полиморфные обновления. Добавление переменных экземпляра. Классы с переменной `Self`.

Дополнительно: π -исчисление и теория параллелизма.

Рекомендуемая литература:

1. Benjamin Pierce. Types and Programming Languages. — MIT Press, 2002.

Перевод на русский язык: Бенджамин Пирс. Типы в языках программирования. — Добросвет, 2012. — 680 с.

2. Robert Harper. Practical Foundations for Programming Languages.
3. Ilya Sergey. Programs and Proofs Mechanizing Mathematics with Dependent Types