

«Функциональное программирование»: вопросы к экзамену

Экзаминационный билет содержит два вопроса: первый — по λ -исчислению и прочей теории, второй — по языку Haskell.

Теория

1. Сравнение функционального и императивного подходов к программированию. Функциональная модель вычислений.
2. Чистое λ -исчисление. λ -термы, свободные и связанные переменные. Классические комбинаторы, комбинаторная логика.
3. Подстановка, лемма подстановки. Одношаговая и многошаговая β -редукция, β -эквивалентность.
4. α -эквивалентность. Индексы Де Брауна. η -эквивалентность и принцип экстенциональности.
5. Кодирование булевых значений, кортежей в чистом бестиповом λ -исчислении.
6. Кодирование чисел Чёрча в чистом бестиповом λ -исчислении. Арифметические операции над ними.
7. Теорема о неподвижной точке. Y -комбинатор. Решение рекурсивных уравнений на термы.
8. Нормальная форма. Редукционные графы.
9. Теорема Чёрча-Россера. Параллельная β -редукция. Полная эволюция.
10. Следствия теоремы Чёрча-Россера.
11. Стратегии редукции. Теорема о нормализации. Механизмы вызова в функциональных языках.
12. Функция предшествования для чисел Чёрча. Комбинатор примитивной рекурсии.
13. Просто типизированное λ -исчисление в стиле Карри. Предтермы. Утверждения о типизации. Контексты. Правила типизации.
14. Просто типизированное λ -исчисление в стиле Чёрча. Предтермы. Утверждения о типизации. Контексты. Правила типизации.
15. Свойства систем просто типизированного λ -исчисления.
16. Связь между системами Карри и Чёрча. Проблемы разрешимости. Сильная и слабая нормализация.
17. Правило `foldr/build` и реализация высокоуровневых оптимизаций в GHC.
18. Понятие главного (наиболее общего) типа. Подстановки типа и их композиция. Унификаторы.
19. Алгоритм унификации.
20. Алгоритм построения системы ограничений для типизации терма.
21. Главная пара и главный тип. Теорема Хиндли-Милнера.
22. Обобщения алгоритма Хиндли-Милнера. `let`-полиморфизм и его ограничения.
23. Структурное представление алгебраических и рекурсивных типов. μ -нотация. Эквирекурсивные и изорекурсивные типы.
24. Реализация рекурсивных через неподвижную точку в Haskell. Катаморфизмы.
25. Анаморфизмы и гилеморфизмы.
26. Зипперы (молнии). Контексты с дыркой.
27. Экзистенциальные типы в Haskell.
28. Линзы ван Лаарховена.

Haskell

1. Основы программирования на Haskell. Связывания. Рекурсия. Базовые конструкции языка. Частичное применение. Бесточечный стиль.
2. Основные встроенные типы языка Haskell. Параметрический полиморфизм. Система модулей.
3. Операторы и их сечения в Haskell.
4. Строгие и нестрогие функции. Ленивое и энергичное исполнение. Форсирование, слабая головная нормальная форма.
5. Списки, стандартные функции для работы с ними. Генерация (выделение) списков.
6. Алгебраические типы данных. Сопоставление с образцом, его семантика. Полиморфные и рекурсивные типы данных.
7. Трансляция образцов в Kernel. Синонимы в образцах, ленивые и охранные образцы. Образцы в λ - и `let`-выражениях.
8. Объявления `type` и `newtype`. Метки полей. Строгие конструкторы данных.
9. Классы типов. Объявления представителей. Классы типов `Eq`, `Ord`, `Enum` и `Bound`.
10. Внутренняя реализация классов типов.
11. Стандартные классы типов: `Num` и его наследники, `Show` и `Read`.
12. Полугруппы и моноиды. Представители класса типов `Monoid`.
13. Правая и левая свёртки списков. Энергичные версии. Развертки.
14. Класс типов `Foldable` и его представители.
15. Класс типов `Functor` и его представители.
16. Класс типов `Applicative` и его представители.
17. Классы типов `Alternative` и `MonadPlus` и их представители.
18. Аппликативные парсеры.
19. Класс типов `Traversable` и его представители.
20. Монады. Класс типов `Monad`. Монада `Identity`. Законы монад. `do`-нотация.
21. Класс типов `MonadFail`, его история и представители.
22. Стандартные монады: `Maybe` и списки.
23. Ввод-вывод в чистых языках. Монада `IO`. Взаимодействие с файловой системой.
24. Монада `Reader`.
25. Монада `Writer`.
26. Монада `State`.
27. Монада `Except`.
28. Мультипараметрические классы типов. Роль классов `MonadReader`, `MonadWriter`, `MonadState` и `MonadError` в `mtl`.
29. Трансформеры монад. Библиотеки `transformers` и `mtl`.