

**«Функциональное программирование»: вопросы к экзамену
СПбАУ, осенний семестр 2017**

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.
2. Основы λ -исчисления. λ -термы, свободные и связанные переменные.
3. Подстановка λ -терма. Лемма подстановки.
4. α - и β -конверсии. η -конверсия и экстенциональная эквивалентность.
5. Расширения чистого лямбда-исчисления. δ -конверсия.
6. Кодирование булевых значений, кортежей в чистом бестиповом λ -исчислении.
7. Кодирование чисел Чёрча в чистом бестиповом λ -исчислении.
8. Теорема о неподвижной точке. Y-комбинатор.
9. Редуксы. Одношаговая и многошаговая редукция. Нормальная форма. Редукционные графы.
10. Теорема Чёрча-Россера и её следствия.
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. Свойство универсальности правой свертки. Закон слияния foldr-map.
18. Понятие главного (наиболее общего) типа. Подстановки типа и их композиция. Унификаторы.
19. Алгоритм унификации.
20. Алгоритм построения системы ограничений.
21. Теорема Хиндли-Милнера.
22. Рекурсивные типы. μ -нотация.
23. Катаморфизмы.
24. Анаморфизмы и гилеоморфизмы.
25. Зипперы. Контексты с дыркой.

Haskell

1. Основы программирования на Haskell. Связывание. Рекурсия. Базовые конструкции языка.
2. Основные встроенные типы языка Haskell. Система модулей. Частичное применение, каррирование.
3. Операторы и их сечения в Haskell. Бесточечный стиль.
4. Ошибки. Основание. Строгие и нестрогие функции. Ленивое и энергичное исполнение.
5. Алгебраические типы данных. Сопоставление с образцом, его семантика.
6. Объявления `type` и `newtype`. Метки полей.
7. Списки, стандартные функции для работы с ними. Генерация (выделение) списков.
8. Специальный полиморфизм. Классы типов. Объявления представителей. Классы типов `Eq`, `Ord`, `Enum` и `Bound`.
9. Внутренняя реализация классов типов.
10. Стандартные классы типов: `Num` и его наследники, `Show` и `Read`.
11. Моноиды. Представители класса типов `Monoid`.
12. Свёртки списков. Правая и левая свёртки. Энергичные версии. Развертки.
13. Класс типов `Foldable` и его представители.
14. Класс типов `Functor` и его представители.
15. Класс типов `Applicative` и его представители.
16. Классы типов `Alternative` и `MonadPlus`.
17. Апликативные парсеры.
18. Класс типов `Traversable` и его представители.
19. Монады. Класс типов `Monad`. Законы для монад. `do`-нотация.
20. Стандартные монады: `Maybe` и списки.
21. Ввод-вывод в чистых языках. Монада `IO`. Взаимодействие с файловой системой.
22. Монада `Reader`.
23. Монада `Writer`.
24. Монада `State`.
25. Монада `Except`.
26. Трансформеры монад. Библиотеки `transformers` и `mtl`.
27. Линзы ван Лаарховена.