

# Функциональное программирование

Цели, способы, применимость

Алексей Пирогов, 2019

# О чём поговорим?

- Функциональное программирование, а **что** это?
- **Зачем** нужна ещё одна парадигма?
- **Как** применять ФП в обычной жизни программиста с пользой?

Что?

# Что?

Сначала немного истории:

- В начале было **Лямбда Исчисление...**
- **ЛИ** против **Машины Тьюринга** (тезис Чёрча-Тьюринга)
- да-да, **LISP**
- (и понеслось...)

# Что?

Определение *(всего лишь одно из, но удобное)*:

*ФП, это отделение чистого кода<sup>(?)</sup> от кода с побочными эффектами<sup>(?)</sup>.*

# Что?

Ключевые свойства ФП:

- Функции высших порядков
- Иммутабельность данных

Зачем?

# Зачем?

Чистый код просто

- анализировать (рассуждать о)
- тестировать
- переиспользовать
- вычислять параллельно и/или конкурентно



# Зачем?

Ещё больше профитов даёт ленивость (при условии чистоты).

Как?

# Как?

Как получить максимальную пользу от ФП?

Писать функционально:

- компиляторы, парсеры, генераторы чего-либо
- сложную логику, лежащую на DSL
- высококонкурентный код (*Erlang, это про тебя*)
- GUI (*aga, React*)

Как?

Как научиться (этому вашему) ФП?

Изучить **Haskell** во имя добра!

# Как?

Как просветлиться ещё больше?

Изучить **Clojure** (ибо LISP и макросы)

Изучить **Erlang** (ибо акторы)

Как?

Как изучить ФП на моём привычном языке, да чтобы побыстрее?

**Никак.** Учите Haskell

Bcë!