

# Функциональный язык программирования с динамической типизацией и ML-подобным синтаксисом

Выполнила: Волкова Ю.А.  
Руководитель: Дубанов А.В.

# Задачи

- Разработать синтаксис языка, удобный для написания сценариев и первоначального обучения программированию.
- Разработать компилятор для этого языка.

# Основные этапы работы

- Разработка синтаксиса языка
- Написание компилятора для языка

# Особенности языка

- Динамическая типизация (Scheme)
- Сопоставление с образцом (Prolog, Haskell, OCaml)
- Повторные переменные (Prolog)
- Отступы вместо ограничителей блоков кода (Haskell)

# Разработка компилятора

- Чтение входного потока
- Лексический анализатор
- Синтаксический анализатор
- Семантический анализатор
- Генератор кода

# Примеры кода на Языке

1. Вычисление факториала числа:

```
n! 0 <- 1  
n! n <- n * n! (n - 1)
```

2. Подсчёт количества вхождений элемента  $x$  в список  $xs$ :

```
count x [] <- 0  
count x [ x : xs ] <- 1 + count x xs  
count x [ y : xs ] <- count x xs
```

### 3. Сумма произвольного числа аргументов

```
sum <- 0
sum x : xs <- x + sum . xs
```

### 4. Фильтр элементов списка по функции-предикату

```
filter pred? [] <- []
filter pred? [x : xs] <-
  if | pred? x -> [x] ++ filter pred? xs
    |           -> filter pred? xs
```

### 5. Вычисление чисел Фибоначчи

```
fibonacci 0 <- 1
fibonacci 1 <- 1
fibonacci n <- fibonacci (n - 1) + fibonacci (n - 2)

memo fibonacci
```

# Технико-экономические требования

- Количество исполнителей: 3.
- Суммарные затраты: 5 486 200 рублей.
- Итоговый чистый дисконтированный доход: 2 019 802 рубля.



# Заключение

- Предложен функциональный язык программирования с динамической типизацией и ML-подобным синтаксисом.
- Разработан компилятор для этого языка.