

»= Haskell

История Haskell

Что такое Haskell?

Haskell (hæskəl) —
стандартизированный чистый функциональный язык
к программирования общего назначения. Является
одним из самых распространённых языков
программирования с поддержкой отложенных
вычислений.



С чего всё начиналось?

Функциональное программирование как раз основано на идеях из комбинаторной логики и лямбда-исчисления.

Но теория так и оставалась теорией, пока в начале 50-х прошлого века Джон МакКарти не разработал язык Lisp (1958), который стал первым почти функциональным языком программирования. На протяжении многих лет у Lisp не было конкурентов. Позднее появились функциональные языки программирования APL (1964), ISWIM (1966) и FP (1977), которые не получили столь широкого распространения.

The word "Lisp" is rendered in a large, dark green, stylized font. The letters are thick and blocky, with a slightly irregular, hand-drawn appearance. The 'i' has a dot, and the 's' is a simple, rounded shape. The overall style is reminiscent of a logo or a brand name.

Miranda

История языка Haskell начинается в 1987 году. Один за другим появлялись новые функциональные языки программирования. После выхода Miranda (Research Software Ltd, 1985 год) возрос интерес к ленивым вычислениям: к 1987 году возникло более дюжины нестрогих чисто функциональных языков программирования.



Наконец-то!

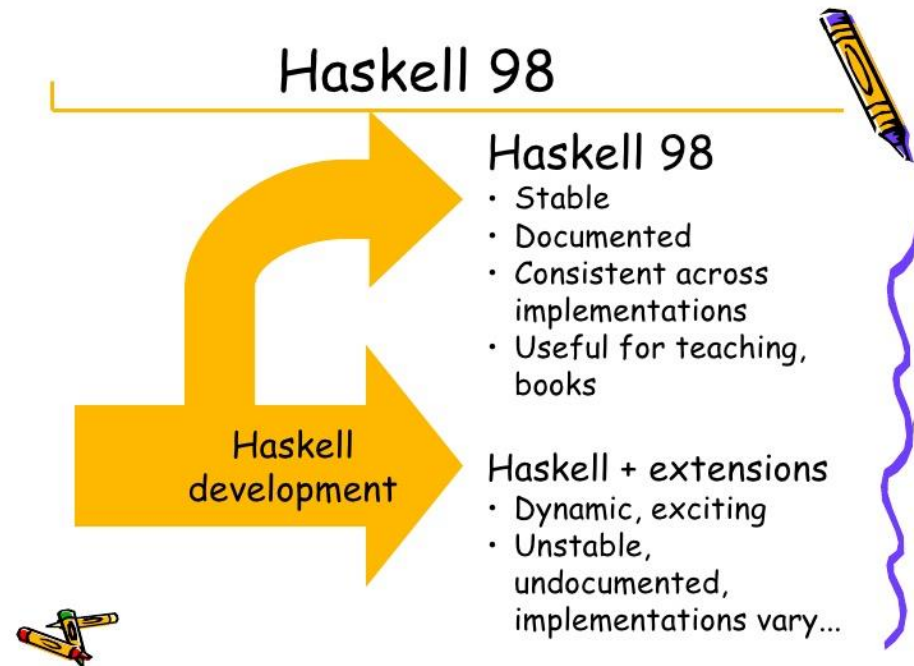
Miranda использовался наиболее широко, но это было запатентованное ПО. На конференции по функциональным языкам программирования и компьютерной архитектуре (FPSA, 1987) в Портленде (Орегон) участники пришли к соглашению, что должен быть создан комитет для определения открытого стандарта для подобных языков. Целью комитета являлось объединение существующих функциональных языков в один общий, который бы предоставлял базис для будущих исследований в разработке функциональных языков программирования.

Так появился Haskell. Он был назван в честь одного из основателей комбинаторной логики Хаскела Карри (Haskell Curry).



Дальнейшее развитие

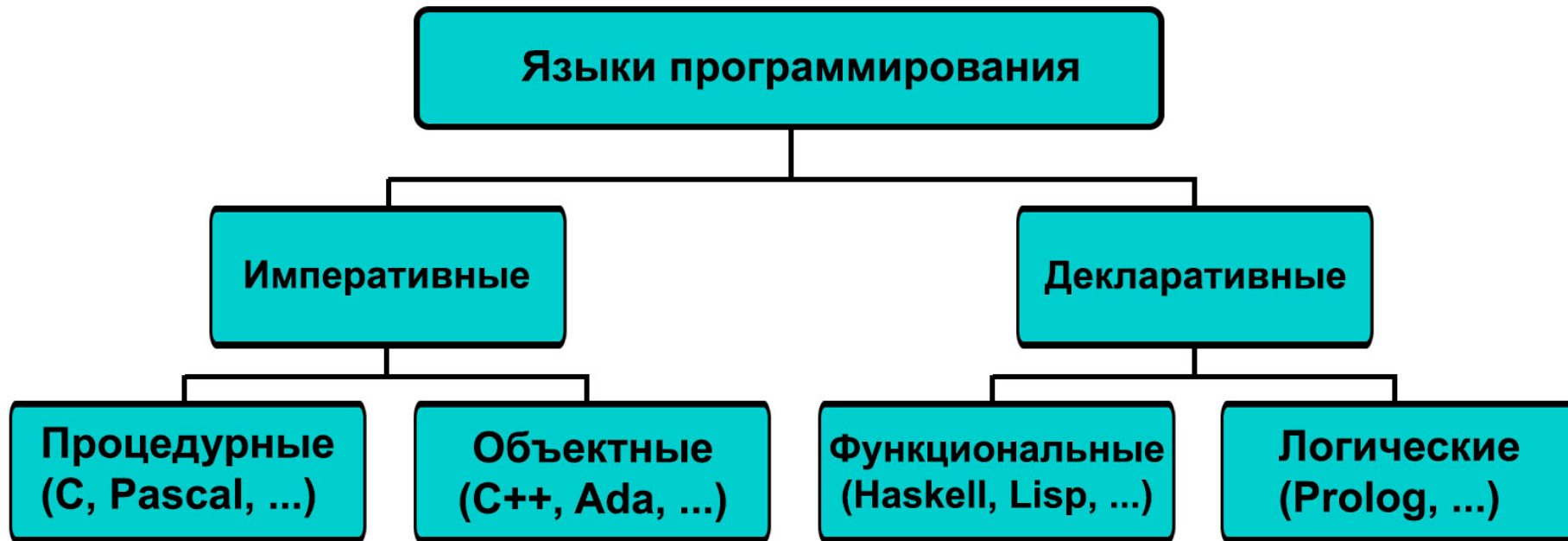
В конце 1997 года версии Haskell 98 необходимо было специфицировать в одну стабильную, минимальную и портативную версию языка и сопроводительную стандартную библиотеку для изучения, как основу будущего развития. Комитет положительным образом принял создание расширений и вариаций haskell 98 путем добавления и внедрения экспериментальных особенностей.



Особенности Haskell

Особенности:

- Функциональный язык;



Особенности:

- Haskell — «ленивый» язык;

```
emacs-25.2@debian
import Data.IORef

data SCM = Environment (IORef Bool) SCM | Empty'Environment

global :: IO SCM
global = Environment <$> newIORef False <*> pure Empty'Environment

print' :: SCM -> IO ()
print' ls =
  case ls of
    Empty'Environment -> pure ()
    Environment first rest -> readIORef first >=> putStr . show >> print' rest

print' :: SCM -> IO ()
print' ls = putStr "(" *> print' ls *> putStrLn ")"

main :: IO ()
main = global >=>
  \ls -> case ls of
    Empty'Environment -> pure ()
    Environment first _ -> print' ls *>
      modifyIORef first (const True) *>
        print' ls *>
          main

U:--- ioref.hs All (25,0) (Haskell)
```



Особенности:

- Модульный язык;

Sorting in Haskell

```
sort []      = []
sort (x:xs) = sort [y | y<-xs, y<x] ++
                [x] ++
                sort [y | y<-xs, y>=x]
```

Sorting in C

```
int partition(int y[], int f, int l);
void quicksort(int x[], int first, int last) {
    int pivIndex = 0;
    if(first < last) {
        pivIndex = partition(x, first, last);
        quicksort(x, first, (pivIndex-1));
        quicksort(x, (pivIndex+1), last);
    }
}

int partition(int y[], int f, int l) {
    int up, down, temp;
    int cc;
    int piv = y[f];
    up = f;
    down = l;
    do {
        while (y[up] <= piv && up < l) {
            up++;
        }
        while (y[down] > piv) {
            down--;
        }
        if (up < down) {
            temp = y[up];
            y[up] = y[down];
            y[down] = temp;
        }
    } while (down > up);
    temp = piv;
    y[f] = y[down];
    y[down] = piv;
    return down;
}
```

Особенности:

- Статически Типизированный язык;

```
1 square :: Int -> Int
2 square a = a * a
3
4 summ :: Float -> Float
5 summ b = b + b
6
7 main = do
8   print(square 6)
9   print(summ 6)
```



Вывод:

```
> ghc -o main main.hs
[1 of 1] Compiling Main                ( main.hs, main.o )
Linking main ...
> ./main
36
12.0
> 
```

Особенности:

- Ремонтпригодный язык;

BEFORE AND AFTER **FUNCTIONAL PROGRAMMING** CAME INTO MY LIFE



Haskell, взгляд изнутри

Говорят, Haskell — язык для гениев и академиков. Правда?

Кто сказал **ГЕНИЙ** ?



Что и где написано на Haskell?



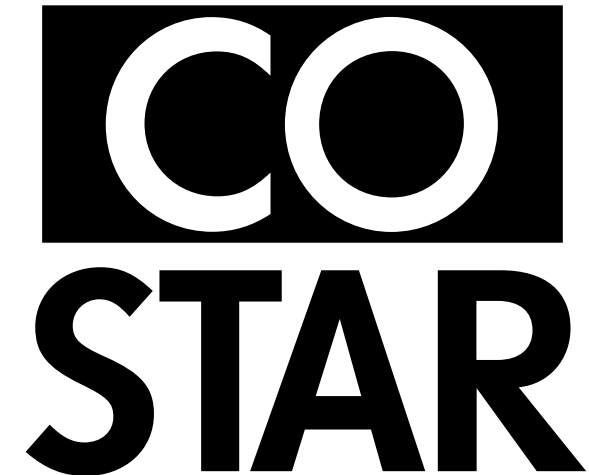
- Для повседневных задач вроде скриптинга, симуляции железа, инструментов для удалённого управления системами транспортного средства и т.д в **Eaton**
- Процесс распознавания музыки приложением **Chordify**
- Компилятора системы безопасности и создание вспомогательных средств разработки в **KasperskyOS**
- Бэкенд внутренних сервисов **BIOCAD** для обработки и хранения данных при разработке препаратов



KasperskyOS®

BIOCAD
Biotechnology Company

- Разработка блокчейнов и криптовалюты в **Cardano**
- Бэкенд **Co–Star** — астрологическом приложении для персонализированных гороскопов и ретроградных Меркуриев
- Бэкенд мессенджера **Wire**
- Исследования филогенетических графов американским музеем естественной истории



Спасибо за внимание!

