

If686 - Paradigmas de Linguagens Computacionais

Trabalho 1

Eduardo Almeida (eaa4)

O que é um paradigma de linguagem computacional?

- Modelo de programação suportado por linguagens que agrupam características particulares comuns
- Diferem nos conceitos e abstrações utilizadas para representar os elementos de um programa

Programação Funcional

- Baseia-se em funções matemáticas
- Sêmantica e Sintática mais simples
- Alto nível de abstração
- Evita estados ou dados mutáveis
- Lazy evaluation
- Não há alocação explícita de memória

Programação Imperativa

- Fundamenta-se no conceito da máquina de Turing
- Instruções sequenciais
- Possui Laços
- Estados modelados por variáveis globais e locais

Origens da Programação Funcional

- Cálculo Lambda (1930) – Pode ser considerada a primeira linguagem de programação funcional, apesar de nunca ter sido projetada para ser executada em uma máquina
- LISP (1950) – desenvolvida por John McCarthy no MIT. Mesmo não sendo puramente funcional, introduziu a maioria das características encontradas nas modernas linguagens funcionais

Origens da Programação Funcional

- ML e Miranda (1970) – ML foi criada pela Universidade de Edimburgo (Polimorfismo e Inferência de tipos), e Miranda por David Turner na Universidade de Kent (Lazy)
- Haskell (1980) – lançada em uma tentativa de juntar muitas ideias na pesquisa de programação funcional

Aplicação

- Facebook
- Twitter (Scala)
- Microsoft (F#)
- Google (Artigos de recuperação de dados)
- Netflix
- Dropbox

Referências

<http://protocoloti.blogspot.com.br/2012/04/paradigmas-de-programacao.html>

http://www.ppgsc.ufrn.br/~rogerio/material_auxiliar/CLP20131_language_ns_imperativas_funcionais.pdf

http://pt.wikipedia.org/wiki/Programa%C3%A7%C3%A3o_funcional

<http://kodumaro.blogspot.com.br/2010/08/paradigmas-de-programacao.html>

<http://computacao.gigamundo.com/progc/04-introducao-aos-paradigmas-das-linguagens-de-programacao.pdf>

<http://protocoloti.blogspot.com.br/2012/04/paradigmas-de-programacao.html>