

PLC

Conceitos

Paradigma de Linguagem Computacional

“Um paradigma de linguagem é a metodologia ou visão de construção de programas adotada na implementação da linguagem, a qual influencia diretamente na forma de aprendizado de desenvolvimento de sistemas de informação.”

- Linguagens se agrupam em paradigmas.

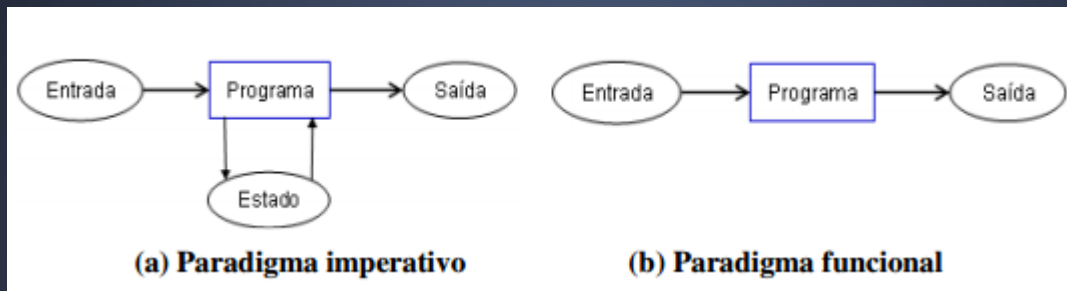
Características fundamentais no paradigma de programação funcional

Focaliza o processo de resolução do problema.

A visão funcional resulta num programa que descreve as operações que devem ser efetuadas para resolver o problema.

O modelo computacional é o de uma função que, portanto, mapeia entradas em saídas de forma determinística.

Programação imperativa x Programação funcional



Imperativa: Mapeamento indireto (estados modelados por variáveis globais e locais)

Funcional: Mapeamento direto (funções definidas)

Programação funcional x Programação imperativa

Variáveis de instância, variáveis globais e estado mutável não existem!

Não há alocação explícita de memória nem declaração explícita de variáveis ->

Essas operações ocorrem automaticamente nos pontos de entrada e saída da função ->

Efeitos colaterais no cálculo da função são eliminados ->

Ao não permitir efeitos colaterais em funções, a linguagem oferece transparência referencial ->

O resultado da função será o mesmo para um dado conjunto de parâmetros não importando onde, ou quando, seja avaliada

^

Facilidade em comprovar a correção do programa

^

Identificação automática de computações independentes para execução paralela

Recursividade >> Laços (Construção de programação imperativa)

Origens da programação funcional

1930 - Cálculo Lambda (Alonzo Church) - Descrição formal do cálculo de uma função

Não para computadores

1950 - LISP (John McCarthy - MIT) ... Scheme

Não puramente funcional

1970 - ML, Miranda

1980 - Haskell

Grandes empresas que utilizam programação funcional



(Erlang)



(F#)



(Scala)



(Artigos de recuperação de dados)

Tipos de sistemas desenvolvidos usando-se linguagens funcionais

- Compilador Haskell: The Glasgow Haskell Compiler (GHC)
- Frameworks para desenvolvimento web: Yesod, Happstack
- Gerenciador de janelas do X: XMonad
- Gerenciador de fontes com um conceito novo (teoria de patches): Darcs

OBS: Todos esses em Haskell!

Referências

http://www.oocities.org/whisatugu/cap01_plp.pdf

[http://www.cin.ufpe.](http://www.cin.ufpe.br/~in1007/transparencias/jai/Jai2008Augusto.pdf)

[br/~in1007/transparencias/jai/Jai2008Augusto.pdf](http://www.cin.ufpe.br/~in1007/transparencias/jai/Jai2008Augusto.pdf)

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

<http://eu-hunz.blogspot.com.br/2012/08/programacao-funcional-pura-e-util.html>

<http://borba.blog.br/2010/04/porque-linguagens-funcionais-sao-importantes/>