

Fundamentos de Programação

Unidade 10

Noções de Programação Funcional e de
Programação Lógica

Renato Freitas

Universidade Federal do Ceará - UFC

Russas, outubro de 2024

Programa da Aula

- 1 Introdução à Programação Funcional
- 2 Funções Metamáticas
- 3 Linguagens de Programação Funcional
- 4 Exemplos JavaScript



Introdução à Programação Funcional

Introdução

- Paradigma que surgiu em 1960
- Motivada pelo desenvolvimento de IA e subcampos
- Atender necessidade não atendida pela programação imperativa
- Antiga mas não vingou devido ao pouco poder de processamento à época
- Baseada em funções matemáticas



Característica da Programação Funcional

- Sua computação é vista como uma função matemática mapeando entradas e saídas
- Não há a noção de estado
- Variáveis imutáveis
- Não há laço de repetição
- Recursividade
- Mais confiável, menos propenso a erros
- Sem efeitos colaterais
- Linguagens funcionais puras e não-puras ou híbridas



Funções Metamáticas

- Uma função matemática é um mapeamento de membros de um conjunto, chamado de conjunto domínio, para outro, chamado de conjunto imagem (SEBESTA, 2018).
- As funções são geralmente aplicadas a um elemento em particular do conjunto domínio, fornecido como um parâmetro para a função.
- Uma função leva a, ou retorna, um elemento do conjunto imagem.
- Uma das características fundamentais das funções matemáticas é que a ordem de avaliação de suas expressões de mapeamento é controlada por recursão e expressões condicionais, e não por sequência e repetição iterativa, comuns nas linguagens de programação imperativas.
- elas não têm efeitos colaterais, sempre definem o mesmo valor quando fornecido o mesmo conjunto de argumentos.



Funções Metamáticas

- Funções matemáticas típicas são geralmente escritas como **um nome de função**, seguido de **uma lista de parâmetros** entre parênteses, seguidos pela **expressão de mapeamento**.

Função Simples

$cubo(x) \equiv x * x * x$, onde x é um número real ($x \in \mathbb{R}$)

- O símbolo \equiv é usado para significar “é definido como”.
- O parâmetro x pode representar qualquer membro do conjunto domínio

Função Simples Mais Formalmente

$cubo : \mathbb{R} \rightarrow \mathbb{R}$



Funções Metamáticas

- Aplicações de funções matemáticas são especificadas por um par que contém o **nome da função** com um **elemento do conjunto domínio**.

Aplicação da função $cubo(x) \equiv x * x * x$

$cubo(2.0)$ leva ao valor 8.0

- Não há parâmetro desvinculado. Todos são do conjunto domínio da função.
- Não há uma variável externa (local ou global)



Funções Metamáticas

Função Lambda

- primeiros trabalhos teóricos acerca de funções separaram a tarefa de defini-las da de nomeá-las.
- Notação criada por Alonzo Church em 1941.
- Método para definir funções sem nome: lista de parâmetros e mapeamento da função.

Expressão Lambda

$$\lambda(x) x * x * x$$

Aplicando a Expressão Lambda

$(\lambda(x) x * x * x)(2)$ resulta no valor 8



Funções de Ordem Superior

- Uma função de ordem superior, ou forma funcional, é uma que recebe funções como parâmetros ou que leva a uma função como resultado, ou ambos
- Composição de Funções

Forma Funcional

$$h \equiv f \circ g$$

Aplicando a Forma Funcional

$$f(x) \equiv 1 + x$$

$$g(x) \equiv x^2$$

então h é definida como

$$h(x) \equiv f(g(x)) \text{ ou } h(x) \equiv 1 + (x^2)$$

Funções de Ordem Superior

Aplicar-para-todos

- Forma funcional que recebe uma única função como um parâmetro. Se aplicada para uma lista de argumentos, aplicar-para-todos aplica seu parâmetro funcional para cada um dos valores no argumento lista e coleta os resultados em uma lista ou em uma sequência.
- Denotada por α .

Forma Funcional Aplicar-pata-todos

$$h(x) \equiv x * x, \text{ então}$$

$$\alpha(h, (2, 3, 4)) \equiv [h(2), h(3), h(4)] \text{ resulta em } (4, 9, 16)$$



Linguagens de Programação Funcional

- O objetivo do projeto de uma linguagem de programação funcional é mimetizar funções matemáticas ao máximo possível.
- abordagem para a solução de problemas fundamentalmente diferente de abordagens usadas com linguagens imperativas
- A execução de uma função sempre produz o mesmo resultado quando fornecidos os mesmos parâmetros. Esse recurso é chamado de **transparência referencial**.
- Há linguagens pura e não-puras ou híbridas.
 - ▶ Lisp, Scheme, Haskell, F#, Common Lisp, ML



Exemplos Funções Aplicar-para-todos



Fundamentos de Programação

Unidade 10

Noções de Programação Funcional e de
Programação Lógica

Renato Freitas

Universidade Federal do Ceará - UFC

Russas, outubro de 2024

Obrigado!