

UFSC / CTC / INE

Disciplina: Paradigmas de Programação

CCO: INE5416 / SIN:INE5636

Prof. Dr. João Dovicchi*

Roteiro 6 - Classes e Tipos

0.1 Classes em Haskell \neq Classes em OOP

Embora Haskell também possa ser orientado a objeto, o conceito de classes em Haskell têm a função de parametrizar o polimorfismo.

Parte 1.

1. Pesquise a definição de polimorfismo em linguagens de programação.
2. Diferencie o conceito de Polimorfismo Ad Hoc e o conceito de Polimorfismo Paramétrico.
3. Compare o polimorfismo em linguagens como C++, Java, Python e Haskell.
4. Compare o conceito de classe de OOP em C++ e em Haskell (não O'Haskell).
5. Observe como as classes em Haskell definem operadores e funções e tente diferenciar dos mecanismos de herança em C++ quanto aos métodos e funções.

*<http://www.inf.ufsc.br/~dovicchi> --- joao.dovicchi@ufsc.br

As classes em Haskell são construtores de tipos do sistema e oferecem suporte ao polimorfismo paramétrico.

Parte 2.

1. Verifique nas definições de classes em Haskell, como são organizadas as instâncias e como funciona as declarações de tipos.
2. Observe a hierarquia das classes em Haskell e a herança entre elas quanto à tipagem de funções e argumentos.
3. Compare os métodos em C++, em Python e em Haskell. Observe que Haskell não tem a noção de objeto nem alteração de estado.
4. Observe como os métodos podem ser mais seguros e "error free" já que são fortemente tipados. (compare com as declarações funcionais em Python).
5. Estudo como funcionam as estruturas algébricas "Listas" e "Tuplas" (n-uplas).
6. Funções em Haskell são tipos de dados abstratos, assim como `Maybe`, `Either`, `Ordering` e `()`.

Obs.: Os tipos `Maybe`, `Either` serão vistos depois em Mônadas e Functores.