

# Roteiro 10 - Paradigmas de Programação

Emmanuel Podestá Junior

Novembro 2015

## 1 Parte 1

- Homologia Álgebraica é o estudo de funtores homológicos e as estruturas que eles englobam. Elas extraem informações de cadeias complexas e apresentam-as em forma de invariantes, módulos ou espaços topológicos. Um espaço topológico, por exemplo, pode ser definido como um conjunto de pontos e seus vizinhos, que satisfazem um conjunto de axiomas.
- Teoria das Categorias é um ramo da matemática que formaliza uma estrutura matemática e seus conceitos em termos de objetos e flechas. Um Functor, neste ramo, pode ser definido como uma estrutura que é preservada entre mapeamentos de diferentes categorias. De forma simplificada, podemos pensar em um functor como um homomorfismo entre categorias.
- Em teoria das categorias, uma Mônada é um functor que mapeia uma categoria para ela mesma, junto com duas transformações naturais. Em programação funcional, mônada é uma estrutura que representa computações definidas em um conjunto de passos. Isso possibilita a construção de pipelines, que permitem o processamento de informação por passos.
- A classe Monad em Haskell pode ser pensada como uma descrição composta de computações. A sua essência é a suparação da composição da linha de tempo e a habilidade da computação carregar, implicitamente, dados extras. A classe Functor é usada para tipos que podem ser mapeados.

## 2 Parte 2

- É possível notar a abordagem do código em Haskell por mônadas, desviando de uma divisão por zero no "rst". Contudo, ainda precisamos utilizar um if para construir o comportamento do "//". Caso algum dos operadores seja igual à zero, é retornado o tipo **Nothing**. O tratamento da divisão por zero torna-se mais prática, pois não é preciso que o programador necessariamente trate essa divisão, o programa irá tratar por conta própria de acordo com o que foi estabelecido, utilizando o **Just** e o **Nothing**.