Precedências em Haskell

Paradigmas da Computação - Atividade 8, questão 5

**Felipe de Campos Santos - 17200441**

**Universidade Federal de Santa Catarina**

**05/03/2021**

## Operadores de precedência

Operadores de precedência em haskell são funções descritas de uma maneira padrão para este fim, como equações, sendo inteiramente descritos por símbolos (ao contrario de funções comuns que são alfanuméricas), como por exemplo os operadores de concatenação em listas:

| (++) :: [a] -> [a] -> [a] [] ++ ys = ys (x:xs) ++ ys = x : (xs++ys) |
| --- |

E como operadores de precedência são funções, é válido tambem declarar eles parciamente, ou seja, em *seções*:

| (x+) = \y -> x+y (+y) = \x -> x+y (+) = \x y -> x+y |
| --- |

assim temos uma cobertura de casos melhor. No exemplo acima, sabemos que “x+” seguido de um y qualquer, deve retornar x+y, assim como “+y” precedido por um x retorna o mesmo, e “+” entre um x e um y também.

## Declarações *fixit*

Estas podem ser associadas à qualquer declarador de precedência, dando a eles uma “ordem de importância”, principalmente associados à equações. Por exemplo, sabemos que matematicamente em uma equação (sem parenteses) a multiplicação vem antes da soma, então a declaração fixit de uma multiplicação está associada à um nível maior de importância do que uma declaração fixit de soma. Os níveis vao de 0 a 9, sendo 9 o mais importante.

| **infix** 5 + **infix** 7 \* |
| --- |