

Exercícios Práticos 2

- 1) Um método bastante conhecido para encontrar aproximações de raízes de uma função real é o método de Newton. O método consiste em, dada uma função $f : \mathbb{R} \rightarrow \mathbb{R}$ diferenciável, realizar uma estimativa inicial x_0 para a raiz e, a partir disso, realizar sucessivas aproximações melhores para a raiz seguindo a seguinte definição recursiva:

$$x_{n+1} = x_n - f(x_n)/f'(x_n)$$

Implemente uma função `newton` que recebe uma função `f :: Double -> Double`, uma estimativa inicial `x0 :: Double` e uma precisão `eps :: Double` e retorna uma boa aproximação para uma raiz de `f`. Uma boa aproximação, neste caso, será o primeiro x_n tal que $|x_n - x_{n-1}| < \text{eps}$.

Considere a seguinte aproximação para a derivada da função `f`:

$$f'(x_0) = (f(x_0 + \text{delta}) - f(x_0)) / \text{delta}, \text{delta} = 1\text{e-}6$$

- 2) Determine o tipo das funções abaixo mostrando os passos até obter o resultado. Caso não seja possível determinar o tipo, explique por quê.

- a) `(.) thrice map`
- b) `swap map thrice`
- c) `tail . head`

Dados:

```
(. ) :: (b -> c) -> (a -> b) -> a -> c
map  :: (a -> b) -> [a] -> [b]
thrice :: (a -> a) -> a -> a
thrice f x = f (f (f x))
swap  :: a -> (a -> b) -> b
swap f g = g f
```

- 3) Em 2018, a escolha do melhor filme do Oscar seguiu um sistema de votação preferencial, que funciona da seguinte forma: cada votante faz um ranking dos indicados, de acordo com sua preferência. Se o filme que aparece **mais** vezes em primeiro lugar tem **mais** da metade dos primeiros lugares, ele é o vencedor. Caso contrário, o filme com **menos** primeiros lugares é eliminado (ou seja, retirado dos rankings de cada votante) e o processo é repetido com os novos rankings até que se obtenha um vencedor.

Suponha que um ranking de um votante seja representado por uma lista de filmes, em que o filme na posição 0 é o primeiro lugar. Por exemplo, a lista `["The Shape of Water", "Dunkirk", "Get Out"]` significa que o jurado

escolheu The Shape of Water como melhor filme, Dunkirk como segundo melhor filme e Get Out como terceiro melhor filme.

Escreva uma função `winner` que, dado uma lista de listas de filmes representando os rankings de cada jurado, indique quem foi o filme vencedor. Se houver dois filmes empatados com menos primeiros lugares, elimine aquele com menos segundos lugares. Se novamente os dois filmes empataram, elimine aquele com menos terceiros lugares e assim por diante. Considere que sempre será possível realizar o desempate.

```
winner [[ "The Shape of Water", "Dunkirk", "Get Out" ],  
        [ "Get Out", "Dunkirk", "The Shape of Water" ],  
        [ "Dunkirk", "The Shape of Water", "Get Out" ]  
= "Dunkirk"
```

Os exercícios práticos devem ser realizados individualmente e enviados por e-mail com o assunto [IF686EC] EXERCÍCIOS PRÁTICOS 02 para monitoria-if686-ec-l@cin.ufpe.br até as 23:59 de sexta (06.04.2018).

As resoluções dos exercícios devem estar em arquivos diferentes, um arquivo por exercício com os nomes no formato Q[número do exercício].hs