

Avaliação da I Unidade

Paradigma Funcional utilizando Haskell

Entregar código fonte, os testes e a documentação
Os testes devem ser colocados em um arquivo separado
O discente deverá implementar todas as funções
(não utilizar funções prontas biblioteca)

- 1) Elaborar em Haskell um programa para calcular a quantidade de dias entre duas datas

```
calcularDias :: [String] -> Int

> calcularDias [ "10/03/1950", "28/11/1972" ]
8299
```

- 2) Elaborar em Haskell um programa para contar o número de caracteres, palavras, linhas, espaços, vogais e números de um texto

```
type NumeroCaracteres = Int
type NumeroPalavras = Int
type NumLinhas = Int
type NumEspacos = Int
type NumVogais = Int
type QuantNumeros = Int
contar :: String -> [ (NumeroCaracteres,NumeroPalavras,NumLinhas,NumEspacos,NumVogais,QuantNumeros) ]

> contar "Alonzo Church"
[ (13,2,1,1,4,0) ]
```

- 3) A cifra de Vigenère é um método de criptografia que usa uma série de diferentes cifras de César. Pesquise sobre criptografia e a cifra de Vigenère e implemente o algoritmo em Haskell.

```
type TextoClaro = String
type Chave = String
type TextoCifrado = String

> vigenere :: TextoClaro -> Chave -> TextoCifrado

vigenere "atacar" "oi"
"obokoz"
```

- 4) Elaborar em Haskell um programa para receber um número N e escrever o mesmo por extenso. Considere $N \geq 0$ e $N < 10.000$

```
extenso :: Int -> String

> extenso 3453
"três mil quatrocentos e cinquenta e três"
```