

Ensino de Programação – MEINF

Teste de avaliação

11 de Fevereiro de 2014 (10h00)

Dispõe de **2:00 horas** para realizar este teste.

Questão 1 (8v = 2+2+2+2)

Responda às alíneas seguintes:

- a) Defina uma função que calcula a lista com todos os divisores de um dado número inteiro. Por exemplo:

```
divisores :: Int -> [Int]

> divisores 18
= [1,2,3,6,9,18]
```

- b) Especifique uma função que calcula o máximo divisor comum de dois números inteiros. Por exemplo:

```
mdc :: Int -> Int -> Int

> mdc 18 24
= 6
```

- c) Especifique uma função que calcula a interseção de duas listas de inteiros. Por exemplo:

```
intercepcao :: [Int] -> [Int] -> [Int]

> intercepcao [1,2,3,6,9,18] [1,2,3,4,6,8,12,24]
= [1,2,3,6]
```

- d) Especifique uma função que calcula o máximo divisor comum de uma lista de inteiros. Exemplo:

```
mdclist :: [Int] -> Int

> mdclist [24,9,18]
= 3
```

Questão 2 (12v = 2+2+2+2+2+1+1)

Considere as seguintes definições de tipos de dados para os tipos numéricos fração e número misto. Uma fração é composta por um par de inteiros, o numerador e o denominador. Por sua vez, um número misto é constituído por um inteiro e uma fração própria (o numerador é menor do que o denominador):

```
data Frac = Fracao Int Int
  deriving Show

data NumMisto = NMisto Int Frac
  deriving Show
```

e especifique as funções que se descrevem a seguir:

- a) Especifique uma função que recebe uma fração e calcula a sua forma reduzida:

```
reduzFrac :: Frac -> Frac

> reduzFrac (Fracao 4 8)
= Fracao 1 2
```

b) Especifique um função que recebe uma fração e a converte num número misto:

```
frac2Misto :: Frac -> NumMisto
```

```
> frac2Misto (Fracao 22 6)
```

```
= NMisto 3 (Fracao 4 6)
```

c) Especifique a função inversa da anterior, ou seja, recebe um número misto e transforma-o numa fração (na sua forma mais reduzida):

```
misto2Frac :: NumMisto -> Frac
```

```
> misto2Frac (NMisto 3 (Fracao 4 6))
```

```
= Fracao 11 3
```

d) Especifique uma função que calcula a soma de duas frações:

```
somaFrac :: Frac -> Frac -> Frac
```

```
> somaFrac (Fracao 1 8) (Fracao 4 6)
```

```
= Fracao 19 24
```

e) Especifique uma função que calcula a soma de dois número mistos:

```
somaMisto :: NumMisto -> NumMisto -> NumMisto
```

```
> somaMisto (NMisto 3 (Fracao 4 6)) (NMisto 1 (Fracao 2 5))
```

```
= NMisto 5 (Fracao 1 15)
```

f) Especifique uma função que calcula o somatório de uma lista de frações e apresenta o resultado na forma de uma fração reduzida:

```
somaLFrac :: [Frac] -> Frac
```

```
> somaLFrac [Fracao 1 2, Fracao 1 3, Fracao 1 4, Fracao 1 5]
```

```
= Fracao 77 60
```

g) Considere o seguinte tipo de dados:

```
data Fracionario = F Frac
```

```
                | N NumMisto
```

```
deriving Show
```

Especifique uma função que calcula o somatório de uma lista de elementos deste tipo:

```
somaLFracionario :: [Fracionario] -> Fracionario
```

```
> somaLFracionario [N (NMisto 2 (Fracao 2 3)), F (Fracao 1 3)]
```

```
= F (Fracao 3 1)
```