

Programação Funcional

8 de Junho de 2016

Teste 616

DCC/FCUP

Nome: _____

Nº mecanográfico: _____

- **Duração: 2h + 30m tolerância.**
- **Este teste contém 6 questões e 4 páginas.**
- **Responda às questões no espaço marcado no enunciado.**
- **Pode usar funções auxiliares e/ou do prelúdio-padrão de Haskell**

1. (30%) Responda a cada uma das seguintes questões, indicando **apenas** o resultado de cada expressão.

(a) `drop 4 [1,3,4,5,6,9]` = _____

(b) `"abc":[[]] ++ "dce":[]` = _____

(c) `length (['a', 'b'] : ['3', 'a'] : ['b'] : [])` = _____

(d) `zipWith (+) [0,2..] [1,3..10]` = _____

(e) `map (>5) [2,4..10]` = _____

(f) `foldr (==) False [True,False,True]` = _____

(g) Sem usar explicitamente a lista dada, defina a seguinte lista em compreensão:

`[2,-3,4,-5,6,-7,8,-9,10,-11]` = _____

(h) Considere a seguinte definição em Haskell:

```
h = let f (y:z) = f z - y
      f _      = 0
      in f [3,2,1]
```

A avaliação da expressão `h` tem como resultado: _____

(i) Indique um tipo admissível para `filter (>2)`: _____

(j) Indique um tipo admissível para `[id, not]`: _____

(k)

```
data E a = C a | D (E a) (E a)
eval (C c) f = c
eval (D x y) f = f (eval x f) (eval y f)
```

O tipo inferido pelo Haskell para a função `eval` é: _____

(l) Indique o tipo mais geral de `(f x n = x!!n)` :

2. (15%) Acerca de listas.

- (a) Defina uma função `somavalor`, que dada uma lista de inteiros e um valor `x`, determina a lista obtida da lista dada, somando `x` a cada elemento.
- (b) Defina a função `maiorque` que dada um elemento `x` e uma lista `xs`, retorna `True` se `x` é maior do que todos os elementos de `xs` e `False` caso contrário.

Nota: Pode utilizar funções do prelúdio-padrão e/ou listas em compreensão mas não deve usar **recursão**.

3. (15%) Acerca de matrizes

- (a) Implemente uma função `matid` que dado um inteiro `n` produz uma matriz de dimensão `n` por `n`, com 1's na diagonal principal e 0's nas restantes posições. Deverá representar a matriz como uma lista de listas, correspondentes às linhas da matriz.
- (b) Implemente uma função `simetrica`, que dada uma matriz quadrada (representada por uma lista de listas, determina se a matriz é simétrica.

4. (15%) Listas infinitas

- (a) Defina uma função `strings` que calcula a lista infinita de todas as sequências de letras minúsculas.
- (b) Utilizando a função da alínea anterior, defina uma função `menores` que dado um inteiro `n` devolve todas as cadeias de letras minúsculas, de tamanho inferior a `n`.

5. (15%) Considere a seguinte declaração de tipo:

```
data Arv a = Folha a | No (Arv a) (Arv a)
```

- (a) Defina uma função `soma`, que dada uma árvore, calcule a soma dos valores na árvore.
- (b) Defina uma função `listar`, que dada uma árvore, devolva a lista dos elementos na árvore.

6. (10%) Responda (**apenas**) a uma das seguintes alíneas, usando indução matemática.

- (a) Considerando as funções definidas na questão anterior, mostre que para qualquer árvore t , $\text{sum } (\text{listar } t) = \text{soma } t$.

Sugestão: demonstre o seguinte resultado adicional $\text{sum } (xs ++ ys) = \text{sum } xs + \text{sum } ys$.

- (b) Usando a seguinte definição da função `concat`:

```
concat :: [[a]] -> [a]
concat [] = []
concat (x:xs) = x ++ concat xs
```

mostre que, $\text{map } f (\text{concat } xs) = \text{concat } (\text{map } (\text{map } f) xs)$.

Nota: pode utilizar qualquer propriedade da função `map` que tenha sido demonstrada nas aulas, ou demonstrar qualquer resultado adicional que facilite a prova.