

Métodos de programação III

LMCC & LESI, Universidade do Minho

Ano lectivo 2004/2005

Ficha Teórico-Prática Nº1

Este texto está escrito em **literate Haskell**. Isto é, pode ser interpretado como um documento \LaTeX ou como um puro programa na linguagem Haskell. O documento é obtido "compilando" este ficheiro em \LaTeX : `latex ficha1.lhs`. O programa Haskell incluído neste ficheiro é compilado/interpretado directamente pelos compiladores/interpretadores de Haskell. Para isso o código Haskell deve estar delimitado pelo comando `begin` e `end code`. Responda às perguntas sobre Haskell neste próprio ficheiro e assim produzir o programa e a sua documentação.

1 Expressões Regulares

1.1 *Escreva expressões regulares para definir as seguintes linguagens regulares:*

1. *Frases constituídas por zero ou mais símbolos a seguidos por um ou mais símbolos b .*
2. *Frases, não vazias, constituídas por símbolos a ou b .*
3. *Linguagem cujas frases são os dígitos.*
4. *Linguagem dos números inteiros.*
5. *Linguagem que define os identificadores de linguagens de programação. Considere que um identificador começa obrigatoriamente por uma letra minúscula, e é seguida por letras ou dígitos.*
6. *Linguagem dos números reais. Considere que -3 , $+11.12$, $-.14$, 14 são frases da linguagem. Porém, $+$, $11.$, $++12$, $11.2.3$ não são frases desta linguagem.*

1.2 *Prove que a frase $abbc$ é uma frase da linguagem definida pela seguinte expressão regular: $a + (ab^+c + b^*)$*

2 Expressões Regulares em Haskell

2.1 Defina o tipo de dados algébrico *RegExp* em Haskell para representar expressões regulares.

Solução

```
--  
-- Módulo de Expressões Regulares em Haskell  
--  
-- Métodos de Programação III  
-- 2004/2005  
--  
  
module RegExp where
```

2.2 Escreva as expressões regulares do exercício 1 usando os construtores do tipo *RegExp*.

Solução

2.3 Escreva um função *showRE* que imprime uma expressão regular na sua notação tradicional. Por exemplo, se fizer o *showRE* em Hugs da primeira expressão regular escrita em Haskell no exercício anterior obtém-se:

```
RegExp> showRE zeroAsMoreBs  
"((a)*(b(b)*))"
```

Solução

2.4 Usualmente utiliza-se uma notação própria para definir algumas construções que ocorrem frequentemente em expressões regulares. Por exemplo, a expressão regular aa^* é escrita como a^+ , e $a + \epsilon$ como $a?$.

1. Re-escreva o código Haskell para permitir escrever expressões usando esta notação.
2. Re-escreva as expressões regulares que definem a linguagens dos números inteiros e reais.

Solução
