

1º Teste

Programação Funcional – 1º Ano, LEI / LCC / MIEF
26 de Novembro de 2014

Duração: 90 min

O formato **csv** (comma separated values) serve para descrever tabelas de uma forma textual: cada linha da tabela corresponde a uma linha do texto, enquanto que os elementos de cada linha se encontram separados por vírgulas.

Por exemplo, a *string* "2,3,6,4\n12,3,12,4\n3,-4,5,7" pode ser usada para descrever a matriz

$$\begin{bmatrix} 2 & 3 & 6 & 4 \\ 12 & 3 & 12 & 4 \\ 3 & -4 & 5 & 7 \end{bmatrix}$$

1. Complete a definição das função `stringToMat` e `mat2String` que convertem *strings* desse formato em matrizes e vice-versa:

```
type Mat    = [Vector]
type Vector = [Float]

stringToMat :: String -> Mat
stringToMat s = map stringToVector (lines s)

matToString :: Mat -> String
matToString m = unlines (map vectorToString m)
```

Para isso apresente as definições das funções

- (a) `vectorToString :: Vector -> String` que converte cada linha da matriz numa *String*. Por exemplo, `vectorToString [2,3,6,4]` deve dar como resultado a *string* "2,3,6,4"
- (b) `stringToVector :: String -> Vector` que converte cada linha de texto numa lista de *Floats*. Por exemplo, `stringToVector "2,3,6,4"` deve dar como resultado a lista `[2,3,6,4]`.

Use as funções `read` e `show` que permitem converter entre *strings* e *Floats*.

2. A **norma** (ou comprimento) de um vector $x = (x_1, x_2, \dots, x_n)$ define-se como

$$\bar{x} = \sqrt{x_1^2 + x_2^2 + \dots + x_n^2}$$

Defina a função `norma :: Vector -> Float` que calcula a norma de um vector. Use de preferência funções de ordem superior.

3. Defina a função `sortByNorm :: Mat -> Mat` que, dada uma matriz, reorganiza as suas linhas de forma a elas aparecerem por ordem crescente da sua norma.
4. Para representar matrizes esparsas (em que o número de elementos nulos é muito grande) é costume usar outra representação em que só se guardam os elementos não nulos da matriz (bem como a respectiva posição). Considere então o seguinte tipo alternativo para representar matrizes:

```
type MatE = [(Pos,Float)]
type Pos = (Int,Int) -- indice da linha e indice da coluna
```

Defina as funções de conversão entre as duas representações, i.e., as funções

```
matMatE :: Mat -> MatE
matEMat :: MatE -> Mat
```