Programação Funcional
6ª Aula — A Cifra de César

Pedro Vasconcelos
DCC/FCUP

2014

A cifra de César

- Um dos métodos mais simples para codificar um texto.
- Cada letra é substituida pela que dista k posições no alfabeto.
- Quando ultrapassa a letra 'z', volta à letra 'a'.
- Utilizada pelo imperador Júlio César (100 AC–44 AC).

Exemplo: para $k = 3$, a substituição é:

```
a b c d e f g h i j k l m n o p q r s t u v w x y z
d e f g h i j k l m n o p q r s t u v w x y z a b c
```

Logo, "`ataque`" é codificado como "`dwdtxh`".

Problema

Escrever uma função

```
cifrar :: Int -> String -> String
```

para implementar a cifra de César com um deslocamento dado (Exercício 3.1).

O módulo *Data.Char*

Vamos usar algumas funções sobre caracteres definidas no módulo *Data.Char*, e.g.:

```
ord :: Char -> Int
```
— código numérico dum caracter

```
chr :: Int -> Char
```
— caracter dum código numérico

Para usar este módulo, colocamos a seguinte declaração no programa:

```
import Data.Char
```

Resolução

Começamos por definir duas funções de conversão entre as letras 'a'...'z' e os inteiros no intervalo 0...25.

```
let2int :: Char -> Int
let2int x = ord x - ord 'a'

int2let :: Int -> Char
int2let n = chr (n + ord 'a')
```

NB: Estas funções assumem que os argumentos estão nos intervalos certos!

Resolução (cont.)

Definimos agora uma função para deslocar *k* posições no alfabeto as letras minúsculas; outros caracteres ficam inalterados.

```
deslocar :: Int -> Char -> Char
deslocar k x
     | minuscula x = int2let ((let2int x+k)`mod`26)
     | otherwise   = x

minuscula :: Char -> Bool
minuscula x = x>='a' && x<='z'
```

Resolução (cont.)

A cifra de César é definida aplicando a função *deslocar* a cada caracter da cadeia dada.

```
cifrar :: Int -> String -> String
cifrar k xs = [deslocar k x | x<-xs]
```

Resolução (cont.)

Também podemos usar deslocamentos negativos; por exemplo, para descodificar uma mensagem cifrada com *cifrar k* usamos *cifrar* $(-k)$:

```
> cifrar 3 "haskell e' fixe"
"kdvnhoo h' ilah"
> cifrar (-3) "kdvnhoo h' ilah"
"haskell e' fixe"
```

Quebrar a cifra

Vamos agora ver como quebrar a cifra, isto é, encontrar o deslocamento usado para cifrar uma mensagem.

Quebrar a cifra (cont.)

As letras do alfabeto têm frequências relativas características de cada língua; para o Português (em percentagens):

```
-- frequencia relativa das letras 'a'..'z'
tabela :: [Float]
tabela = [13.9, 1.0, 4.4, 5.4, 12.2, 1.0,
          1.2, 0.8, 6.9, 0.4, 0.1, 2.8, 4.2,
          5.3, 10.8, 2.9, 0.9, 6.9, 7.9, 4.9,
          4.0, 1.3, 0.0, 0.3, 0.0, 0.4]
```

Fonte: `http://www.ncc.up.pt/~rvr/Main/TabelasLP.html`.

Quebrar a cifra (cont.)

Plano:

1. Calcular as frequências relativas no texto cifrado
2. Deslocar a tabela $0 \ldots 25$.
3. Escolher o deslocamento que melhor corresponde à frequência do Português.

Quebrar a cifra (cont.)

Um método *standard* para comparar *frequências observadas* $[o_1, \ldots o_n]$ com *frequências esperadas* $[e_1, \ldots e_n]$ é o teste chi-quadrado: quanto menor for o valor de

$$\sum_{i=1}^{n} \frac{(o_i - e_i)^2}{e_i}$$

melhor é a correspondência.