

Processamento de Linguagens e Compiladores 3º ano  
**Expressões Regulares e GAWK**  
Grupo 14

Artur Queiroz  
A77136

Rafael Fernandes  
A78242

Rafaela Pinho  
A77293

15 de Outubro de 2017

## **Resumo**

Este trabalho foca os conceitos básicos do funcionamento GAWK e das expressões regulares utilizadas para descrever padrões. Neste relatório descrevemos as decisões tomadas e as dificuldades encontradas, bem como pequenos exemplos para que qualquer um que o leia perceba facilmente como funciona o nosso projeto.

# Conteúdo

<b>1</b>	<b>Introdução</b>	<b>2</b>
<b>2</b>	<b>Ficheiros de internacionalização</b>	<b>3</b>
2.1	Descrição informal do problema . . . . .	3
2.2	Especificação do Requisitos . . . . .	3
<b>3</b>	<b>Concepção/desenho da Resolução</b>	<b>4</b>
3.1	Estruturas de Dados . . . . .	4
3.2	Algoritmos . . . . .	4
<b>4</b>	<b>Codificação e Testes</b>	<b>5</b>
4.1	Problemas de implementação, Decisões e Alternativas . . . . .	5
4.1.1	Problemas de implementação . . . . .	5
4.1.2	Decisões . . . . .	5
4.1.3	Alternativas . . . . .	5
4.2	Testes realizados e Resultados . . . . .	5
<b>5</b>	<b>Conclusão</b>	<b>8</b>
<b>6</b>	<b>Código do Programa</b>	<b>9</b>

# Capítulo 1

## Introdução

Neste trabalho vamos filtrar vários ficheiros de internacionalização, utilizando expressões regulares e o sistema de filtragem de texto GAWK. Uma parte importante do sistema linux são as expressões regulares e a capacidade de procurar num input um determinado padrão, o que facilita muitas das operações. As expressões regulares são uma sequência de caracteres, sucinta e flexível, que identifica palavras ou padrões de caracteres. Combinando-as com o sistema de filtragem, conseguimos extrair informações dos ficheiros POs que são documentos que contêm traduções de uma língua para outra. Este trabalho tem como objetivos aprofundar o conhecimento do GAWK, do sistema Linux e as suas ferramentas, bem como aprender melhor o funcionamento das expressões regulares.

## Capítulo 2

# Ficheiros de internacionalização

### 2.1 Descrição informal do problema

É necessária um "script" que permita:

- a) Analisar vários ficheiros de formato PO, e devolver o número de traduções e os seus tradutores, bem como alguns metadados acharmos relevantes.
- b) Através de ficheiros PO's criar dicionários de triangulação de Português para Francês.
- c) Reformatar ficheiros de formato PO pra LaTeX.

### 2.2 Especificação do Requisitos

Os requisitos mínimos deste trabalho são saber utilizar Linux, bash e GAWK.

## Capítulo 3

# Concepção/desenho da Resolução

### 3.1 Estruturas de Dados

### 3.2 Algoritmos

Utilizamos as seguintes expressões regulares:

- i) `/^msgid/`
- ii) `/^"Language:/`
- iii) `<.*@.*> / && !/^"Report/ && !/FIRST/ && !/^"Language/`
- iv) `/^"Language-Team: *\n"/`
- v) `/^"Language-Team:/`
- vi) `/portugu[êe]se?s?/`
- vii) `/^msgid[ \t]*/`
- viii) `/^msgstr[ \t]*/`
- ix) `/^msgid */`
- x) `/^msgstr */`

## Capítulo 4

# Codificação e Testes

### 4.1 Problemas de implementação, Decisões e Alternativas

#### 4.1.1 Problemas de implementação

De um modo geral no trabalho, tivemos problemas com a formatação de alguns ficheiros, dado não se encontrarem na formatação prevista, e com o LaTeX.

#### 4.1.2 Decisões

No problema 1 proposto no exercício, decidimos ignorar os "msgid" que não contivessem nenhuma mensagem ou que contivessem mais de uma tradução. Escolhemos como metadados mais importantes a linguagem, a percentagem de blocos, a quantidade de blocos por ficheiro, e quantos e quem são os tradutores. Decidimos também que a linguagem só está no "Language". No problema 2 decidimos utilizar a "Language-team" para definir a linguagem de tradução, pois nenhum dos ficheiros apresentava linguagem no campo "Language". Usamos uma matriz descritiva, sendo as linhas a frase em inglês e os descritivos das colunas a linguagem portuguesa ou francesa. Assumimos só a linguagem portuguesa e a francesa e imprimimos unicamente o que tem tradução em triangulação. No último problema ignoramos o "

n" e o "#", devido ao LaTeX, substituindo por e por -, respetivamente.

#### 4.1.3 Alternativas

Uma alternativa seria modificar os ficheiros manualmente de modo a estarem no formato certo. Outra alternativa seria fazer inúmeros casos especiais.

### 4.2 Testes realizados e Resultados

Mostram-se a seguir alguns testes feitos (valores introduzidos) e os respectivos resultados obtidos:

```
Terminal
artur ~/UM/ano3/PLC/PLC/trabalhoP(git: master)
$ gawk -f ex *.po
Quantidade de blocos: 3744
Percentagem de blocos com traducao: 89.7703%
Ha 9 tradutores, eles sao:
    Pedro Morais
    José Nuno Pires
    David Barzilay
    Tiago Pasqualotto
    Pedro Macedo
    Rodrigo Padula de Oliveira
    Fabio Gomes
    Antonio S. de A. Terceiro
    carlinhos ceconi
Ha 1 linguagens, elas sao:
    pt
artur ~/UM/ano3/PLC/PLC/trabalhoP(git: master)
$
```

Figura 4.1: Exercicio 1

```
Terminal
artur ~/UM/ano3/PLC/PLC/trabalhoP(git: master)
$ gawk -f ex2 PT/*.po FR/*.po > dicionario_PT-FR
artur ~/UM/ano3/PLC/PLC/trabalhoP(git: master)
$ head dicionario_PT-FR
PT      ->      FR
Konqueror -> Konqueror
desenvolvedor (suporte a SSL) -> développeur (Gestion de SSL)
Dois Pares -> Deux paires
KATômico -> KAtomic
Geral -> Général
&Nível -> Niveau
desenvolvedor (JavaScript) -> développeur (JavaScript)
Mediano -> Moyen
Mostrar título -> Afficher l'ombre des pièces
artur ~/UM/ano3/PLC/PLC/trabalhoP(git: master)
$
```

Figura 4.2: Exercicio 2



```

artur ~/UM/ano3/PLC/PLC/trabalhoP(git: master)
$ gawk -f ex3 constelacoes.po > output_ex3.tex
artur ~/UM/ano3/PLC/PLC/trabalhoP(git: master)
$ latex output_ex3.tex | tail

Underfull \hbox (badness 10000) in paragraph at lines 5--384

Underfull \hbox (badness 10000) in paragraph at lines 5--384

[1] [2] [3] [4] [5] [6] [7] [8] [9] (./output_ex3.aux) )
(see the transcript file for additional information)
Output written on output_ex3.dvi (9 pages, 10220 bytes).
Transcript written on output_ex3.log.
artur ~/UM/ano3/PLC/PLC/trabalhoP(git: master)
$ dvi2pdf output_ex3.dvi
artur ~/UM/ano3/PLC/PLC/trabalhoP(git: master)
$ evince output_ex3.pdf

```

Figura 4.3: Exercício 3-terminal



Figura 4.4: Exercício 3-resultado

## Capítulo 5

# Conclusão

Com este trabalho concluímos que as expressões regulares são excelentes auxiliares na procura do que pretendemos em vários ficheiros simultaneamente. Para além disso, este trabalho permitiu a familiarização com o LaTeX, já que até então não era uma ferramenta utilizada regularmente. Como perspetiva futura, e mencionado anteriormente, sugerimos a alteração dos ficheiros que não estão com o formato correto, de modo a não conterem "lixo" no resultado da nossa procura. Esta opção seria a mais sensata, dado que criar casos especiais para esses ficheiros é um procedimento mais complexo.

## Capítulo 6

# Código do Programa

O exercício 2.5 a) está resolvido no ficheiro ex.

O exercício 2.5 b) está resolvido no ficheiro ex2.

O exercício 2.5 c) está resolvido no ficheiro ex3.