

Universidade do Minho

Trabalho Prático II

Mestrado Integrado em Engenharia Informática

Processamento de Linguagens.

2º Semestre — 2018/2019

a77523 - Alexandre Martins
a74814 - João Bernardo Freitas
a74806 - João Amorim

21 de Abril 2019
Braga

Resumo

Este projeto tem como objetivo a interação dos alunos com a ferramenta *GAWK* permitindo o aumento da capacidade destes relativamente à escrita de expressões regulares para filtrar e transformar textos. Este relatório pretende sumarizar os esforços efetuados para alcançar o objetivo proposto utilizando a ferramenta *GAWK*.

Contents

1	Introdução	2
1.1	Estrutura do Relatório	2
2	Problema Proposto	2
2.1	Descrição do problema	2
3	Solução	3
3.1	Resposta às questões do enunciado	3
4	Conclusão	10

1 Introdução

1.1 Estrutura do Relatório

O enunciado foi atribuído de acordo com o menor número de aluno de todos os elementos do grupo. Como tal o relatório debruçar-se-à sobre o enunciado número dois **Processador de Processos de Formação**.

Começaremos por introduzir o problema proposto, passaremos para a concepção da solução e, finalmente, iremos realizar uma análise crítica ao trabalho elaborado.

Na secção enunciada como **Problema Proposto** apresentaremos o problema retratando o formato sobre o qual iremos ter de extrair informação.

Na secção **Solução**, iremos abordar os métodos utilizados na resposta ao enunciado.

Finalmente, na secção **Conclusão** enumeraremos as dificuldades sentidas na resolução do enunciado, mencionando como foram ultrapassadas.

2 Problema Proposto

2.1 Descrição do problema

O nosso problema prático aborda arquivos **CSV**, *Comma Separated Values*, que são arquivos de texto separam valores por vírgulas, usado normalmente em *software* como **Microsoft Excel** e **LibreOffice Calc**.

Foi-nos então apresentado um enunciado com os seguintes objetivos:

- Criar um ficheiro através do ficheiro **formacao.csv** efetuando uma limpeza desse mesmo;
- Contar o número de registos com código numérico e mostrando para esses o código, o título, a descrição e as notas;
- Identificar os tipos diferentes e calcular o número de processos por tipo;
- Desenhar um grafo em DOT que relacione cada código com os diplomas jurídico-administrativos usados.

Para a resolução destas questões foi nos fornecido, pelos docentes, o seguinte ficheiro:

formacao.csv.

3 Solução

3.1 Resposta às questões do enunciado

- Alínea A)

Para a solução da *alínea A*), é necessário:

- Remover todas as linhas em branco;
- Remover todas as linhas com todos os campos vazios;
- Sempre que encontrar um **record** com o primeiro campo vazio, esse deve tomar o valor **NIL**

Após leitura do ficheiro reparamos que, para facilitar a leitura, é necessário remover todos os **newlines** do ficheiro, com exceção dos que estão no fim de um **record**.

Reparamos também que o ficheiro tinha 27 **fields** por **record**, tinha como **File Separator FS** e **Record Separator RS**, ; e ;;;;;;\r\n respectivamente. Começamos então por definir essa informação num ficheiro *.gawk*.

```
BEGIN {FS=";" ;NF=27;RS=";;;;;;\r\n";}
```

Para remover as linhas vazias e facilitar a futura leitura decidimos remover todos os \r\n do ficheiro, para tal utilizamos a função gsub.

```
{gsub("\r\n","")}
```

Para removermos as linhas que têm todos os campos vazios criamos um ciclo **for** que percorre todos os campos de um **record**, se todos os campos estiverem vazios ignora-se a linha através da keyword **next**.

```
{for(i=1;i<=NF;i++){if($i==""){k=0}else{k=1;break};if(k==0) next}}
```

Temos agora que adicionar mais uma vez \r\n ao ficheiro, só que desta vez vamos só colocar no fim de cada **record**. Para tal criamos outro ciclo **for** que percorre todos os campos de um **record** e adiciona \r\n no fim.

```
{for(i=1;i<=NF ;i++){if(i==NF) printf "\r\n"}}
```

Por fim, para escrever o **record** inteiro, adicionando **NIL** ao primeiro campo se ele estiver vazio, criamos mais dois ciclos **for**.

```
$1!="" {for(i=1;i<=NF ;i++){printf $i";";}next}  
$1=="" {printf "NIL;";for(i=2;i<=NF;i++){printf $i";";}next}
```

Temos então que a resposta a alínea A é obtida através do seguinte programa AWK.

- Alínea B)

Para a solução da *alínea B)* era necessário apresentar os códigos numéricos com seu título, descrição e notas respectivas através do ficheiro obtido na alínea anterior.

Após leitura do ficheiro reparamos que os campos pertinentes á resolução desta alínea são

- Campo nº2-Código
- Campo nº4-Título
- Campo nº3-Descrição
- Campo nº27-Nota

Visto que existe a possibilidade de alguns dos campos estarem vazios, decidimos que nesses casos vamos escrever **N/A**.

Temos então o seguinte programa **.gawk**.

```
BEGIN {FS=" ";NF=27;RS="\r\n";}
NR>=3 {
    if($2=="")$2="N/A";
    if($3=="")$3="N/A";
    if($4=="")$4="N/A";
    if($27=="")$27="N/A";
    printf "CODIGO->"$2
        "\nTITULO->"$3"\nDESCRICAO->"$4"\nNOTA->"$27"\n-----
        -----\n";total++}
END {printf "\nNumero total de registos:" total "\n"}
```

Obtemos então a resposta á alínea B através do seguinte comando.

```
gawk -f TP2B.gawk <limpo.csv
```

```

CODIGO->750.30.602
TITULO->Reconhecimento, creditação e validação de competências e qualificações
DESCRICÃO->"Ações de validação e valorização de conhecimentos, aptidões, competências e qualificações adquiridas pela experiência de ensino, laboral e de vida, através da atribuição de equivalência ou reconhecimento de um grau de habilitação académica ou profissional.Inicia com a verificação e análise do percurso formativo e termina com a definição da qualificação.Inclui elaboração do portefólio individual que explicita e organiza as evidências das competências adquiridas."
NOTA->N/A
-----
CODIGO->750.30.602.01
TITULO->Reconhecimento, creditação e validação de competências e qualificações: verificação das condições
DESCRICÃO->"Inicia com a verificação e análise do percurso formativo e termina com relatório preliminar.Inclui elaboração do portefólio individual que explicita e organiza as evidências das competências adquiridas."
NOTA->"Critério de densidade informacional:Enumeração e tipificação dos casos recuperável através das estatísticas.Esta síntese não permite recuperar o particular e individual, mas apenas o geral.A aplicação de um critério de amostragem aleatória sobre este PN permitirá conhecer em detalhe alguns casos e obter um subconjunto representativo de todas as características da população-alvo.Toma ainda como base o facto da imprevisibilidade dos estudos nesta área e a impossibilidade actual de conservar toda a produção.A constituição de uma amostra significativa do universo de onde é retirada, mediante a aplicação de métodos estatísticos com uma margem de erro controlada, será suficiente para representar o universo.A grelha para seleção da amostragem encontra-se no final desta página.Extra amostra poderão ser conservadas os fat files ou os processos referentes a casos com maior impacto social."
-----
CODIGO->750.30.602.02
TITULO->Reconhecimento, creditação e validação de competências e qualificações: atribuição de equivalência ou reconhecimento
DESCRICÃO->"Inicia com análise do relatório preliminar da análise do percurso formativo e termina com a definição da qualificação.Inclui reuniões e elaboração de pareceres."
NOTA->"Critério de densidade informacional:Informação não recuperável noutro PN."
-----
Numero total de registos é:29

```

Figure 2: Resposta da alínea B.

- Alínea C)

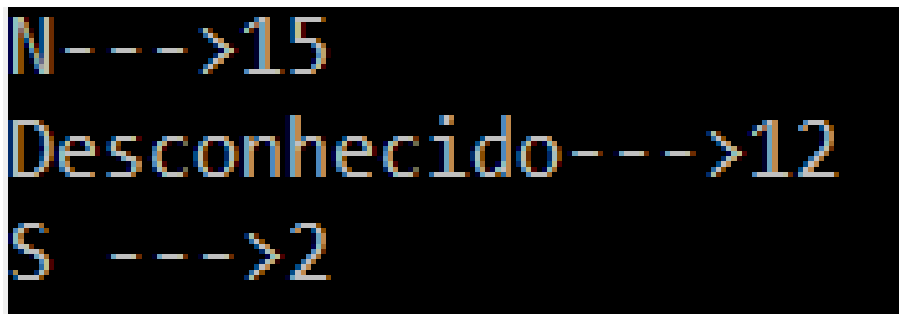
Para a resolução da *alínea C)* é necessário identificar todos os tipos de processos e contar o número de processos por tipo. Após leitura do ficheiro reparamos que o tipo está no campo **11**.

Decidimos então que a maneira mais fácil de resolver esta alínea seria através de um *array*.

```
BEGIN {FS=" ";NF=27;RS="\r\n";}
NR>=3 {tipo[$11]++}
END {for(x in tipo){if(x!=""){print x"--->" tipo[x]}else{print
      "Desconhecido--->" tipo[x]}}}
```

Através do seguinte comando obtemos a resposta.

```
gawk -f TP2C.gawk <limpo.csv
```



```
N--->15
Desconhecido--->12
S --->2
```

Figure 3: Resposta da alínea C.

- Alínea D)

Para esta alínea decidimos que era necessário a criação de um ficheiro **.dot** através do ficheiro *limpo.csv*.

Para tal o formato do ficheiro tem que ser:

```
digraph{
rankdir=LR
    a->b
}
```

Para escrevermos as duas primeiras e última linhas definimos o seguinte:

```
BEGIN {FS="";NF=27;RS="\r\n";print "digraph{";print "rankdir=LR"}
END{print "}"}
```

Para obtermos os pares código-diplomas começamos por analisar o ficheiro. Após a análise verificamos que os campos pertinentes a esta alínea eram:

- Campo nº2-Código
- Campo nº8-Diplomas jurídico-administrativos REF
- Campo nº9-Diplomas jurídico-administrativos complementar

Devido às inconsistências relativas á utilização de " decidimos que era melhor remover todas as ocorrências desse mesmo através da seguinte função:

```
{gsub("\", "")}
```

Reparamos também que era possível que os campos referentes aos diplomas estarem vazios e que um campo poderia ter um ou mais diplomas. Como tal, para além de filtrar os campos sem diplomas, decidimos usar a função **split**, que irá separar uma *String* em *Arrays* de *Strings* de acordo com um separador dado como input. Neste caso o separador é #.

```
NR>=3 {if($8!=""){split($8,lei,"#");for(i=1;lei[i]!=NULL;i++){printf
    $2"->\\"lei[i]";\n}}}}
NR>=3 {if($9!=""){split($9,lei,"#");for(i=1;lei[i]!=NULL;i++){printf
    $2"->\\"lei[i]";\n}}}}
```

Temos então que o programa **.gawk** final é:

```
BEGIN {FS="";NF=27;RS="\r\n";print "digraph{";print "rankdir=LR"}
    {gsub("\", "")}
NR>=3 {if($8!=""){split($8,lei,"#");for(i=1;lei[i]!=NULL;i++){printf
    $2"->\\"lei[i]";\n}}}}
NR>=3 {if($9!=""){split($9,lei,"#");for(i=1;lei[i]!=NULL;i++){printf
    $2"->\\"lei[i]";\n}}}}
END{print "}"}
```

Através do seguinte comando:

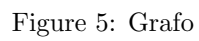
```
gawk -f TP2D.gawk <limpo.csv >grafo.dot
```

Obtemos o seguinte ficheiro:

```
digraph{
rankdir=LR
750.10.001->"Lei 45/2013";
750.10.001->"Lei 45/2012";
750.10.001->"DL 92/2011";
750.10.001->"Lei 60/2011";
750.10.001->"Portaria 181-D/2015";
750.10.001->"Lei 9/2009";
750.10.001->"Lei 2/2008";
750.10.001->"Portaria 1500/2007";
750.10.001->"Portaria 832/2007";
750.10.001->"Portaria 401/2007";
750.10.001->"DL 24/2006";
750.10.001->"DL 174/2001";
750.10.001->"Lei 166/99";
750.10.001->"DL 70-A/2000";
750.10.001->"DL 174/2001";
750.10.001->"DL 4/98";
750.10.001->"DL 48/86";
750.10.001->"DL 50/98 alterado pelos DL 70-A/2000 e DL 174/2001";
750.10.002->"Portaria 1141/2005";
750.10.002->"DL 174/2001";
750.10.002->"DL 29/2001";
750.10.002->"DL 70-A/2000";
750.10.002->"DL 174/2001";
750.10.002->"DL 48/86";
750.10.002->"DL 50/98 alterado pelos DL 70-A/2000 e DL 174/2001";
750.10.300->"Lei 51/2012";
750.10.300->"Lei 49/2005";
750.10.300->"DL 50/98";
750.10.300->"Lei 47/86";
750.10.300->"Lei 21/85";
750.10.300->"DL 174/2001";
750.10.300->"DL 29/2001";
750.10.300->"DL 48/86";
750.10.300->"Portaria 213/2009";
750.10.300->"Portaria 400/2007";
750.10.600->"DL 174/2001";
750.10.600->"Lei 2/2008";
750.10.601->"DL 174/2001";
750.10.602->"DL 29/2001";
750.10.602->"DL 3/2008";
750.20.001->"DL 91/2013";
750.20.001->"DL 139/2012";
750.20.001->"DL 6/2001";
750.20.001->"DL 48/86";
750.20.001->"Lei 5/97";
750.20.002->"Despacho Normativo 13/2014";
750.20.002->"DL 79/2014";
750.20.002->"DL 91/2013";
```

Figure 4: Conteúdo do ficheiro grafo.dot.

```
dot -Tpdf grafo.dot -o grafo.pdf
```



4 Conclusão

Visto que o objectivo deste trabalho era aumentar contacto com a ferramenta *GAWK*, capacidade de escrever *Expressões Regulares*, desenvolver *Processadores de Linguagens Regulares* e introduzir os alunos á escrita de grafos em **DOT** podemos afirmar que foi um sucesso.

Porém, no decorrer da resolução deste trabalho, foram surgindo alguns desafios que foram ultrapassados, nomeadamente: o formato dos *newlines* dos ficheiros.csv, bem como o tratamento da informação recolhida através da ferramenta **GAWK**.

Em suma, o trabalho foi bastante benéfico para todos os elementos, sendo que aprofundou o entendimento das expressões regulares e pressupostos da ferramenta *GAWK*.