Processamento de Linguagens e Compiladores
LCC+MiEFis (3ºano)

Trabalho Prático nº 1 (GAWK)

Ano lectivo 17/18

1 Objectivos e Organização

Este trabalho prático tem como principais **objectivos**:

- aumentar a experiência de uso do ambiente Linux e de algumas ferramentas de apoio à programação;
- aumentar a capacidade de escrever *Expressões Regulares (ER)* para descrição de *padrões de frases*;
- desenvolver, a partir de ERs, sistemática e automaticamente *Processadores de Linguagens Regulares*, que filtrem ou transformem textos;
- utilizar o sistema de produção para *filtragem de texto* GAWK.

Para o efeito, esta folha contém 4 enunciados, dos quais deverá resolver um escolhido em função do número do grupo (NGr) usando a fórmula $exe = (NGr \ \% \ 5) \ + \ 1$.
Neste 1º TP que se pretende que seja resolvido rapidamente (1 semana), os resultados pedidos são simples e curtos. Aprecia-se a imaginação/criatividade dos grupos ao incluir outros processamentos!

Deve entregar a sua solução **até FIXME**
O programa desenvolvido será apresentado aos membros da equipa docente, totalmente pronto e a funcionar (acompanhado do respectivo relatório de desenvolvimento) e será defendido por todos os elementos do grupo, em data a marcar.
O **relatório** a elaborar, deve ser claro e, além do respectivo enunciado, da descrição do problema, das decisões que lideraram o desenho da solução e sua implementação (incluir a especificação GAWK), deverá conter exemplos de utilização (textos fontes diversos e respectivo resultado produzido). Como é de tradição, o relatório será escrito em LaTeX.

2 Enunciados

Para sistematizar o trabalho que se pede em cada uma das propostas seguintes, considere que deve, em qualquer um dos casos, realizar a seguinte lista de tarefas:

1. Especificar os padrões de frases que quer encontrar no texto-fonte, através de ERs.
2. Identificar as acções semânticas a realizar como reacção ao reconhecimento de cada um desses padrões.
3. Identificar as Estruturas de Dados globais que possa eventualmente precisar para armazenar temporariamente a informação que vai extraindo do texto-fonte ou que vai construindo à medida que o processamento avança.
4. Desenvolver um Filtro de Texto para fazer o reconhecimento dos padrões identificados e proceder à transformação pretendida, com recurso ao Sistema de Produção GAWK.

2.1 Processador de CETEMPúblico

Ver ficheiro `natura.di.uminho.pt/~jj/pl-18/TP1/CORPORA1/`. Genericamente os corpora agrupam (grandes quantidades) de textos aos quais adicionam informação de anotação frásica (parágrafos(`<p>`), frases(`<s>`), multi-word-expressions (`<mwe>`)), e morfossintática (exemplo: lema, categoria gramatical, etc).

O formato CETEMPúblico, usa tags xml para a anotação frásica, e colunas separadas por tab para a informação morfossintática de cada palavra. As colunas presentes são: palavra, secção, semestre, lema, pos(part of speech), tempoVerbal-modo, num-pessoa, Género , árvore, etc.

Analise alguns extractos.

Construa um ou mais programas Awk que processem o CETEMPúblico de modo a:

- contar o número de Extratos, Paráfrafos e Frases.
- extrair a lista das multi-word-expressions e respectivo número de ocorrências.
- calcule a lista dos verbos PT: (Lema, para palavras com pos=V) e respectivo número de ocorrências.
- determinar o dicionário implícito no córpora – calcule a lista das palavras associando-lhes os possíveis (lema, pos)

2.2 Processador de textos preanotados com Freeling

Ver ficheiro `natura.di.uminho.pt/~jj/pl-18/TP1/CORPORA2/`. Genericamente os corpora agrupam (grandes quantidade) de textos aos quais adicionam informação de anotação frásica e morfossintática (exemplo: lema, categoria gramatical, etc).

O formato Freeling, usa separa extratos com uma linha em branco, e usa colunas separadas por espaços para a informação morfossintática de cada palavra. As colunas presentes são: num, palavra, lema, pos-tag, pos(part of speech), features, ..., árvore.

Analise alguns extractos.

Construa um ou mais programas Awk que processem corpora freeling de modo a:

- contar o número de Extratos.
- calcule a lista dos personagens do Harry Potter (nomes próprios) e respectivo número de ocorrências.
- calcule a lista dos verbos, substantivos, adjectivos e advérbios PT: e crie um ficheiro com cada uma destas listas.
- determinar o dicionário implícito no córpora – lista contendo os lema, pos e palavras dele derivadas.

2.3 Processador / sincronizador de Legendas

Ver ficheiro `natura.di.uminho.pt/~jj/pl-18/TP1/SUBTITLES/`
Considere o seguinte extracto de legendas formato srt:

```
1
00:00:48,344 --> 00:00:49,500
Chamada: recebida

2
00:00:49,707 --> 00:00:53,128
-Está tudo pronto?
-Você não tinha de me substituir.

3
00:00:53,328 --> 00:00:56,014
Eu sei, mas quero fazer um turno.

4
00:01:06,485 --> 00:01:08,943
Você gosta dele, não?
Gosta de observá-lo.
```

As linhas 1, 5, 10, 14 contêm os identificadores de legenda. As linhas 2, 6, 11, 15 contêm os tempos de início e desaparecimento da legenda. As legendas são separadas por linha em branco.
Considere ainda que dispomos legendas do mesmo filme em várias línguas, mas que frequentemente diferem no tempo inicial e duração do filme.

1. Construa um sincronizador de legendas:

   ```
   srtsync a.srt b.srt a1 b1 a2 b2 > b-sync.srt
   ```

   que recalcule os tempos de entrada e saída das legendas de b.srt de modo que as legendas com números b1 e b2 de b fiquem com as entradas sincronizadas respectivamente com as legendas a1 e a2 de a.

2. Construa um processador de srt que:

   - retire os identificadores de legenda.
   - coloque as legendas numa única linha juntando-as com "|"
   - marque com traço horizontal os intervalos com mais de 2 segundos de silêncio.

   ```
   00:00:53,328 --> 00:00:56,014 Eu sei, mas quero fazer um turno.
   00:00:56,014 --> 00:01:06,485 ================================
   00:01:06,485 --> 00:01:08,943 Você gosta dele, não?|Gosta de observá-lo.
   ```

2.4 Processador de Thesaurus 1

Ver ficheiros em `natura.di.uminho.pt/~jj/pl-18/TP1/THE/` que pode corrigir/completar. Os ficheiros fornecidos `....mdic` descrevem numa sintaxe simples as entradas (triplos `termo1, rel, termo2`) de um Thesaurus que se pretende criar automaticamente.
Cada ficheiro mdict contem:

- comentários a ignorar (do símbolo cardinal, até ao fim da linha)
- directivas gerais:
  - `%dom: alimentação` – todos os termos definidos são do domínio *alimentação*; válido até nova indicação de novo domínio. Dom é uma relação e a sua inversa é voc(vocabulário).
  - `%inv: nt : bt` – indica que a relação *nt* (=narrow term), é a inversa *bt* (=broader term).
- tabelas de relações constituídas por uma linha indicadora de relações (começada por `%THE`), seguida de várias linhas com tuplos.

```
%inv:atravessa : é_atravessado_por
%inv:tem_como_instancia : iof
%THE : nt
bebida    : vinho | sumo
sobremesa : pudim | fruta | leite creme |baba de camelo

%THE<rio : nasce_em : atravessa < localidade : foz
rio Cávado : serra do Larouco: Montalegre|Barcelos : Esposende
```

Note que:

- Cada linha tem 1 ou mais termos.
- Os termos são separados por ':' daqueles com que se relacionam.
- A relação entre o termo da coluna 1 e os termos da coluna n é a indicada na posição n da linha `%THE` (nt, instancia, iof, etc.).
- Quando há vários termos com a mesma relação, eles podem ser agrupados com '—'.
- Um campo do cabeçalho pode conter `< classe`, indicando que todos os elementos dessa coluna são intâncias da *classe*. Exemplo `(rio Cávado, iof, rio)(Montalegre, iof, localidade)`

Exemplo de alguns triplos decorrentes do exemplo anterior:

```
(bebida, nt, vinho)(bebida, nt, sumo) (rio Cávado, iof, rio)
(rio, tem_como_instancia, rio Cávado)
(rio Cávado, nasce_em, serra do Larouco) (rio Cávado, atravessa, Montalegre)
(Montalegre, é_atravessado_por, rio Cávado)
```

Escreva programas awk que dado um ou mais mdic:

1. determine a lista dos domínios e das relações usadas.
2. mostre os triplos expandidos correspondentes (um triplo por linha)
3. mostre a informação contidas nos triplos, agrupadas pelo termo1 (formato Thesaurus ISO) – Exemplo de um extrato :

```
rio Cávado
iof:   rio
nasce_em:    serra do Larouco
atravessa:   Montalegre
atravessa:   Barcelos
foz:         Esposende

Barcelos
é_atravessado_por:   rio Cávado
iof: localidade
```

2.5 Processador de thesaurus 2

Usando o mesmo formato e os mesmos ficheiros mdic descritos no enunciado anterior:
Escreva programas awk que dado um ou mais mdic:

1. determine a lista dos domínios e das relações usadas.
2. mostre os triplos expandidos correspondentes (um triplo por linha)
3. Construa um conjunto de páginas HTML (uma página por cada termo1) em que os termos2 hiperliguem às correspondentes páginas.