Processamento de Linguagens

Universidade do Minho, LEI

Ano lectivo 2007/2008

Trabalho Prático N°2

Luis Tiago Mascarenhas - 38172 Mário Ulisses Pires Araujo Costa - 43175
Vasco Almeida Ferreira - 43207

23 de Julho de 2008

Resumo

O intuito deste trabalho é demonstrar os conhecimentos obtidos sobre o par gerador de compiladores lex/yacc na geração de parsers utilizando Gramáticas Tradutoras.

Conteúdo

1 Introdução

O trabalho consiste na geração de um site para as localidades de Portugal face às hierarquias geográficas definidas à custa de uma linguagem (OrgGeo).

Apresentamos de seguinda a gramática dessa linguagem:

```
1  OrGeo     --> Distritos
2
3  Distritos --> Distrito '|'
4            | Distritos Distrito '|'
5
6  Distrito  --> IdD Link Concelhos
7
8  Concelhos --> Concelho
9            | Concelhos '' Concelho
10
11 Concelho  --> Locais '!' IdC Link
12
13 Locais    --> Local
14           | Locais ',' Local
15
```

```
16 Link      --> '>' l
17
18 IdD       --> id
19
20 IdC       --> id
21
22 Local     --> IdL Link
23
24 IdL       --> id
```

2 Descrição do Problema

Como queriamos um ficheiro de grande dimensão para testar e também ter um resultado mais interessante e fiável decidimos começar este trabalho por fazer um programa em C que gerasse um ficheiro que encaixe na gramática acima descrita.
Inicialmente precisavamos de uma "base de dados" que tivesse, com grande fidelidade todas as freguesias de Portugal; descobrimos um pack de 3 ficheiros no site dos CTT (Correios), que passamos a descrever de seguida:

Ficheiro **distritos.txt** (28 linhas)

```
1 ID_DISTRITO1:NOME_DISTRITO1
2 .
3 .
4 .
5 ID_DISTRITOn:NOME_DISTRITOn
```

exemplo:

```
1 03;Braga
2 04;Braganca
3 05;Castelo Branco
4 06;Coimbra
```

Ficheiro **concelhos.txt** (307 linhas)

```
1 ID_DISTRITO:ID_CONCELHO1:NOME_CONCELHO1
2 .
3 .
4 .
5 ID_DISTRITO:ID_CONCELHOn:NOME_CONCELHOn
```

exmplo:

```
1 18;10;Oliveira de Frades
2 18;11;Penalva do Castelo
3 18;12;Penedono
4 18;13;Resende
5 18;14;Santa Comba Dao
6 18;15;Sao Joao da Pesqueira
7 18;16;Sao Pedro do Sul
```

Ficheiro **todos_cp.txt** tem informação referente a todos os códigos postais do país (274537 linhas)

```
1 ID_DISTRITO;ID_CONCELHO;ID_FREGUESIA1;NOME_FREGUESIA1;.+
2 .
3 .
4 .
5 ID_DISTRITO;ID_CONCELHO;ID_FREGUESIAn;NOME_FREGUESIAn;.+
```

exemplo:

```
01;01;249;Alcafaz;;;;;;;;;;;3750;011;AGADAO
01;01;250;Caselho;;;;;;;;;;;3750;012;AGADAO
01;01;251;Corga da Serra;;;;;;;;;;;3750;013;AGADAO
01;01;252;Foz;;;;;;;;;;;3750;014;AGADAO
01;01;253;Guistola;;;;;;;;;;;3750;015;AGADAO
01;01;254;Guistolinha;;;;;;;;;;;3750;016;AGADAO
01;01;255;Lomba;;;;;;;;;;;3750;017;AGADAO
01;01;256;Povinha;;;;;;;;;;;3750;018;AGADAO
01;01;257;Vila Mendo;;;;;;;;;;;3750;019;AGADAO
01;01;60359;Felgueira;;;;;;;;;;;3750;020;AGADAO
01;01;60560;Boa Aldeia;;;;;;;;;;;3750;021;AGADAO
```

Nesta fase não escolhemos fazer com o Lex pois pareceu-nos que iamos usar um tanque de guerra para matar uma formiga.

2.1 Optimização da "base de dados"

Como queriamos ter informação sobre cada localidade de Portugal decidimos que teriamos que ter links para a Wikipedia, pois só esta teria a informação mais fiel e completa que conseguiamos "encontrar".

Aproveitamos os conhecimentos ganhos com o primeiro trabalho, sobre o conhecimento dos links da Wikipedia, para usar o dump que tinhamos usado na primeira fase. Detectamos alguns padrões:
Para os Distritos:

- Distito_d[eao]_NOMEDISTRITO

Para os Concelhos:

- NOMECONCELHO
- NOMECONCELHO_(Concelho)

Para os Distritos:

- NOMEFREGUESIA
- NOMEFREGUESIA_(freguesia)
- NOMEFREGUESIA_(CONCELHO_A_QUE_PERTENCE)

Inicialmente pensamos em fazer o download de cada página da Wikipedia com os nomes das localidades do ficheiro dos CTT, mas iriamos perder muito mais tempo, possivelmente iria ser bem mais fiel, mas impraticável.

Decidimos usar o dump da Wikipedia, um ficheiro com 2.1Gb (perto de 2 milhões de linhas) e através do comando:

```
cat wikipediaPT.xml | grep "<title>.*</title>" | \
                      sed '/.*[!"#\$&:%;,=*+|0-9].*$/d' | \
                      sed -e '/[A-Z][A-Z].*/d' > titles.txt
```

Sacamos só os titulos (link) de cada artigo, eliminamos também tudas as entradas que tinham números, alguns sinais de pontuação entre outras optimizações. Desta forma geramos o ficheiro titles.txt com 645012 linhas, bem mais pequeno do que o que tinhamos inicialmente.

Tudo isto para, no programa em C podermos procurar se determinada localidade tem uma entrada na Wikipedia. Isto é feito utilizando um simples system do C, decidimos utilizar este método (nada inteligente) pois tentamos previamente fazer uma solução com forks, mas estes demoravam

mais tempo a ser criados do que a fazer realmente trabalho útil.

Assim sendo, com a ajuda deste programa conseguimos ter um ficheiro que respeita a gramática em cima anunciada;

```
Beja>http://pt.wikipedia.org/wiki/Distrito_de_Beja
        Aljustrel>http://pt.wikipedia.org/wiki/Aljustrel_(freguesia),
        Carregueiro>http://pt.wikipedia.org/wiki/Carregueiro,
        Corte Vicente Anes>http://pt.wikipedia.org/wiki/Corte_Vicente_Anes,
        Estacao Caminhos de Ferro>http://pt.wikipedia.org/wiki/Estacao_Caminhos_de_Ferro,
        Focinho de Cao>http://pt.wikipedia.org/wiki/Focinho_de_Cao,
.
.
.
        !Aljustrel>http://pt.wikipedia.org/wiki/Aljustrel;
        Almodovar>http://pt.wikipedia.org/wiki/Almodovar_(freguesia),
        Candenilhas>http://pt.wikipedia.org/wiki/Candenilhas,
        Cerca da Junqueira>http://pt.wikipedia.org/wiki/Cerca_da_Junqueira,
        Corte de Baixo>http://pt.wikipedia.org/wiki/Corte_de_Baixo,
        Corte Zorrinho>http://pt.wikipedia.org/wiki/Corte_Zorrinho,
        Corvatos>http://pt.wikipedia.org/wiki/Corvatos,
.
.
.
        !Vidigueira>http://pt.wikipedia.org/wiki/Vidigueira|

Aveiro>http://pt.wikipedia.org/wiki/Distrito_de_Aveiro
        Alcafaz>http://pt.wikipedia.org/wiki/Alcafaz,
        Caselho>http://pt.wikipedia.org/wiki/Caselho,
        Corga da Serra>http://pt.wikipedia.org/wiki/Corga_da_Serra,
        Foz>http://pt.wikipedia.org/wiki/Foz,
        Guistola>http://pt.wikipedia.org/wiki/Guistola,
        Guistolinha>http://pt.wikipedia.org/wiki/Guistolinha,
        Lomba>http://pt.wikipedia.org/wiki/Lomba,
.
.
.
```

3 Gramática tradutora

```
OrGeo     : Distritos { d = $1; }
          ;
```

A estrutura Distrito é uma lista ligada que guarda a árvore de toda a informação contida no OrGeo.

```
Distritos : Distrito '|' { $$ = $1; }
          | Distritos Distrito '|' { $$ = catDistritos($1,$2); }
          ;
Distrito  : IdD Link Concelhos { $$ = addDistrito($1, $2, $3); }
          ;
```

Os Distritos são uma lista ligada dos mesmos.
As funções catDistrito recebe uma lista ligada, um id e um Link e constrói uma lista ligada de distritos, com inserção á cabeça.

```
Concelhos : Concelho { $$ = $1; }
          | Concelhos ';' Concelho { $$ = catConcelhos($1, $3); }
          ;
Concelho  : Locais '!' IdC Link { $$ = addConcelho($1, $3, $4); }
          ;
```

O que acontece com os Concelhos é a mesma coisa que nos distritos.

```
Locais    : Local { $$ = $1; }
          | Locais ',' Local { $$ = catFreguesias($1, $3); }
```

```
3               ;
4   Link        : '>' l { $$ = $2; }
5               ;
6   IdD         : id { $$ = $1; }
7               ;
8   IdC         : id { $$ = $1; }
9               ;
10  Local       : IdL Link { $$ = addFreguesia($1, $2); }
11              ;
12  IdL         : id { $$ = $1; }
13              ;
```

4 Conclusão

Foi muito importante, depois de saber usar analisadores lexicos, termos aprendido uma ferramenta mais alto nível, com suporte a gramáticas.
Todos os conceitos dados nas aulas foram aplicados com sucesso neste trabalho final.
Devido á rapidez com que terminamos o trabalho ficamos com a sensação de que nos podia ter sido exigido mais.

O que aprendemos com esta cadeira foi de tal forma interessante que elementos do grupo aprofundaram conhecimentos nesta área e avançaram para outros geradores de compiladores, como o **ANTLR**.

Resumidamente achamos que na cadeira poderia ter sido dada mais importância á teoria de parsers, por forma a complementar esta parte prática.