

Universidade do Minho
Escola de Engenharia

Mestrado Integrado em Engenharia Informática

Processamento de Linguagens

Relatório do Trabalho Prático 1

75248 Ana Sofia Gomes Marques
A65277 Flávio Manuel Machado Martins

Grupo 27

4 Abril 2020

Conteúdo

1	Introdução	i
2	Descrição do problema	ii
3	Implementação	iii
3.1	Padrões identificados e respectivas ERs	iii
3.1.1	Início comentário	iii
3.1.2	User	iii
3.1.3	Date e timestamp	iv
3.1.4	Comentário	iv
3.1.5	Respostas	v
3.1.6	Comentário final	v
3.2	Desenvolvimento	vi
3.2.1	Contextualização Inicial	vi
3.2.2	Start Conditions	vii
4	Resultados	x
5	Conclusão	xiv

1 Introdução

Este relatório visa apresentar as decisões tomadas na realização do primeiro trabalho prático da Unidade Curricular de Processamento de Linguagens.

Procuramos assim justificar todas as ponderações feitas na formulação do problema, assim como a abordagem tomada, de forma a cumprir com as tarefas que nos foram propostas que visam na secção seguinte deste relatório.

Optamos por escolher o enunciado 4, uma vez que, á semelhança do 5 e 6, eram os mais semelhantes com aquilo que tínhamos feito nas aulas práticas da disciplina, só que desta vez não para um ficheiro xml mas sim html.

Na realização deste projeto construimos um analisador léxico, com recurso a flex, e com a utilização de expressões regulares(ERS) capazes de desenvolver de forma sistemática e automaticamente Processadores de Linguagens Regulares.

Este teve ainda a ajuda de extratos de código, desenvolvido na linguagem de programação c, de forma a que fossem extraídos do ficheiro HTML fornecido certos parâmetros, que depois foram transformados no formato JSON pretendido.

2 Descrição do problema

Para este primeiro projeto optamos assim pela realização do enunciado 4, o Transformador Público2NetLang, que se apresenta abaixo:

2.4 Transformador Público2NetLang

Analise com todo o cuidado o ficheiro http://natura.di.uminho.pt/~jj/pl-20/TP1/Publico_extraction_portuguese_comments_4.html o qual contém os comentários (85 neste exemplo) a uma notícia publicada no jornal O Público, extraídos da página HTML da versão online do dito jornal.

Para se fazer um estudo sócio-linguístico de forma e conteúdo dos comentários que a notícia suscitou, os dados relevantes à análise pretendida devem ser extraídos do ficheiro HTML fornecido e devem ser transformados no formato JSON a seguir mostrado.

```
"commentThread": [
  {
    "id": "STRING",
    "user": "STRING",
    "date": "STRING",
    "timestamp": NUMBER,
    "commentText": "STRING",
    "likes": NUMBER,
    "hasReplies": TRUE/FALSE,
    "numberOfReplies": NUMBER

    "replies": [ ]
  },.....
]
```

Construa então um filtro de texto, recorrendo ao gerador FLex, que realize o processamento explicado, tendo em consideração que as respostas¹ que surjam a um dado comentário devem ser aninhadas, na forma de uma lista, dentro do campo "replies" do dito comentário, seguindo evidentemente o mesmo formato apresentado².

Figura 1: Enunciado escolhido pelo grupo

Para qualquer um dos enunciados escolhidos foi nos sugerido que tivéssemos em conta a realização das seguintes 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 extraíndo 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 Gerador FLex .

3 Implementação

3.1 Padrões identificados e respectivas ERs

```
<li class="comment" data-comment-id="6f949889-2606-4749-1c42-08d7471cb23d">
  <div class="comment_inner">
    <div class="comment_meta">
      <span class="avatar comment_avatar">
        <span class="avatar_pad">
          
        </span>
      </span>
      <h5 class="comment_author">
        <a href="https://natura.di.uminho.pt/utilizador/perfil/275d9ced-3b68-4dc4-bd47-441c76edf850" rel="nofollow">Joao Vieira de Sousa </a>
      </h5>
      <span class="comment_reputation comment_reputation-r2" title="Experiente"><i aria-hidden="true" class="i-check"></i></span>
      <!--<span class="comment_location">Cruz Quebrada Dafundo</span-->
      <time class="datetime comment_datetime" datetime="2019-10-02T22:50:07.00">
        <a class="comment_permalink">02.10.2019 22:50</a>
      </time>
    </div>
    <div class="comment_content">
      <p>
        Pois foi o Rio tirou-lhe a teta e ele agora tenta vingar-se, mas palpita-me que não terá muita sorte
      </p>
    </div>
  </div>
</li>
<ol class="comments_list">
  <li class="comment" data-comment-id="93506ff9-5244-4c37-455e-08d7471e83ae">
    <div class="comment_inner">
      <div class="comment_meta">
        <span class="avatar comment_avatar">
          <span class="avatar_pad">
            
          </span>
        </span>
        <h5 class="comment_author">
          <a href="https://natura.di.uminho.pt/utilizador/perfil/ad91ec31-3d7e-45f5-afd3-01905b95c359" rel="nofollow">Vieira </a>
        </h5>
        <span class="comment_reputation comment_reputation-r4" title="Moderador"><i aria-hidden="true" class="i-check"></i></span>
        <!--<span class="comment_location">Cruz Quebrada Dafundo</span-->
        <time class="datetime comment_datetime" datetime="2019-10-03T00:10:15.203">
          <a class="comment_permalink">03.10.2019 00:10</a>
        </time>
      </div>
      <div class="comment_content">
        <p>
          A central de corrupcao do PSD so da' a mana aos
          apaniguados, Rui Rio nao vai fugir 'a regra ate porque a ordem dos
          contabilistas tem umas contas a acertar com o sr Rui Rio quando este la
          passou e mamou 27000 Euros por ano por 2 sessoes da assembleia geral. No
          departamento de urbanismo da CM do Porto parece que tambem se mamou la
          muito por altura do Rui Rio.
        </p>
      </div>
    </div>
  </li>
</ol>
<form class="form comments_form expanded" data-abide="" data-t="hagfba-t" style="display:none"></form>
</li>
```

Figura 2: Extrato dum exemplo de html fonte

3.1.1 Início comentário

O primeiro padrão identificado foi o de início de comentário, todos os comentários, incluindo respostas, iniciam por:

```
<li class="comment"
```

Como tal facilmente desenvolvemos uma ER capaz de identificar o início de comentários, mais concretamente:

```
<li \ class=\ "comment \ "
```

3.1.2 User

```
<h5 class="comment_author">
  <a href="https://natura.di.uminho.pt/utilizador/perfil/ad91ec31-3d7e-45f5-afd3-01905b95c359" rel="nofollow">Vieira </a>
</h5>
```

Figura 3: Exemplo de identificação do user no ficheiro html

De seguida identificamos o padrão onde podíamos encontrar o user autor do comentário.

```
<h5 class="comment__author">
```

Isto deu origem a uma ER:

```
\<h5{ws} class \="comment\_\_author"\>
```

Sendo que foi ainda verificado que seria necessário ignorar tudo que se encontra entre `<a href e rel="e o `, tendo uma condição para mudar de estado em `</h5>`, ora para estas situações foram identificadas, respectivamente, as seguintes ERs:

```
\<a{ws} href[>]+\>
```

```
\</a\>
```

```
\</h5\>
```

3.1.3 Date e timestamp

De seguida fomos procurar pela data dos comentários e foi observável o padrão muito simples:

```
<a class="comment__permalink">03.10.2019 00:10</a>
```

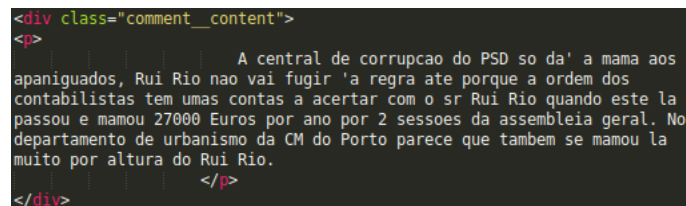
Daqui é muito fácil assumir a necessidade de três ERs, uma para identificar esta secção(`<a ... >`), uma para a data(`xx.xx.xxxx`), uma para timestamp(`xx:xx`). Como tal obtemos as ERs:

```
\<a{ws} class \="comment\_\_permalink"\>
```

```
([0-2][1-9]|3[0-1])\.(0[1-9]|1[0-2])\.[0-9]{4}
```

```
([0-1][0-9]|2[0-4]):[0-5][0-9]
```

3.1.4 Comentário



```
<div class="comment__content">
<p>
A central de corrupcao do PSD so da' a mama aos
apaniguados, Rui Rio nao vai fugir 'a regra ate porque a ordem dos
contabilistas tem umas contas a acertar com o sr Rui Rio quando este la
passou e mamou 27000 Euros por ano por 2 sessoes da assembleia geral. No
departamento de urbanismo da CM do Porto parece que tambem se mamou la
muito por altura do Rui Rio.
</p>
</div>
```

Figura 4: Exemplo de comentário no ficheiro html.

De seguida chegamos ao conteúdo do comentário em si, como podemos verificar na figura 4, precisamos de ERs para tratar do `<div class="comment__content">`, `<p>`, `</p>` e `</div>`.

```
\<div{ws} class \="comment\_\_content"\>
```

```
\<p\>
```

```
\</p\>
```

```
\</div\>
```

```

<ol class="comments_list">
  <li class="comment" data-comment-id="93506ff9-5244-4c37-455e-08d7471e83ae">
    <div class="comment_inner">
      <div class="comment_meta">
        <span class="avatar comment_avatar">
          <span class="avatar_pad">
            
          </span>
        </span>
        <h5 class="comment_author">
          <a href="https://natura.di.uminho.pt/utilizador/perfil/ad91ec31-3d7e-45f5-afd3-01905b95c359" rel="nofollow">Vieira </a>
        </h5>
        <span class="comment_reputation comment_reputation-r4" title="Moderador"><i aria-hidden="true" class="i-check"></i></span>
        <!--span class="comment_location"></span-->
        <time class="dateline comment_dateline" datetime="2019-10-03T00:10:15.203">
          <a class="comment_permalink">03.10.2019 00:10</a>
        </time>
      </div>
      <div class="comment_content">
        <p>
          A central de corrupcao do PSD so da' a mama aos
          apaniguados, Rui Rio nao vai fugir 'a regra ate porque a ordem dos
          contabilistas tem umas contas a acertar com o sr Rui Rio quando este la
          passou e mamou 27000 Euros por ano por 2 sessoes da assembleia geral. No
          departamento de urbanismo da CM do Porto parece que tambem se mamou la
          muito por altura do Rui Rio.
        </p>
      </div>
    </div>
  </li>
</ol>
<form class="form comments form_expanded" data-abide="" data-t="hagfba-t" style="display:none"></form>

```

Figura 5: Exemplo de respostas

3.1.5 Respostas

Por fim, chegamos a parte das respostas, sendo que, quando existem, as respostas são nada mais do que uma lista de comentários semelhante ao que temos até agora preparado, pelo que apenas será necessário identificar que estamos numa zona de respostas e aplicar as mesmas ERs com a devida distinção entre resposta e comentário principal. Como podemos ver na Figura 5 uma lista de respostas começa por

```
<ol class="comments__list">
```

, dando origem à ER:

```
<ol{ws}class\=\"comments\_\_list\">
```

e termina em

```
</ol>
```

dando origem a:

```
<\/ol>
```

3.1.6 Comentário final

Por fim identificamos que todos os comentários terminam na zona com o extrato:

```
form class="for comments__form_expanded"
```

que deu origem à nossa ultima ER:

```
<form{ws}class\=\"form{ws}comments\_\_form{ws}expanded\"
```

3.2 Desenvolvimento

3.2.1 Contextualização Inicial

```
%{
/* Declaracoes C diversas */
#include <stdio.h>
FILE *origem;
FILE *f;
FILE *r;
int id;
int replyid;
int noReplies;
}%
%option noyywrap
%x comment user datetime com
ws [ \t]
qname [a-zA-ZÀ-ú0-9_\.?:!\ ]*
```

Figura 6: Declarações iniciais

Na figura 6 podemos ver algumas declarações iniciais com as quais iremos trabalhar, começando pelos files, origem estará ligada ao ficheiro html utilizado como fonte, o f ao ficheiro destino e o r a um ficheiro temporario que irá guardar as respostas ao longo do processo para cada comentário. Temos tambem os ints id, correspondente ao id em que o processo se encontra de resposta, o replyid que vai contabilizando o id de respostas, e um noReplies on guardamos o número de respostas do comentário em processamento. Podemos ainda ver as ERs definidas para o espaço em ws (whitespace) e em qname podemos utilizar para encontrar frases. Por fim temos as start conditions que existem durante o processamento, nomeadamente, comment, user, datetime, com e par, cada um com regras e objetivos de processamento independentes.

```
int parse(char* source, char* destino) {
    id = 1;
    replyid = 0;
    origem = fopen(source, "r");
    if(origem) yyin = origem;
    f = fopen(destino, "w");
    r = fopen("replies.json", "w");
    if (f==NULL) {
        printf("erro na criação de ficheiro");
        return 0;
    }
    fprintf(f, "\ncommentThread\": [");
    yylex();
    fprintf(f, "\n");
    fclose(f);
    fclose(r);
    remove("replies.json");
    return id;
}

int main(int argc, char *argv[]) {
    parse(argv[1], argv[2]);

    return 0;
}
```

Figura 7: Main

Na figura 6 apresentamos o nosso main, que utiliza como auxilio a função parse, na função main chamamos o parse dando desde logo como source o 1º argumento da execução e destino o 2º. Ora no parse podemos onbserve a abertura para leitura do ficheiro origem, a abertura do ficheiro destino para escrita, e ainda a abertura dum ficheiro de replies, que irá ser um ficheiro auxiliar e apagado no final da execução. Após abertura dos ficheiros, começamos por escrever a parte inicial do ficheiro conforme o formato pedido no enunciado e iniciamos então o yylex.

3.2.2 Start Conditions

A start condition comment inicia assim que encontramos um `<li class="comment">`, utilizando a ER definida anteriormente. Quando estamos em comment iremos ter uma serie de ERs, em que basicamente iremos fazer o processamento dum comentário, incluindo respostas, até chegar ao final do mesmo, e iniciando a partir desta start condition as restantes mais importantes no projeto.

```
<comment>\<h5{ws}class=\"comment\"_\"_author\"> {
                                                    if (replyid == 0) {
                                                        noReplies = 0;
                                                        if (id>1) fprintf(f,\"\\n\");
                                                        else fprintf(f,\"\\n\");
                                                        fprintf(f,\"\\t{\\n\");
                                                        fprintf(f,\"\\t\\id\\\": \\\"%d\\\",\\n\",id);
                                                        fprintf(f,\"\\t\\user\\\": \\\" \");
                                                        id++;
                                                    } else {
                                                        if(replyid>1) fprintf(r,\"\\n\");
                                                        fprintf(r,\"\\t{\\n\");
                                                        fprintf(r,\"\\t\\id\\\": \\\"%d\\\",\\n\",replyid);
                                                        fprintf(r,\"\\t\\user\\\": \\\" \");
                                                        replyid++;
                                                    }
                                                    BEGIN user;
                                                    }
```

Figura 8: Inicio processamento user

Na figura 8 podemos ver como dentro da start condition comment, iniciamos o processamento do utilizador, imprimindo alguns dados sobre o mesmo para o ficheiro adequado e fazendo então BEGIN user.

```
<user>\<a{ws}href[~]+> {;}
<user>\</a> {;}

/* O resto é o username do user */
<user>[ \\t\\n]{2,} {fprintf(f,\" \");}
<user>{qname} {
                if (replyid == 0) fprintf(f,\"%s\", yytext);
                else fprintf(r,\"%s\", yytext);
            }

/* Fim da secção de user */
<user>\</h5> {
                BEGIN comment;
                if (replyid == 0) fprintf(f,\"\\\",\\n\");
                else fprintf(r,\"\\\",\\n\");
            }
```

Figura 9: Processamento user

De seguida, dentro da start condition user utilizamos as ERs também definidas anteriormente para identificar o username, ignorando as partes de texto descartáveis e regressando a comment ao encontrar a condição de saída previamente definida.

```

<comment>\<a{ws}class=\\"comment\_\_permalink\\">
<datetime>([0-2][1-9][3[0-1])\\.([0-9][1[0-2])\\.([0-9]{4}
{
    BEGIN datetime;
    if (replyid == 0) {
        fprintf(f, "\\t\\\"date\\\": \\\"%s\\\",\\n\", yytext);
    } else {
        fprintf(r, "\\t\\t\\\"date\\\": \\\"%s\\\",\\n\", yytext);
    }
}

<datetime>([0-1][0-9][2[0-4]):[0-5][0-9]
{
    if (replyid == 0) {
        fprintf(f, "\\t\\\"timestamp\\\": \\\"%s\\\",\\n\", yytext);
    } else {
        fprintf(r, "\\t\\t\\\"timestamp\\\": \\\"%s\\\",\\n\", yytext);
    }
    BEGIN comment;
}

```

Figura 10: Processamento date e timestamp

Novamente recorrendo apenas a ERs definidas previamente, fazemos a entrada na start condition datetime e o processamento de date e timestamp do comentário, sendo que ambas aparecem sempre por ordem regressamos a comment após processar o timestamp.

```

/*----- Comment Content----- */
<comment>\<div{ws}class=\\"comment\_\_content\\">
{
    if (replyid == 0) {
        fprintf(f, "\\t\\\"commentText\\\": \\\"\\\"");
    } else {
        fprintf(r, "\\t\\t\\\"commentText\\\": \\\"\\\"");
    }
    BEGIN com;
}
{;}
{;}
{fprintf(f, " ");}
{
    if (replyid == 0) fprintf(f, "%s", yytext);
    else fprintf(r, "%s", yytext);
}
{
    if (replyid == 0) {
        fprintf(f, "\\\",\\n");
        fprintf(f, "\\t\\\"likes\\\": 0,\\n");
    } else {
        fprintf(r, "\\\",\\n");
        fprintf(r, "\\t\\t\\\"likes\\\": 0,\\n");
        fprintf(r, "\\t\\t\\\"hasReplies\\\": FALSE\\n");
        fprintf(r, "\\t\\t\\\"numberOfReplies\\\": 0\\n");
        fprintf(r, "\\t\\t\\\"replies\\\": [ ]\\n");
        fprintf(r, "\\t\\t");
    }
    BEGIN comment;
}
}
<com>\<p>
<com>\</p>
<com>[ \\t\\n]{2,}
<com>{qname}
{
    if (replyid == 0) fprintf(f, "%s", yytext);
    else fprintf(r, "%s", yytext);
}
<com>\</div>
{
    if (replyid == 0) {
        fprintf(f, "\\\",\\n");
        fprintf(f, "\\t\\\"likes\\\": 0,\\n");
    } else {
        fprintf(r, "\\\",\\n");
        fprintf(r, "\\t\\t\\\"likes\\\": 0,\\n");
        fprintf(r, "\\t\\t\\\"hasReplies\\\": FALSE\\n");
        fprintf(r, "\\t\\t\\\"numberOfReplies\\\": 0\\n");
        fprintf(r, "\\t\\t\\\"replies\\\": [ ]\\n");
        fprintf(r, "\\t\\t");
    }
    BEGIN comment;
}
}

```

Figura 11: Processamento conteudo

Finalmente chegamos ao conteúdo do comentário, nesta secção podemos ver que utilizamos varias ERs. Ora inicialmente, ainda em comment, procuramos por `< divclass = "comment_content">`, escrevendo então no ficheiro a parte "commentText": "e fazendo de seguida o BEGIN com. Na start condition com fazemos alguns filtros para limpeza dos comentários, nomeadamente dos `<p>`, `</p>` e de onde encontremos mais do que um espaço. De seguida processamos a ER qname, definida anteriormente e que encontra frases, escrevendo o que vamos encontrando. Por fim, encontrando `</div>` sabemos que o processamento do conteúdo está terminado e fazemos a escrita correspondente no ficheiros.

```

/*----- Replies? ----- */

/* Caso encontre <ol class="comments_list">
significa que os comentários seguinte, até </ol> são respostas do comentário anterior,
portanto começamos a trabalhar com o id de resposta(reply)
*/

<comment>\<ol{ws}class=\\"comments\_list\\">      {
                                                    replyid++;
                                                    }
<comment>\</ol>                                     {
                                                    noReplies = replyid-1;
                                                    replyid = 0;
                                                    }

|

/* Chegando ao final do comentário processamos a escrita do hasReplies, numberOfReplies e das replies,
caso existam, estando escritas temporariamente num ficheiro replies.json
*/
<comment>\<form{ws}class=\\"form{ws}comments\_form{ws}expanded\\"      {
                                                                    {
if(noReplies==0) fprintf(f, "\\t\\\"hasReplies\\\": FALSE,\\n");
else fprintf(f, "\\t\\\"hasReplies\\\": TRUE,\\n");
fprintf(f, "\\t\\\"numberOfReplies\\\": %d,\\n\\n", noReplies);
fprintf(f, "\\t\\\"replies\\\": [");

fclose(r);
FILE *file;
size_t nread;
char buf[100];
file = fopen("replies.json", "r");
if (file) {
nread = fread(buf, 1, sizeof buf, file);
if (nread>0) fprintf(f, "\\n");
while (nread > 0) {
fwrite(buf, sizeof(char), nread, f);
nread = fread(buf, 1, sizeof buf, file);
}
if (ferror(file)) {
/* deal with error */
}
fclose(file);
}
if (noReplies>0) fprintf(f, "\\n\\t]\\n\\t");
else fprintf(f, " ]\\n\\t");
r = fopen("replies.json", "w");

BEGIN 0;
}

```

Figura 12: Processamento respostas

Por fim, podemos ver que utilizando novamente as ERs definidas anteriormente, quando encontramos $\langle olclass = "comments_list" \rangle$ sabemos que iremos processar respostas, ora, iremos então definir o replyid a 1, o que irá influenciar toda a escrita a passar a ser realizada para o ficheiro r, e iniciamos a sc comment, iniciando basicamente o processamento do início, isto irá correr o programa normalmente, mas basicamente escrevendo em r em vez de f, até encontra $\langle /ol \rangle$, ou seja o fim da lista, então definirá o noReplies para este comentário utilizando o replyid actual e colocará de volta o replyid a 0, para prosseguir o normal funcionamento.

Ao chegar à ER também definida previamente que encontra $\langle formclass = "formcomments_formexpanded" \rangle$ sabemos que estamos no final dum comentário, Aqui iremos verificar a existência de respostas, com recurso ao noReplies, o que irá influenciar a escrita do numberOfReplies e, de seguida fazer a leitura e escrita das respostas a este comentário, com recurso ao ficheiro temporário que criamos para o efeito. Após escrito ele é reaberto de modo a reiniciar o seu conteúdo. Por fim regressamos ao estado inicial, por estamos prontos a processar o próximo comentário.

4 Resultados

Nesta secção são apresentados alguns exemplos de utilização do nosso programa, contendo o texto fonte e o respectivo resultado, depois de aplicado o programa ao texto fonte e obtido o ficheiro json pretendido.

Aqui temos o exemplo de um comentário com 0 replies, o comentário número 50.

```

2200 <li class="comment" data-comment-id="9831cb8c-50b5-4777-276f-98b74633975a">
2201 <div class="comment_inner">
2202 <div class="comment_meta">
2203 <span class="avatar comment_avatar">
2204 <span class="avatar_size">
2205 <img alt="" data-interchange="https://static.publiccdn.com/files/homepage/img/avatar_74x74.png_small, (https://static.publiccdn.com/files/homepage/img/avatar_74x74.png retina)" data-resize="s19flh-interchange" data-t="k3lcfz-t" id="s19flh-interchange" src="" data-cs="2" data-kind="parent" data-kind="ghost">
2206 </span>
2207 </span>
2208 <div class="comment_author">
2209 <a href="/utilizador/perfil/038aeb95-64fe-4337-8bd3-e663287c91br" rel="nofollow">Manuel Cruz </a>
2210 </div>
2211 <span class="comment_reputation comment_reputation-r1" title="Iniciante"><i aria-hidden="true" class="i-check"></i></span>
2212 <!--span class="comment_location">Carcavelos</span-->
2213 <time class="comment_datetime comment_datetime">2019-10-02T08:46:30.18Z</time>
2214 <a class="comment_permalink">02.10.2019 08:46</a>
2215 </time>
2216 </div>
2217 <div class="comment_content">
2218 <p>
2219 <!--p>
2220 <div>
2221 <div>
2222 <div>
2223 <div>
2224 <div>
2225 <div>
2226 <div>
2227 <div>
2228 <div>
2229 <div>
2230 <div>
2231 <div>
2232 <div>
2233 <div>
2234 <div>
2235 <div>
2236 <div>
2237 <div>
2238 <div>
2239 <div>
2240 <div>
2241 <div>
2242 <div>
2243 <div>
2244 <div>
2245 <div>
2246 <div>
2247 <div>
2248 <div>
2249 <div>
2250 <div>
2251 <div>
2252 <div>
2253 <div>
2254 <div>
2255 <div>
2256 <div>
2257 <div>
2258 <div>
2259 <div>
2260 <div>
2261 <div>
2262 <div>
2263 <div>
2264 <div>
2265 <div>
2266 <div>
2267 <div>
2268 <div>
2269 <div>
2270 <div>
2271 <div>
2272 <div>
2273 <div>
2274 <div>
2275 <div>
2276 <div>
2277 <div>
2278 <div>
2279 <div>
2280 <div>
2281 <div>
2282 <div>
2283 <div>
2284 <div>
2285 <div>
2286 <div>
2287 <div>
2288 <div>
2289 <div>
2290 <div>
2291 <div>
2292 <div>
2293 <div>
2294 <div>
2295 <div>
2296 <div>
2297 <div>
2298 <div>
2299 <div>
2300 <div>
2301 <div>
2302 <div>
2303 <div>
2304 <div>
2305 <div>
2306 <div>
2307 <div>
2308 <div>
2309 <div>
2310 <div>
2311 <div>
2312 <div>
2313 <div>
2314 <div>
2315 <div>
2316 <div>
2317 <div>
2318 <div>
2319 <div>
2320 <div>
2321 <div>
2322 <div>
2323 <div>
2324 <div>
2325 <div>
2326 <div>
2327 <div>
2328 <div>
2329 <div>
2330 <div>
2331 <div>
2332 <div>
2333 <div>
2334 <div>
2335 <div>
2336 <div>
2337 <div>
2338 <div>
2339 <div>
2340 <div>
2341 <div>
2342 <div>
2343 <div>
2344 <div>
2345 <div>
2346 <div>
2347 <div>
2348 <div>
2349 <div>
2350 <div>
2351 <div>
2352 <div>
2353 <div>
2354 <div>
2355 <div>
2356 <div>
2357 <div>
2358 <div>
2359 <div>
2360 <div>
2361 <div>
2362 <div>
2363 <div>
2364 <div>
2365 <div>
2366 <div>
2367 <div>
2368 <div>
2369 <div>
2370 <div>
2371 <div>
2372 <div>
2373 <div>
2374 <div>
2375 <div>
2376 <div>
2377 <div>
2378 <div>
2379 <div>
2380 <div>
2381 <div>
2382 <div>
2383 <div>
2384 <div>
2385 <div>
2386 <div>
2387 <div>
2388 <div>
2389 <div>
2390 <div>
2391 <div>
2392 <div>
2393 <div>
2394 <div>
2395 <div>
2396 <div>
2397 <div>
2398 <div>
2399 <div>
2400 <div>
2401 <div>
2402 <div>
2403 <div>
2404 <div>
2405 <div>
2406 <div>
2407 <div>
2408 <div>
2409 <div>
2410 <div>
2411 <div>
2412 <div>
2413 <div>
2414 <div>
2415 <div>
2416 <div>
2417 <div>
2418 <div>
2419 <div>
2420 <div>
2421 <div>
2422 <div>
2423 <div>
2424 <div>
2425 <div>
2426 <div>
2427 <div>
2428 <div>
2429 <div>
2430 <div>
2431 <div>
2432 <div>
2433 <div>
2434 <div>
2435 <div>
2436 <div>
2437 <div>
2438 <div>
2439 <div>
2440 <div>
2441 <div>
2442 <div>
2443 <div>
2444 <div>
2445 <div>
2446 <div>
2447 <div>
2448 <div>
2449 <div>
2450 <div>
2451 <div>
2452 <div>
2453 <div>
2454 <div>
2455 <div>
2456 <div>
2457 <div>
2458 <div>
2459 <div>
2460 <div>
2461 <div>
2462 <div>
2463 <div>
2464 <div>
2465 <div>
2466 <div>
2467 <div>
2468 <div>
2469 <div>
2470 <div>
2471 <div>
2472 <div>
2473 <div>
2474 <div>
2475 <div>
2476 <div>
2477 <div>
2478 <div>
2479 <div>
2480 <div>
2481 <div>
2482 <div>
2483 <div>
2484 <div>
2485 <div>
2486 <div>
2487 <div>
2488 <div>
2489 <div>
2490 <div>
2491 <div>
2492 <div>
2493 <div>
2494 <div>
2495 <div>
2496 <div>
2497 <div>
2498 <div>
2499 <div>
2500 <div>
2501 <div>
2502 <div>
2503 <div>
2504 <div>
2505 <div>
2506 <div>
2507 <div>
2508 <div>
2509 <div>
2510 <div>
2511 <div>
2512 <div>
2513 <div>
2514 <div>
2515 <div>
2516 <div>
2517 <div>
2518 <div>
2519 <div>
2520 <div>
2521 <div>
2522 <div>
2523 <div>
2524 <div>
2525 <div>
2526 <div>
2527 <div>
2528 <div>
2529 <div>
2530 <div>
2531 <div>
2532 <div>
2533 <div>
2534 <div>
2535 <div>
2536 <div>
2537 <div>
2538 <div>
2539 <div>
2540 <div>
2
```

Figura 13: Comentário sem respostas no texto fonte

```
{
  "id": "50",
  "user": "Manuel Cruz ",
  "date": "02.10.2019",
  "timestamp": "08:46",
  "commentText": " Excelente anlise como prprio de um intrprete da cultura ",
  "likes": 0,
  "hasReplies": FALSE,
  "numberOfReplies": 0,

  "replies": [ ]
},
```

Figura 14: Resultado do programa para um comentário sem respostas

Temos abaixo o exemplo de um comentário com uma replies, o comentário número 51.

```
2128 </li>
2129 <div class="comment_inner">
2130 <div class="comment_neta">
2131 <span class="avatar comment_avatar">
2132 <span class="avatar_pad">
2133 <img alt="" data-interchange="https://static.publiccdn.com/files/homepage/img/avatar_74x74.png, small||, (https://static.publiccdn.com/files/homepage/img/avatar_74x74.png, retina)" data-resize="yqjuh2-interchange" data-t="5ppn38-t" id="yqjuh2-interchange" src="" data-cs="2" data-kind="parent" data-rs="2" style="display:none;" />
2134 </span>
2135 </span>
2136 <div class="comment_author">
2137 <a href="/utilizador/serf11/a2b439dd-2628-4a5b-91b2-7a78bd276d49" rel="nofollow">Antonio Paulo </a>
2138 </div>
2139 <span class="comment_reputation comment_reputation-r1" title="Iniciante"><i aria-hidden="true" class="i-check"></i></span>
2140 </div>
2141 <div class="comment_location"></div>
2142 <div class="comment_datetime" datetime="2019-10-02T08:41:38.033">
2143 <div class="comment_permalink">02.10.2019 08:41</div>
2144 </div>
2145 <div class="comment_content">
2146 <p>
2147 Touché. Com Rui Rio, Portugal voltaria às corridas de calhambeque tal como aconteceu em tempos à magnífica cidade do Porto....
2148 </p>
2149 </div>
2150 </div>
2151 <div class="comments_list">
2152 <li class="comment" data-comment-id="65fa2e28-76cb-44e2-3b07-88d7471dd7c5">
2153 <div class="comment_inner">
2154 <div class="comment_neta">
2155 <span class="avatar comment_avatar">
2156 <span class="avatar_pad">
2157 <img alt="" data-interchange="https://static.publiccdn.com/files/homepage/img/avatar_74x74.png, small||, (https://static.publiccdn.com/files/homepage/img/avatar_74x74.png, retina)" data-resize="q8o5lx-interchange" data-t="kgmjrx-t" id="q8o5lx-interchange" src="" data-cs="2" data-kind="parent" data-rs="2" style="display:none;" />
2158 </span>
2159 </span>
2160 <div class="comment_author">
2161 <a href="/utilizador/serf11/183a23c5-8011-4d66-ada8-271e2d2c88ce" rel="nofollow">Paulo Valente </a>
2162 </div>
2163 <span class="comment_reputation comment_reputation-r1" title="Iniciante"><i aria-hidden="true" class="i-check"></i></span>
2164 </div>
2165 <div class="comment_location"></div>
2166 <div class="comment_datetime" datetime="2019-10-02T11:08:49.893">
2167 <div class="comment_permalink">02.10.2019 11:08</div>
2168 </div>
2169 <div class="comment_content">
2170 <p>
2171 Que posso dizer? Tenho saudades das corridas de calhambeques, que tal como hoje encheram de turistas a cidade do Porto.
2172 </p>
2173 </div>
2174 </div>
2175 </li>
2176 </div>
2177 <form class="form_comments_form expanded" data-abide="" data-t="cui7d9-t" style="display:none;"></form>
2178 </div>
2179 </div>
2180
```

Figura 15: Comentário com uma resposta no texto fonte

```
{
  "id": "51",
  "user": "Antonio Paulo ",
  "date": "02.10.2019",
  "timestamp": "08:41",
  "commentText": " Touch. Com Rui Rio Portugal voltaria s corridas de calhambeque tal como aconteceu em tempos magnfica cidade do Porto.... ",
  "likes": 0,
  "hasReplies": TRUE,
  "numberOfReplies": 1,

  "replies": [
    {
      "id": "1",
      "user": "Paulo Valente ",
      "date": "02.10.2019",
      "timestamp": "11:08",
      "commentText": "Que posso dizer? Tenho saudades das corridas de calhambeques que tal como hoje encheram de turistas a cidade do Porto.",
      "likes": 0,
      "hasReplies": FALSE,
      "numberOfReplies": 0

    }
  ]
}
```

Figura 16: Resultado do programa para um comentário com uma reposta

13.

```

420 <li class="comment" data-comment-id="71b9d5b-bd8b-4ce7-bf8743683e3c">
421 <div class="comment_inner">
422 <div class="comment_meta">
423 <span class="avatar comment_avatar">
424 <span class="avatar_psd">
425 <img alt="data-interchange=https://static.publiccdn.com/files/homepage/ing/avatar_74x74.png, small], [https://static.publiccdn.com/files/homepage/ing/avatar_74x74.png, retina]" data-resize="wrn75j-interchange" data-t="77q2qb-t" id="wrn75j-interchange" src=
426 </span>
427 <div class="comment_author">
428 <a href="/utilizador/perfil/3375b39-88ee-46e9-8be6-4206f1666816" rel="nofollow">Napoleon Bonaparte </a>
429 </div>
430 <div class="comment_reputation comment_reputation-r1" title="Iniciante"><i aria-hidden="true" class="i-check"></i></span>
431 </div>
432 <div class="comment_location"></div>
433 <div class="comment_datetime" datetime="2019-10-02T20:35:54.5">
434 <a class="comment_permalink">02.10.2019 20:35</a>
435 </div>
436 <div class="comment_content">
437 <p>
438 Abruñhosa!
439 Tas-te a passar, rapaz.
440 Nem a direita é tão pior, nem os socialistas são tão bons. Estes então, são do pior que já tivemos.
441 Vê o exemplo do presidente do partido e naquilo em que o transformou...
442 O Governo é a mesma coisa: um tacho familiar...
443 Francamente, tenho vergonha deste socialismo.
444 E olha que o óssis socialista não é mais que o resultado da conjuntura económica. Quando a economia arrefecer, até os pintelhos do Centeno começam a enrolar ..
445 </p>
446 </div>
447 </div>
448 <div class="comments_list">
449 <li class="comment" data-comment-id="f4d4886e-9125-4b31-3b38-08d7471dd7c5">
450 <div class="comment_inner">
451 <div class="comment_meta">
452 <span class="avatar comment_avatar">
453 <span class="avatar_psd">
454 <img alt="data-interchange=https://static.publiccdn.com/files/homepage/ing/avatar_74x74.png, small], [https://static.publiccdn.com/files/homepage/ing/avatar_74x74.png, retina]" data-resize="ge9e83-interchange" data-t="bx9v9d-t" id="ge9e83-interchange" src=
455 </span>
456 <div class="comment_author">
457 <a href="/utilizador/perfil/ad91ec31-3e7e-45f5-af63-0190b995c339" rel="nofollow">Vieira </a>
458 </div>
459 <div class="comment_reputation comment_reputation-r4" title="Moderador"><i aria-hidden="true" class="i-check"></i></span>
460 </div>
461 <div class="comment_location"></div>
462 <div class="comment_datetime" datetime="2019-10-02T21:18:18.557">
463 <a class="comment_permalink">02.10.2019 21:18</a>
464 </div>
465 <div class="comment_content">
466 <p>
467 A direita trauliteira veio aqui confessar-se, trauliteira e ordinario, 'e a marca de agua dos apoiantes da central de corrupcao do PSD.
468 </p>
469 </div>
470 </div>
471 </li>
472 <li class="comment" data-comment-id="92164046-7055-4a2a-8a1b-08d7471b2fc4">
473 <div class="comment_inner">
474 <div class="comment_meta">
475 <span class="avatar comment_avatar">
476 <span class="avatar_psd">
477 <img alt="data-interchange=https://static.publiccdn.com/files/homepage/ing/avatar_74x74.png, small], [https://static.publiccdn.com/files/homepage/ing/avatar_74x74.png, retina]" data-resize="196bx2-interchange" data-t="u6yo0w-t" id="196bx2-interchange" src=
478 </span>
479 <div class="comment_author">
480 <a href="/utilizador/perfil/86d37218-6251-462e-84b5-96ba0549a719" rel="nofollow">António Gonçalves </a>
481 </div>
482 <div class="comment_reputation comment_reputation-r1" title="Iniciante"><i aria-hidden="true" class="i-check"></i></span>
483 </div>
484 <div class="comment_location"></div>
485 <div class="comment_datetime" datetime="2019-10-03T01:56:29.51">
486 <a class="comment_permalink">03.10.2019 01:56</a>
487 </div>
488 <div class="comment_content">
489 <p>
490 O Napoleon tem vergonha deste socialismo porque não passa de um primário fascista. E também um pobre coitado...
491 </p>
492 </div>
493 </div>
494 </li>
495 <li class="comment" data-comment-id="fc579b22-fd67-4254-3b47-08d7471dd7c5">
496 <div class="comment_inner">
497 <div class="comment_meta">
498 <span class="avatar comment_avatar">
499 <span class="avatar_psd">
500 <img alt="data-interchange=https://static.publiccdn.com/files/homepage/ing/avatar_74x74.png, small], [https://static.publiccdn.com/files/homepage/ing/avatar_74x74.png, retina]" data-resize="8uprmw-interchange" data-t="dw365u-t" id="8uprmw-interchange" src=
501 </span>
502 <div class="comment_author">
503 <a href="/utilizador/perfil/717d2d61-f346-4b88-834f-eb1798ce9987" rel="nofollow">ana cristina </a>
504 </div>
505 <div class="comment_reputation comment_reputation-r4" title="Moderador"><i aria-hidden="true" class="i-check"></i></span>
506 </div>
507 <div class="comment_location"></div>
508 <div class="comment_datetime" datetime="2019-10-03T07:46:30.793">
509 <a class="comment_permalink">03.10.2019 07:46</a>
510 </div>
511 <div class="comment_content">
512 <p>
513 uma chuva de adjetivos para quem ousa radiografar a governação costa. vieira e antónio gonzálves: ao menos, aprendam a argumentar. a era dos insultos era nos anos 70. cresçam.
514 </p>
515 </div>
516 </div>
517 </li>
518 </ol>
519 </div>
520 <div class="form_comments" form="comments" data-uid="1" data-a="Napoleon" style="display:none">

```

Figura 17: Comentário com 3 respostas no texto fonte

```

197 {
198   "id": "13",
199   "user": "Napoleon Bonaparte",
200   "date": "02.10.2019",
201   "timestamp": "20:35",
202   "commentText": "Abrunhosa!Tas-te a passar rapaz.Nem a direita to pior nem os socialistas so to bons.. Estes ento so do pior que j tivemos.",
203   "likes": 0,
204   "hasReplies": TRUE,
205   "numberOfReplies": 3,
206   "replies": [
207     {
208       "id": "1",
209       "user": "Vieira-",
210       "date": "02.10.2019",
211       "timestamp": "21:18",
212       "commentText": "A direita trauliteira veio aqui confessar-setrauliteira e ordinario e a marca de agua dos apoiantes da central de corrupcao do PSD.",
213       "likes": 0,
214       "hasReplies": FALSE,
215       "numberOfReplies": 0,
216       "replies": [ ]
217     },
218     {
219       "id": "2",
220       "user": "Antnio Goncalves",
221       "date": "03.10.2019",
222       "timestamp": "01:56",
223       "commentText": "O Napoleon tem vergonha deste socialismo porque no passa de um primrio fascista. E tambm um pobre coitado...",
224       "likes": 0,
225       "hasReplies": FALSE,
226       "numberOfReplies": 0,
227       "replies": [ ]
228     },
229     {
230       "id": "3",
231       "user": "ana cristina",
232       "date": "03.10.2019",
233       "timestamp": "07:46",
234       "commentText": "uma chuva de adjetivos para quem ousa radiografar a governao costa. vieira e antnio goncalves: ao menosaprendam a argumentar.. a era dos insultos era nos anos 70. cresam.",
235       "likes": 0,
236       "hasReplies": FALSE,
237       "numberOfReplies": 0,
238       "replies": [ ]
239     }
240   ]
241 }

```

Figura 18: Resultado do programa para um comentário com 3 replies

5 Conclusão

Em modo de conclusão consideramos que fomos capazes de realizar com sucesso este primeiro trabalho prático, cumprindo todas as tarefas que nos foram propostas.

A realização deste projeto serviu, á semelhança de outros projetos de outras cadeiras, para aumentar a experiência do uso do ambiente Linux, e ainda de algumas ferramentas de apoio à programação que nunca tínhamos usado até então, nomeadamente, o uso do Flex para gerar filtros de texto em C.

É de saliantar que a parte em que tivemos alguma dificuldade foi na escrita das Expressões Regulares, pela dificuldade na filtragem de um ficheiro html de elevada complexidade.