

UNIVERSIDADE DO MINHO

LICENCIATURA EM CIÊNCIAS DA COMPUTAÇÃO

Processamento de Linguagens e Compiladores

- Trabalho Prático 1

Grupo 3

Relatório

Alef Pinto Keuffer
(A91383)

Catarina Martins Sá Quintas
(A91650)

Ivo Miguel Gomes Lima
(A90214)

8 de novembro de 2021

Conteúdo

1	Introdução	3
1.1	Descrição do Problema	4
1.1.1	Especificação dos Requisitos	4
2	Concepção de Resolução	5
2.1	Tarefa 4 a	5
2.1.1	Implementação	6
2.2	Resultado	6
2.3	Tarefa 4 b	7
2.3.1	Algoritmos	7
2.3.2	Expressões Regulares	9
2.3.3	Implementação	9
2.3.4	Resultado	14
2.4	Tarefa 4 c	15
2.4.1	Algoritmos	15
2.4.2	Implementação	15
2.4.3	Resultado	16
2.5	Tarefa 4 d	16
2.5.1	Resultado	17
3	Conclusão	19
A	Código do Programa	20
B	Código de <i>Output</i> do <i>DOT</i>	27
C	Código do <i>Output HTML</i>	28

Capítulo 1

Introdução

No âmbito da unidade curricular Processamento de Linguagens e compiladores (PLC), foi-nos proposto a realização de um projeto de forma a aprofundar os nossos conhecimentos adquiridos na sala de aula, atingindo os seguintes objetivos:

- Aumentar a capacidade de escrever Expressões Regulares(ER)
- Desenvolver sistematicamente Processadores de Linguagens Regulares, ou Filtros de Texto.
- Familiarizar com o módulo 're' presente no Python.

Para o efeito, criamos um processador de BibTeX. A BibTeX é uma ferramenta de formatação usada em documentos em LaTeX. Um exemplo desta ferramenta pode ser visto abaixo:

```
@techreport{jspell1 ,
  author = "J.J. Almeida and Ulisses Pinto",
  title = "Manual de Utilizador do {JSpell}",
  year = 1994,
  type = "Manual",
  month = "Jul",
  institution = "umdi",
  keyword = "morphology, lexical analysis, jspell",
  abstract = {},
  url = "http://natura.di.uminho.pt/~jj/pln/jspellman.ps.gz",
}
```

Existe um conjunto de campos obrigatórios e facultativos para que um BibTeX seja válido, alguns desses campos são: *article*, *book*, *inproceedings*, *misc*, *proceedings* entre outros.

1.1 Descrição do Problema

1.1.1 Especificação dos Requisitos

A nossa solução deve satisfazer os seguintes requisitos:

- R1 Fazer a contagem das categorias presentes no documento, tais como: *phDThesis*, *Misc*, *InProceeding*, etc .
- R2 Produzir um documento em formato *HTML* com (R2.1) o nome das categorias encontradas e (R2.2) respectivas contagens.
- R3 Filtrar, para cada entrada de cada categoria, a respetiva (R3.1) chave (R3.2) autores, (R3.3) e título. (R3.4) O resultado final deverá ser incluído no documento *HTML* gerado R2.
- R4 Criar um índice de autores, que mapeie cada autor nos respectivos registos, de modo a que posteriormente uma ferramenta de procura do Linux possa fazer a pesquisa.
- R5 Construir um Grafo que mostre, para um dado autor (definido à partida) todos os autores que publicam normalmente com o autor em causa.
- R6 Recorrendo à linguagem *Dot* do *GraphViz*, gerar um ficheiro com o grafo de R5 de modo a que possa, posteriormente, usar uma das ferramentas que processam *Dot* para desenhar o dito grafo de associações de autores.

Capítulo 2

Concepção de Resolução

2.1 Tarefa 4 a

A estratégia para satisfazer R1 consistiu em ler o arquivo linha a linha verificando se a categoria encontrada já aparecia no dicionário, se já existir, irá ser incrementado o número de ocorrências, senão será adicionado como primeira ocorrência, para que depois possa ser produzido um ficheiro *HTML* com todas as categorias e o devido número de ocorrências.

Expressão Regular

De seguida apresentamos a Expressão Regular utilizada para filtrar a informação pedida em R1.

Uma vez que todas as categorias num BibTeX têm como antecedente o carácter @ e terminam numa {}, tornou-se fácil criar um filtro que guarde toda a informação delimitada entre esses dois parâmetros.

Através dessa pequena realização chegamos à seguinte expressão: $\wedge @ (.*) \{$

```
import re

file = open("exemplo-utf8.bib", "r")
read = True
dic = {}
string_ls = [ '<!DOCTYPE _HTML_PUBLIC>\n<HTML>\n_<HEAD>\n_'
               '<TITLE>Categories_in_BibTeX</TITLE>\n_</HEAD>\n_<BODY>' ]
while read:
    linhaFicheiro = file.readline()
    ncat = re.match(r'^@(.*){ ', linhaFicheiro)
    if ncat != None:
        cat_title = ncat.group(1).title()
        dic[cat_title] = dic.get(cat_title,0) + 1
    if not linhaFicheiro:
        read = False
        file.close()

time = lambda v: 's' if v > 1 else ''

for k, v in dic.items():
    string_ls.append(f'_<P>The_category_{k}_appears_{v}_\ntime{time(v)}.</P>')
    string_ls.append(f'_</BODY>\n</HTML>')
with open('output.html', 'w') as file:
    file.write('\n'.join(string_ls))
```

Number of Occurrences of Publication Types

Figura 2.1: Número de Ocorrências das Categorias

2.3 Tarefa 4 b

2.3.1 Algoritmos

Infelizmente a estratégia adotada na primeira questão de unicamente guardar aquilo que interessava tornou-se impraticável pois os parâmetros pedidos nas outras questões não tinham tanta consistência, ou seja, podiam estar envolvidos {} entre {}, podiam alterar a ordem pela qual apareciam, entre outros contratemplos.

Portanto começamos a associar os pares de (tipo da publicação, chave) com os campos da entrada e aplicar diversas manipulações sobre essa estrutura de modo a satisfazer os requisitos.

Para o efeito utilizamos vários algoritmos de agrupamento e filtragem da informação pretendida de modo a conseguir converter o ficheiro BibTeX num ficheiro *HTML*.

Todos os algoritmos tem a sua função explicitada abaixo, sendo cada um destes fundamental para a execução do programa:

L1 get_bib_str: Abre BibTeX **L2** get_pub_type_counts: Conta e enunciar categorias **L3** get_entries: Copia todos os dados BibTeX para um dic **L4** get_valid_group: Guarda título, nome e chave todos os que aparece entre ou em {} **L5** unbrace: Retirar {} dentro de {} ou seja {{cenas}} = cenas **L6** get_author_list: Ordena autores alfabeticamente para ser mais fácil filtrar **L7** invert_name: exemplo: da Cruz, Daniela → Daniela da Cruz **L8** remove_latex_special_chars: Transforma serif, small caps,... **L9** html_create_span: Coloca o ... **L10** html_enclose: Coloca as <cenaz> ... </cenaz> **L11** html_to_small_caps: Coloca o texto assim **L12** html_to_sans_serif: Coloca o texto assim **L13** html_add_attr: colocar um atributo exemplos de cima **L14** get_html_pub_type_index: Mudifica as entradas colocando mais atributos html **L15** get_html_pub_type_index: Mudifica as entradas colocando mais atributos html **L16** str_to_html_math: Processa expressões matemáticas **L17** fix_title: Faz busca dos atributos que tem de ser mudados enviando para a função que as realiza **L18** fix_repeated_authors: Faz a junção de autores que sao a mesma pessoa **L19** format_authors: Escolhe remover acentuações e caracteres especiais que as representam em latex (e.g. " ") do nome dos autores. **L20** remove_consecutive_spaces: Remover os espaços consecutivos **L21** remove_latex_accent: Remove os acentos em latex **L22** last_name_first: Recebe um nome normalizado (e.g. Pedro Filipe H. Pereira) e deve retornar "invertido" (e.g. Pereira, P. F. H.) **L23** is_a_first_last_match: Verifica 1 letra autor 1 com 1 letra autor 2 e o mesmo para a segunda **L24** get_crude_abbrev: Coloca todos os nomes a maiuscula **L25** is_a_first_last_match: Verifica se a letra do primeiro autor e igual a 1 letra do segundo autor e o mesmo para a 2 letra **L26** block_authors_with_two_common_names_v2: Cria blocos com nomes dos autores, quem estiver no mesmo bloco, e a mesma pessoa. **L27** fix_block_func: Cria transitividade nos autores. Sem essa

função, temos blocos A,B e B,C Depois dessa função, vamos ter A,B,C
L28 get_html_pub_type_counts: Coloca em formato *HTML* parâmetros utilizados em R2 **L29** mult_replace: Muda o posicionamento dos nomes através de uma lista em regex

2.3.2 Expressões Regulares

De seguida iremos apresentar as Expressões Regulares utilizadas para normalizar o texto presente no BibTeX e filtrar as informações relevantes. Nestas ER foram utilizadas nos seguintes algoritmos:

- `\b`: **L1**
- `\ \ w+\s*=\s*(?:{((?:[\^{}]+—{(?:[\^{}]+—{[\^{}]*})+})+)}—“([\^”]+)-(\d+))@(\w+){(.+),((?:[\^{}]+—{(?:[\^{}]*—{[\^{}]*})+})+)} : \b` **L3**
- `([\^,]+),\s*([\^,]+) ... \2 \1 : \b` **L7**
- `\(\{—[\^{}]\}\b`
- `<(\w+)([\^>]*)\s*>(.)</\1> <\1\2 attr.upper()="val">\3</\1> : \b` **L13**
- `\textsc{((?:\{—[\^{}]\})+)} : \b` **L17**
- `\band\b : \b` **L19**
- `\s+ : \b` **L20**
- `\ \ W : \b` **L21**
- `\w\w+ : \b` **L26**

2.3.3 Implementação

```
import re
import unicodedata

HTMLPROLOGUE = '<!DOCTYPE\html>\n<HTML\lang="en">\n<HEAD>\n<
meta_charset="utf-8">\n<<<<<<<TITLE>Categories in BibTeX</
TITLE>\n<script_type="text/x-mathjax-config">\nMathJax.Hub.
Config({"extensions":["tex2jax.js"],"jax":["input/TeX","
output/HTML-CSS"],"messageStyle":"none","tex2jax":{"
processEnvironments":false,"processEscapes":true,"inlineMath
":["$","$"],["\\(","\\)"],"displayMath":["$$","$$
"],["\\[","\\]"]},"TeX":{"extensions":["AMSmath.js","
AMSsymbols.js","noErrors.js","noUndefined.js"],"HTML-CSS":{"
availableFonts":["TeX"]}}});\n</script>\n<script_type="text/
javascript">\nasync_src=file:///home/useralef/.vscode/
extensions/shd101wyy.markdown-preview-enhanced-0.6.1/
node_modules/@shd101wyy/mume/dependencies/mathjax/MathJax.js"
_charset="UTF-8"></script>\n</HEAD>\n'
HTMLPILOGUE = '</HTML>'
BIB_EXAMPLE_FILENAME = "exemplo-utf8.bib"
OUTPUT_FILENAME = 'output.html'

#abrir bibtex
```

```
def get_bib_str(filename):
    with open(filename, 'r') as file:
        return re.sub(r'\b}', r'}', file.read())

#contar e enunciar categorias
def get_pub_type_counts(data):
    pub_types_occur = [x[0] for x in data.keys()]
    pub_types = set(pub_types_occur)
    return [(pub_type, pub_types_occur.count(pub_type)) for
            pub_type in pub_types]

#copiar todos os dados bib para um dic {field:field_value}
def get_entries(string):
    d = {}
    field = re.compile(r'(\w+)\s*=\s*
        *((?:((?:[^\{]+\{(?:[^\{]+\{(?:[^\{]*}))+})+})|"(^[^"]
        +)"))')
    for entry in re.finditer(r"@(\w+)\{(.+
        ,((?:[^\{]+\{(?:[^\{]+\{(?:[^\{]*}))+})+)" , string):
        d[entry.group(1).lower(), entry.group(2)] = {
            x[0].lower(): get_valid_group(x, 1, 3)
            for x in field.findall(entry.group(3))}
    return d

#Guardar titulo,nome,chave todos os que aparece entre " " ou em
{}
def get_valid_group(t, begin_or_group, end_or_group):
    for i in range(begin_or_group, end_or_group + 1):
        if v := t[i]:
            return v

# Retirar {} dentro de {} ou seja {{cenas}} = cenas
def unbrace(expression):
    return expression.translate({ord(x):None for x in '{}'})

#Ordenar autores alfabeticamente para ser mais facil filtrar
def get_author_list(data):
    return sorted(set([a for s in data.values() for a in s.get("
        author", [])]))

#Por exemplo: da Cruz, Daniela -> Daniela da Cruz
def invert_name(author_name):
    return re.sub(r"([^\,]+),\s*([^\,]+)", r"\2-\1", author_name)

# Transformar serif, small caps,...
def remove_latex_special_chars(latex_expression):
    return re.sub(r'\\{\{[^\}]\b', r'\1', latex_expression)

#Por o <SPAN> ... </Span>
def html_create_span(expression):
    return html_enclose('span', expression)

#Por as <cenas> ... </cenas>
def html_enclose(tag, string):
```

```

        return rf'<{tag.upper()}>{string}</{tag.upper()}>'

# Colocar o texto assim https://www.w3schools.com/cssref/tryit.
asp?filename=trycss_font-variant
def html_to_small_caps(html_expression):
    return html_add_attr('style', 'font-variant: small-caps',
        html_expression)

#colocar o texto assim https://www.w3schools.com/cssref/tryit.
asp?filename=trycss_font-family
def html_to_sans_serif(html_expression):
    return html_add_attr('style', 'font-family: sans-serif',
        html_expression)

#colocar um atributo exemplos de cima
def html_add_attr(attr, val, html_expression):
    return re.sub(r'<(\w+)([^\>]*)\s*>(.*)</\1>', rf'<\1\2_{attr}.
        upper()}={val}>\3</\1>', html_expression)

# modifica as entradas colocando mais atributos html
def get_html_pub_type_index(data):
    string_ls = [html_enclose('h2', 'Publication_Type_Index')]
    for entry_type in sorted(set(x[0] for x in data)):
        string_ls.append(html_enclose('h3', entry_type))
        for citation_key in [x[1] for x in data if x[0]==
            entry_type]:
            title = data[entry_type, citation_key].get('title', '')
            authors = ', '.join((sorted(data[entry_type,
                citation_key].get('author', ''))))
            string_ls.append(html_enclose('p', f"Key={
                citation_key}<br>Title={fix_title(title)}<br>
                Autores={authors}"))
    return '\n'.join(string_ls)

#Processar expressoes matematicas
def str_to_html_math(string):
    return html_add_attr('class', 'math_inline', html_create_span(
        string))

#faz busca dos atributos que tem de ser mudados enviando para a
funcao que as realiza
def fix_title(title):
    substitutions = [(r'\\textsc{((?:\\{[^\{]+})+)}', lambda m: f'{
        html_to_small_caps(html_create_span(m.group(1)))}'),
        (r'\\textsf{((?:\\{[^\{]+})+)}', lambda m: f'{
            html_to_sans_serif(html_create_span(m.
                group(1)))}'),
        (r'(\$(?:\\.\\\$)+\$)', lambda m: f'{
            str_to_html_math(m.group(1))}')]

    replace = lambda x: mult_replace(x, substitutions)

```

```

        return html_create_span(
            unbrace(
                replace(
                    remove_latex_special_chars(
                        remove_accents(
                            ' '.join(s.strip() for s in title.split('\n'))))))))

#Faz a juncao de autores que sao a mesma pessoa
def fix_repeated_authors(data):
    author_blocks = fix_block_func(
        block_authors_with_two_common_names_v2(get_author_list(
            data)))
    author_dict = {author_name: max(s, key=len) for s in
        author_blocks for author_name in s}
    for d in data.values():
        d['author'] = [author_dict[author] for author in d['
            author']]

# Escolhemos remover acentuacoes e caracteres especiais que as
# representam em latex (e.g. "\\~") do nome dos autores.
def format_authors(data):
    for d in data.values():
        if "author" in d:
            author_lst = [ remove_consecutive_spaces(
                str.strip(
                    invert_name(
                        unbrace(
                            remove_accents(name))))
                for name in re.split(r"\band\b", d["
                    author"].replace("\n", " ")))]
            d['author'] = [author for author in author_lst if
                author]

def remove_consecutive_spaces(name):
    return re.sub(r'\s+', ' ', name)

def remove_latex_accent(name):
    return re.sub(r'\\W', '', name)

def remove_normal_accent(name):
    return ''.join((c for c in unicodedata.normalize('NFD', name)
        ) if unicodedata.category(c) != 'Mn'))

# Recebe um nome normalizado (e.g. Pedro Filipe H. Pereira) e
# deve retornar "invertido" (e.g. Pereira, P. F. H.)
def last_name_first(name):
    initials = ' '.join(get_crude_abbrev(name))[:-2]
    last_name = name.split()[-1]
    return f'{last_name}, {initials}'

#Verificar 1 letra autor 1 com 1 letra autor 2 e o mesmo para a
#segunda
def is_a_first_last_match(author1, author2):

```

```

a1 = get_crude_abbrev(author1)
a2 = get_crude_abbrev(author2)
return a1[0] == a2[0] and a1[-1] == a2[-1]

# colocar todos os nomes a maiuscula
def get_crude_abbrev(name):
    return ''.join(c for c in name if c.isupper())

# Verificar se a letra do primeiro autor e igual a 1 letra do
# segundo autor e o mesmo para a 2 letra
def is_a_first_last_match(author1, author2):
    a1 = get_crude_abbrev(author1)
    a2 = get_crude_abbrev(author2)
    return a1[0] == a2[0] and a1[-1] == a2[-1]

# Criar blocos com nomes dos autores, quem estiver no mesmo
# bloco, e a mesma pessoa.
def block_authors_with_two_common_names_v2(authors):
    res = set()
    for author in authors:
        fs = set()
        for author2 in authors:
            a1 = set(re.findall(r'\w\w+', author))
            a2 = set(re.findall(r'\w\w+', author2))
            if len(a1.intersection(a2)) > 1:
                fs.add(author2)
            elif len(a1) == 1 and len(a1.intersection(a2)) == 1
                and is_a_first_last_match(author, author2):
                fs.add(author2)
        res.add(frozenset(fs))
    return res

#Criar transitividade nos autores. Sem essa funcao, temos
# blocos {A,B} e {B,C} Depois dessa funcao, vamos ter {A,B,C}
def fix_block_func(data):
    res = set()
    for s1 in data:
        q = s1.copy()
        for s2 in data:
            if s1.intersection(s2) != set():
                q = q.union(s2)
        res.add(frozenset(q))
    return res

def get_html_pub_type_counts(data):
    string_ls = [html_encode('h2', 'Number_of_Occurrences_of_
        Publication_Types')]
    pub_counts = sorted(get_pub_type_counts(data), key=lambda x:
        x[1], reverse=True)
    time = lambda v: 's' if v > 1 else ''
    for pub_type, count in pub_counts:
        string_ls.append(html_encode('p', f'Type_{pub_type}_
            appears_{count}_time{time(count)}'))
    return ''.join(string_ls)

```

```

#Muda o posicionamento dos nomes através de uma lista em regex
def mult_replace(string, replacement_list):
    for old, new in replacement_list:
        string = re.sub(old, new, string)
    return string

# Função mais importante que inicializa tudo
def solve(author_name, INPUT_FILENAME=BIB_EXAMPLE_FILENAME):
    html_str_ls = [HTML_PROLOGUE]
    bib_str = get_bib_str(INPUT_FILENAME)

    entries = get_entries(bib_str)
    format_authors(entries)
    fix_repeated_authors(entries)

    html_str_ls.append(html_enclose('body', f'{
        get_html_pub_type_counts(entries)}{
        get_html_common_pub_author(author_name, entries)}{
        get_html_pub_type_index(entries)}{get_html_author_index(
        entries)}'))

    html_str_ls.append(HTML_EPILOGUE)

    with open(OUTPUT_FILENAME, 'w') as file:
        file.write('\n'.join(html_str_ls))

```

2.3.4 Resultado

Publication Type Index

article

```

Key = jj96
Title = NLlex -- a tool to generate lexical analysers for natural language
Autores = Jose Joao Dias de Almeida

Key = Ramalho98
Title = Algebraic specification of documents
Autores = Jose Carlos Ramalho, Jose Joao Dias de Almeida, Pedro Rangel Henriques

Key = RRAH99
Title = SGML documents: Where does quality go?
Autores = Jorge Gustavo Rocha, Jose Carlos Ramalho, Jose Joao Dias de Almeida, Pedro Rangel Henriques

```

Figura 2.2: Índice do Tipo de Publicação

2.4 Tarefa 4 c

2.4.1 Algoritmos

Para esta tarefa, foi necessário acrescentar alguns utilizados que foram as seguintes:

- `get_author_index_dict`: Faz a junção de todos os autores no BibTeX para o dicionário aplicando-lhe um primeiro filtro
- `get_html_author_index`: Faz a transformação de os autores no BibTeX para ficarem no html da alínea 4c)

2.4.2 Implementação

```
#faz a juncao de todos os autores no bib para o dicionario
aplicando lhe um primeiro filtro
def get_author_index_dict(data):
    index = {}
    for key, e in data.items():
        if 'author' in e:
            for author in e['author']:
                author_name = last_name_first(author)
                if author_name not in index:
                    index[author_name] = set()
                index[author_name].add(key[1])
    return index

#Faz a transformacao de os autores no bib para ficarem no html 4
c
def get_html_author_index(data):
    index = sorted(get_author_index_dict(data).items())
    alphabet_order = sorted(set(c[0][0] for c in index))
    string_ls = [html_enclose('h2', 'Author_Index')]
    i = 0
    string_ls.append(html_enclose('h3', alphabet_order[i]))
    for author, citation_keys in index:
        if author[0] != alphabet_order[i]:
            i += 1
            string_ls.append(html_enclose('h3', alphabet_order[i]))
        citation_keys_str = ', '.join(citation_keys)
        string_ls.append(html_enclose('p', f'{author}, {citation_keys_str}'))
    return ''.join(string_ls)
```

Author Index

A

Afonso, S., *linguateca*

Almeida, J. J. D., *linguateca*, RPA99, *xmldt99*, *tesejj*, *speaker:sepln2001*, Reis98, RRAH99, *jj2001x*, *mp2001*, Almeida94c, ABBN98, *dag2002*, *sepln2003*, *museums98*, *graminteractivas1990*, BA97a, *Natura*, Ramalho96, *elpub06-blind*, Barbosa95b, *parguess2002*, *eamt06*, MP07, *elpub06-t2o*, SGML97, AH97, APL2k2.Parguess, Almeida94b, *xata:xmldt*, *cp3a:terminum2003*, Almeida95a, *jj96*, Ramalho95, *xata06:navegante*, *xata07:xmldtmx*, *freq2002*, APL2k2.Synthesis, Almeida96a, *xata04:tx*, *epia-bio-2007*, RSea99, *xata04:mtd*, *epia-music-2007*, *elpub2003*, *avalon:avalinha*, *tlc89*, *xata05:fs*, *harem:rena*, Ramalho98, RAH98, *xata:museudapessoa*, *jj95*, Barbosa2000, *lrec06*, *xata06:xmlauto*, Barbosa95a, Almeida96b, *jspell2002*, Almeida98, RARH98, Barbosa95, *jspell1*, Gis99, ABNO97a, *estruturasdedados90*, *alfarrabio2001*, *sepln06*, ABNO97b, *sepln2004*, *elpub2002*, *avalon:jspell*, Ulisses96, Almeida96c, *xata05:tdt*

Arnold, G., ALHF02

Augustin, I., LGFSSAH08

Azevedo, R. P., LARH09, LPRH07, LPRH07-TM

B

Barbosa, L. S., Barbosa95b, Barbosa95, ABNO97a, ABBN98, ABNO97b, Barbosa2000, BA97a, Barbosa95a

Figura 2.3: Índice de Autores

2.4.3 Resultado

2.5 Tarefa 4 d

Algoritmos

- `get_author_pub_graph`: Recorrendo R5, irá construir um grafo que represente a co-autoria entre os autores.
- `get_dot_graph`: Conforme R6, iremos recorrer a linguagem *Dot* para renderizar o grafo
- `get_html_dot_svg`: Abre o ficheiro transforma a linguagem *Dot* em *HTML*
- `get_html_common_pub_author`: Dado um autor, devolve uma lista dos autores que mais vezes publicaram juntos

Implementação

```
import os
import textwrap
```



```

def get_author_pub_graph(author, data):
    pub_partners = []
    for entry in data.values():
        if 'author' in entry and author in entry['author']:
            for partner in entry['author']:
                if partner != author:
                    pub_partners.append(partner)
    return [(author_name, pub_partners.count(author_name))
            for author_name in set(pub_partners)]

def get_dot_graph(author, data):
    g = sorted(get_author_pub_graph(author, data), key = lambda x:
                x[1])
    string_ls = ['graph{']
    string_ls2 = []
    for partner_author, no_joint_pub in g[-3:]:
        string_ls2.append(f'"{author}" — "{partner_author}" [
            label="{no_joint_pub}"']
    string_ls.append(textwrap.indent('\n'.join(string_ls2), '  '))
    string_ls.append('}')
    return '\n'.join(string_ls)

def get_html_dot_svg(author, data):
    DOT_INPUT_FILENAME = 'dot_input'
    with open(DOT_INPUT_FILENAME, 'w') as file:
        file.write(get_dot_graph(author, data))
    os.system(f'dot -Tsvg -O{DOT_INPUT_FILENAME}')
    with open(DOT_INPUT_FILENAME + '.svg', 'r') as file:
        return re.search(r'<svg(?:\n)+>', file.read()).
            group()

def get_html_common_pub_author(author, data):
    string_ls = [html_encode('h2', 'Author_Graph')]
    string_ls.append(get_html_dot_svg(author, data))
    return ''.join(string_ls)

```

2.5.1 Resultado

Author Graph

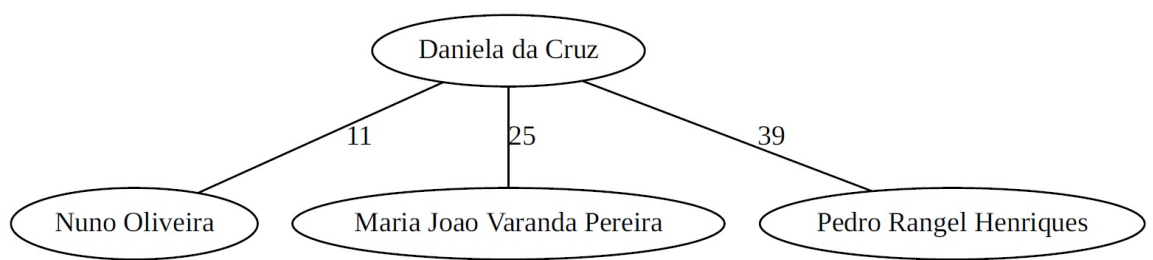


Figura 2.4: Grafo de Autores

Capítulo 3

Conclusão

Através deste projeto foi possível expandir as nossas competências intelectuais sobre o tópico de estudo: Expressões Regulares (ER) que nos possibilitou desenvolver um processador BibTeX utilizando a linguagem *Python*. Consideramos que o produto final cumpre os requisitos mencionados ao longo do mesmo.

Concluindo este projeto, podemos afirmar que foi um trabalho muito desafiante e enriquecedor para cada um de nós, uma vez que tivemos a oportunidade de expandir, aprofundar e aperfeiçoar os nossos conhecimentos.

Apêndice A

Código do Programa

De seguida lista-se o programa completo.

```
import re
import sys
import os
import os.path
import unicodedata

HTMLPROLOGUE = '<!DOCTYPE html>\n<HTML lang="en">\n<HEAD>\n<
meta charset="utf-8">\n<TITLE>Categories in BibTeX</
TITLE>\n<script type="text/x-mathjax-config">MathJax.Hub.
Config({"extensions":["tex2jax.js"],"jax":["input/TeX","
output/HTML-CSS"],"messageStyle":"none","tex2jax":{"
processEnvironments":false,"processEscapes":true,"inlineMath
":["$","$"],"displayMath":["$$","$$"],["\\[","\\]"]},"TeX
":{"extensions":["AMSmath.js","AMSsymbols.js","noErrors.js","
noUndefined.js"]},"HTML-CSS":{"availableFonts":["TeX"]}});</
script><script type="text/javascript" async src="file:///
home/useralef/.vscode/extensions/shd101wyy.markdown-preview-
enhanced-0.6.1/node_modules/@shd101wyy/mume/dependencies/
mathjax/MathJax.js" charset="UTF-8"></script><\/HEAD>\n'
HTMLPEPILOGUE = '<\/HTML>'
BIB_EXAMPLE_FILENAME = "exemplo-utf8.bib"
OUTPUT_FILENAME = 'output.html'

def get_bib_str(filename):
    with open(filename, 'r') as file:
        return re.sub(r'\b}$', r'\n}', file.read())

def get_pub_type_counts(data):
    pub_types_occur = [x[0] for x in data.keys()]
    pub_types = set(pub_types_occur)
    return [(pub_type, pub_types_occur.count(pub_type)) for
            pub_type in pub_types]

def get_entries(string):
    d = {}
```

```

field = re.compile(r'(\w+)\s*=\s*
    *(?:{((?:[^\{]+|{(?:[^\{]+|{[^\{]*})+})+)}|"(?:[^\"]+)"|(\d
    +))')
for entry in re.finditer(r"@(\w+)\{(.+
    ,((?:[^\{]+|{(?:[^\{]+|{[^\{]*})+})+)}" , string):
    d[entry.group(1).lower(), entry.group(2)] = {
        x[0].lower(): get_valid_group(x, 1, 3)
        for x in field.findall(entry.group(3))}
return d

def get_valid_group(t, begin_or_group, end_or_group):
    for i in range(begin_or_group, end_or_group + 1):
        if v := t[i]:
            return v

def unbrace(expression):
    return expression.translate({ord(x):None for x in '{}'})

def get_author_list(data):
    return sorted(set([a for s in data.values() for a in s.get("
        author", [])]))

def invert_name(author_name):
    return re.sub(r"([^\,]+),\s*([^\,]+)", r"\2_\1", author_name)

def remove_latex_special_chars(latex_expression):
    return re.sub(r'\\(\{[^\}]\}\b', r'\1', latex_expression)

def str_to_html_small_caps(expression):
    return html_to_small_caps(html_create_span(expression))

def html_create_span(expression):
    return html_enclose('span', expression)

def html_enclose(tag, string):
    return rf'<{tag.upper()}>{string}</{tag.upper()}>'

def html_to_small_caps(html_expression):
    return html_add_attr('style', 'font-variant:small-caps',
        html_expression)

def html_to_sans_serif(html_expression):
    return html_add_attr('style', 'font-family:sans-serif',
        html_expression)

def html_add_attr(attr, val, html_expression):
    return re.sub(r'<(\w+)([^\>]*)\s*>(.*?)</\1>', rf'<\1\2_{attr}
        upper()>="{val}">\3</\1>', html_expression)

def get_html_pub_type_index(data):
    string_ls = [html_enclose('h2', 'Publication_Type_Index')]
    for entry_type in sorted(set(x[0] for x in data)):
        string_ls.append(html_enclose('h3', entry_type))

```

```

        for citation_key in [x[1] for x in data if x[0]==
            entry_type]:
            title = data[entry_type, citation_key].get('title', '')
            authors = ','.join((sorted(data[entry_type,
                citation_key].get('author', ''))))
            string_ls.append(html_enclose('p', f"Key={
                citation_key}<br>Title={fix_title(title)}<br>
                Autores={authors}")
        return '\n'.join(string_ls)

def str_to_html_math(string):
    return html_add_attr('class', 'math_inline', html_create_span(
        string))

def fix_title(title):
    substitutions = [(r'\\textsc{((?:\\{[^\}]+))}', lambda m: f'{
        html_to_small_caps(html_create_span(m.group(1)))}'),
        (r'\\textsf{((?:\\{[^\}]+))}', lambda m: f'{
        html_to_sans_serif(html_create_span(m.
            group(1)))}'),
        (r'(\$(?:\\.\\\$)+\$)', lambda m: f'{
        str_to_html_math(m.group(1))}')]

    replace = lambda x: mult_replace(x, substitutions)

    return html_create_span(
        unbrace(
            replace(
                remove_latex_special_chars(
                    '\n'.join(s.strip() for s in title.split('\n')))))

def mult_replace(string, replacement_list):
    for old, new in replacement_list:
        string = re.sub(old, new, string)
    return string

def fix_repeated_authors(data):
    author_blocks = fix_block_func(
        block_authors_with_two_common_names_v2(get_author_list(
            data)))
    author_dict = {author_name: max(s, key=len) for s in
        author_blocks for author_name in s}
    for d in data.values():
        d['author'] = [author_dict[author] for author in d['
            author']]

def format_authors(data):
    for d in data.values():
        if "author" in d:
            author_lst = [remove_consecutive_spaces(
                str.strip(

```

```

        invert_name(
            unbrace(
                remove_accents(name))))
        for name in re.split(r"\band\b", d["
            author"].replace("\n", "_"))]
d['author'] = [author for author in author_lst if
    author]

def remove_consecutive_spaces(name):
    return re.sub(r'\s+', '_', name)

def remove_accents(name):
    return remove_latex_accent(remove_normal_accent(name))

def remove_latex_accent(name):
    return re.sub(r'\\W', '', name)

def remove_normal_accent(name):
    return ''.join((c for c in unicodedata.normalize('NFD', name)
        ) if unicodedata.category(c) != 'Mn'))

def last_name_first(name):
    initials = '_'.join(get_crude_abbrev(name))[:-2]
    last_name = name.split()[-1]
    return f'{last_name}_{initials}'

def get_author_index_dict(data):
    index = {}
    for key, e in data.items():
        if 'author' in e:
            for author in e['author']:
                author_name = last_name_first(author)
                if author_name not in index:
                    index[author_name] = set()
                index[author_name].add(key[1])
    return index

def get_author_pub_graph(author, data):
    pub_partners = []
    for entry in data.values():
        if 'author' in entry and author in entry['author']:
            for partner in entry['author']:
                if partner != author:
                    pub_partners.append(partner)
    return [(author_name, pub_partners.count(author_name))
        for author_name in set(pub_partners)]

def get_html_author_index(data):
    index = sorted(get_author_index_dict(data).items())
    alphabet_order = sorted(set(c[0][0] for c in index))
    string_ls = [html_enclose('h2', 'Author_Index')]
    i = 0

```

```

string_ls.append(html_enclose('h3', alphabet_order[i]))
for author, citation_keys in index:
    if author[0] != alphabet_order[i]:
        i += 1
        string_ls.append(html_enclose('h3', alphabet_order[i]
            ))
        citation_keys_str = ', '.join(citation_keys)
        string_ls.append(html_enclose('p', f'{author}, {
            citation_keys_str}'))
return ''.join(string_ls)

def get_crude_abbrev(name):
    return ''.join(c for c in name if c.isupper())

def is_a_first_last_match(author1, author2):
    a1 = get_crude_abbrev(author1)
    a2 = get_crude_abbrev(author2)
    return a1[0] == a2[0] and a1[-1] == a2[-1]

def block_authors_with_two_common_names(authors):
    res = set()
    for author in authors:
        fs = set()
        for author2 in authors:
            if len(set(re.findall(r'\w\w+', author)).intersection(
                re.findall(r'\w\w+', author2))) > 1:
                fs.add(author2)
        if not fs:
            print(author)
        res.add(frozenset(fs))
    return res

def block_authors_with_two_common_names_v2(authors):
    res = set()
    for author in authors:
        fs = set()
        for author2 in authors:
            a1 = set(re.findall(r'\w\w+', author))
            a2 = set(re.findall(r'\w\w+', author2))
            if len(a1.intersection(a2)) > 1:
                fs.add(author2)
            elif len(a1) == 1 and len(a1.intersection(a2)) == 1
                and is_a_first_last_match(author, author2):
                fs.add(author2)
        res.add(frozenset(fs))
    return res

def fix_block_func(data):
    res = set()
    for s1 in data:
        q = s1.copy()
        for s2 in data:

```



```

        if s1.intersection(s2) != set():
            q = q.union(s2)
        res.add(frozenset(q))
    return res

def get_dot_graph(author, data):
    import textwrap
    g = sorted(get_author_pub_graph(author, data), key = lambda x:
                x[1])
    string_ls = ['graph{']
    string_ls2 = []
    for partner_author, no_joint_pub in g[-3:]:
        string_ls2.append(f'"{author}" — "{partner_author}" [
            label="{no_joint_pub}"'])
    string_ls.append(textwrap.indent('\n'.join(string_ls2), '  '))
    string_ls.append('}')
    return '\n'.join(string_ls)

def get_html_pub_type_counts(data):
    string_ls = [html_enclose('h2', 'Number of Occurrences of
        Publication Types')]
    pub_counts = sorted(get_pub_type_counts(data), key=lambda x:
                        x[1], reverse=True)
    time = lambda v: 's' if v > 1 else ''
    for pub_type, count in pub_counts:
        string_ls.append(html_enclose('p', f'Type-{pub_type}-
            appears-{count}-time{time(count)}'))
    return ''.join(string_ls)

def get_html_dot_svg(author, data):
    DOT_INPUT_FILENAME = 'dot_input'
    with open(DOT_INPUT_FILENAME, 'w') as file:
        file.write(get_dot_graph(author, data))
    os.system(f'dot -Tsvg -O{DOT_INPUT_FILENAME}')
    with open(DOT_INPUT_FILENAME + '.svg', 'r') as file:
        return re.search(r'<svg(?:\n)+</svg>', file.read()).
            group()

def get_html_common_pub_author(author, data):
    string_ls = [html_enclose('h2', 'Author Graph')]
    string_ls.append(get_html_dot_svg(author, data))
    return ''.join(string_ls)

def solve(author_name, INPUT_FILENAME=BIB_EXAMPLE_FILENAME):
    html_str_ls = [HTMLPROLOGUE]
    bib_str = get_bib_str(INPUT_FILENAME)

    entries = get_entries(bib_str)
    format_authors(entries)
    fix_repeated_authors(entries)

```

```

html_str_ls.append(html_enclose('body', f'{
    get_html_pub_type_counts(entries)}{
    get_html_common_pub_author(author_name, entries)}{
    get_html_pub_type_index(entries)}{get_html_author_index(
    entries)}'))

html_str_ls.append(HTML_EPILOGUE)

with open(OUTPUT_FILENAME, 'w') as file:
    file.write('\n'.join(html_str_ls))

```

Apêndice B

Código de *Output* do *DOT*

Apresenta-se agora aquilo que é escrito no ficheiro *dot_input*.

```
graph{
  "Daniela da Cruz" — "Nuno Oliveira" [label="11"]
  "Daniela da Cruz" — "Maria Joao Varanda Pereira" [label="25"]
  "Daniela da Cruz" — "Pedro Rangel Henriques" [label="39"]
}
```

Apêndice C

Código do *Output HTML*

Apresenta-se agora aquilo que é escrito no ficheiro *Output.html*.

```
<!DOCTYPE html>
<HTML lang="en">
<HEAD>
<meta charset="utf-8">
  <TITLE>Categories in BibTeX</TITLE>
<script type="text/x-mathjax-config"> MathJax.Hub.Config({
  extensions:["tex2jax.js"],"jax":["input/TeX","output/HTML-
  CSS"],"messageStyle":"none","tex2jax":{"processEnvironments
  ":false,"processEscapes":true,"inlineMath":[["$","$"]],"
  displayMath":[]},"TeX":{"extensions":["AMSMath.js","
  AMSsymbols.js","noErrors.js","noUndefined.js"]},"HTML-CSS
  ":{"availableFonts":["TeX"]}}}); </script> <script type="text
  /javascript" async src="file:///home/useralef/.vscode/
  extensions/shd101wyy.markdown-preview-enhanced-0.6.1/
  node_modules/@shd101wyy/mume/dependencies/mathjax/MathJax.js
  " charset="UTF-8"></script> </HEAD>

<BODY><H2>Number of Occurrences of Publication Types</H2><P>Type
  inproceedings appears 112 times</P><P>Type article appears
  33 times</P><P>Type techreport appears 11 times</P><P>Type
  incollection appears 5 times</P><P>Type phdthesis appears 1
  time</P><P>Type book appears 1 time</P><P>Type mastersthesis
  appears 1 time</P><P>Type misc appears 1 time</P><H2>Author
  Graph</H2><svg width="578pt" height="131pt"
  viewBox="0.00 0.00 577.59 131.00" xmlns="http://www.w3.org
  /2000/svg" xmlns:xlink="http://www.w3.org/1999/xlink">
<g id="graph0" class="graph" transform="scale(1 1) rotate(0)
  translate(4 127)">
<polygon fill="white" stroke="transparent" points="-4,4 -4,-127
  573.59,-127 573.59,4 -4,4"/>
<!-- Daniela da Cruz -->
<g id="node1" class="node">
<title>Daniela da Cruz</title>
<ellipse fill="none" stroke="black" cx="249.74" cy="-105" rx
  ="68.49" ry="18"/>
```

```

<text text-anchor="middle" x="249.74" y="-101.3" font-family="
Times, serif" font-size="14.00">Daniela da Cruz</text>
</g>
<!-- Nuno Oliveira -->
<g id="node2" class="node">
<title>Nuno Oliveira</title>
<ellipse fill="none" stroke="black" cx="61.74" cy="-18" rx
="61.99" ry="18"/>
<text text-anchor="middle" x="61.74" y="-14.3" font-family="
Times, serif" font-size="14.00">Nuno Oliveira</text>
</g>
<!-- Daniela da Cruz&#45;&#45;Nuno Oliveira -->
<g id="edge1" class="edge">
<title>Daniela da Cruz&#45;&#45;Nuno Oliveira</title>
<path fill="none" stroke="black" d="M217,-89.19C182.39,-73.55
128.15,-49.02 93.8,-33.49"/>
<text text-anchor="middle" x="174.74" y="-57.8" font-family="
Times, serif" font-size="14.00">11</text>
</g>
<!-- Maria Joao Varanda Pereira -->
<g id="node3" class="node">
<title>Maria Joao Varanda Pereira</title>
<ellipse fill="none" stroke="black" cx="249.74" cy="-18" rx
="108.58" ry="18"/>
<text text-anchor="middle" x="249.74" y="-14.3" font-family="
Times, serif" font-size="14.00">Maria Joao Varanda Pereira</
text>
</g>
<!-- Daniela da Cruz&#45;&#45;Maria Joao Varanda Pereira -->
<g id="edge2" class="edge">
<title>Daniela da Cruz&#45;&#45;Maria Joao Varanda Pereira</
title>
<path fill="none" stroke="black" d="M249.74,-86.8C249.74,-72.05
249.74,-50.92 249.74,-36.18"/>
<text text-anchor="middle" x="256.74" y="-57.8" font-family="
Times, serif" font-size="14.00">25</text>
</g>
<!-- Pedro Rangel Henriques -->
<g id="node4" class="node">
<title>Pedro Rangel Henriques</title>
<ellipse fill="none" stroke="black" cx="472.74" cy="-18" rx
="96.68" ry="18"/>
<text text-anchor="middle" x="472.74" y="-14.3" font-family="
Times, serif" font-size="14.00">Pedro Rangel Henriques</text>
</g>
<!-- Daniela da Cruz&#45;&#45;Pedro Rangel Henriques -->
<g id="edge3" class="edge">
<title>Daniela da Cruz&#45;&#45;Pedro Rangel Henriques</title>
<path fill="none" stroke="black" d="M286.82,-89.87C326.95,-74.57
390.74,-50.26 432.18,-34.46"/>
<text text-anchor="middle" x="381.74" y="-57.8" font-family="
Times, serif" font-size="14.00">39</text>
</g>
</g>

```

</svg><H2>Publication Type Index</H2>
<H3>article</H3>
<P>Key = jj96
Title = NLlex — a tool to generate
lexical analysers for natural language
Autores =
Jose Joao Dias de Almeida</P>
<P>Key = Ramalho98
Title = Algebraic specification of
documents
Autores = Jose Carlos Ramalho, Jose Joao
Dias de Almeida, Pedro Rangel Henriques</P>
<P>Key = RRAH99
Title = SGML documents: Where does
quality go?
Autores = Jorge Gustavo Rocha, Jose
Carlos Ramalho, Jose Joao Dias de Almeida, Pedro Rangel
Henriques</P>
<P>Key = speaker:sepln2001
Title = Text to speech — a
rewriting system approach
Autores = Alberto Manuel
Brandao Simoes, Jose Joao Dias de Almeida</P>
<P>Key = parguess2002
Title = Grabbing parallel corpora
from the web
Autores = Alberto Manuel Brandao
Simoes, J. Alves de Castro, Jose Joao Dias de Almeida</P>
<P>Key = sepln2003
Title = NATools — A Statistical
Word Aligner Workbench
Autores = Alberto Manuel
Brandao Simoes, Jose Joao Dias de Almeida</P>
<P>Key = xmldt2
Title = — Down-Translating XML

Autores = Alberto Manuel Brandao Simoes</P>
<P>Key = sepln2004
Title = Distributed Translation
Memories implementation using WebServices
Autores =
Alberto Manuel Brandao Simoes, Jose Joao Dias de Almeida,
Xavier Gomez Guinovart</P>
<P>Key = sepln06
Title = A Client-Server Architecture
for building Parallel Corpora applications
Autores
= Alberto Manuel Brandao Simoes, Jose Joao Dias de Almeida</P>
>
<P>Key = KMHVZ04
Title = Grammatical Approach to
Problem Solving
Autores = Maria Joao Varanda
Pereira, Marjan Mernik, Pedro Rangel Henriques, Tomaz Kosar,
Viljem Zumer</P>
<P>Key = HVMLGW05
Title = Automatic Generation of
Language-based Tools using LISA System
Autores =
Hui Wu, Jeff Gray, Maria Joao Varanda Pereira, Marjan Mernik,
Mitja Lenic, Pedro Rangel Henriques</P>
<P>Key = RMHV06
Title = AspectLISA: an aspect-oriented
compiler construction system based on attribute grammars</
SPAN>
Autores = Damijan Rebernak, Maria Joao Varanda
Pereira, Marjan Mernik, Pedro Rangel Henriques</P>
<P>Key = RMHCV06
Title = Specifying Languages using
aspect-oriented approach: AspectLISA
Autores =
Damijan Rebernak, Daniela da Cruz, Maria Joao Varanda Pereira
, Marjan Mernik, Pedro Rangel Henriques</P>
<P>Key = GDH06
Title = AG-based interactive system to
retrieve information from XML documents
Autores =
Alda Lopes Gancarski, Anne Doucet, Pedro Rangel Henriques</P>
<P>Key = BH98
Title = A Framework and Patterns for the
Specification of Reactive Systems
Autores =
Leonor Barroca, Pedro Rangel Henriques</P>

<P>Key = RAH98
Title = Algebraic Specification of Documents
Autores = Jose Carlos Ramalho, Jose Joao Dias de Almeida, Pedro Rangel Henriques</P>

<P>Key = RARH98
Title = SGML Documents: Where does quality go?
Autores = Jorge Gustavo Rocha, Jose Carlos Ramalho, Jose Joao Dias de Almeida, Pedro Rangel Henriques</P>

<P>Key = GRH06
Title = Metamorphosis – A Topic Maps Based Environment to Handle Heterogeneous Information Resources
Autores = Giovani Rubert Librelotto, Jose Carlos Ramalho, Pedro Rangel Henriques</P>

<P>Key = JGRH04
Title = XCSL Tutorial
Autores = Giovani Rubert Librelotto, Jose Carlos Ramalho, Marta Jacinto, Pedro Rangel Henriques</P>

<P>Key = JGRH03
Title = XCSL: XML Constraint Specification Language
Autores = Giovani Rubert Librelotto, Jose Carlos Ramalho, Marta Jacinto, Pedro Rangel Henriques</P>

<P>Key = GRH04
Title = TM-Builder: An Ontology Builder based on XML Topic Maps
Autores = Giovani Rubert Librelotto, Jose Carlos Ramalho, Pedro Rangel Henriques</P>

<P>Key = GRH05a
Title = Gera o automatica de interfaces Web para Sistemas de Informa o: Metamorphosis
Autores = Giovani Rubert Librelotto, Jose Carlos Ramalho, Pedro Rangel Henriques</P>

<P>Key = RH98a
Title = Qualidade na Publica o Electr nica: como control -la?
Autores = Jose Carlos Ramalho, Pedro Rangel Henriques</P>

<P>Key = MSH05
Title = Utilizando uma Base de Dados XML Nativa aplicada ao tratamento de erros num sistema de logs
Autores = Giovana Mendes, Nuno Alberto Silva, Pedro Rangel Henriques</P>

<P>Key = ALHF02
Title = O Uso da Linguagem RS em Rob tica
Autores = Giovani Rubert Librelotto, Gustavo Arnold, Jaime Fonseca, Pedro Rangel Henriques</P>

<P>Key = CHV08ja
Title = Alma versus DDD
Autores = Daniela da Cruz, Maria Joao Varanda Pereira, Pedro Rangel Henriques</P>

<P>Key = FPCH08jb
Title = Language in a Model-Based Engineering Environment for Control Systems — An Approach for Compiler Implementation
Autores = Daniela da Cruz, Elisabete Ferreira, Pedro Rangel Henriques, Rogerio Paulo</P>

<P>Key = PMCH08j
Title = Program Comprehension for Domain-Specific Languages (invited paper)
Autores = Daniela da Cruz, Maria Joao Varanda Pereira, Marjan Mernik, Pedro Rangel Henriques</P>

<P>Key = CHV07
Title = Constructing program animations using a pattern-based approach
Autores = Daniela da Cruz, Maria Joao Varanda Pereira, Pedro Rangel Henriques</P>

<P>Key = LARH09
Title = Topic Maps Constraint Languages : understanding and comparing
Autores = Giovani Rubert Librelotto, Jose Carlos Ramalho, Pedro Rangel Henriques, Renato Preigschadt de Azevedo</P>

<P>Key = CBHP09
Title = Code Inspection Approaches for Program Visualization
Autores = Daniela da Cruz, Maria Joao Varanda Pereira, Mario Beron, Pedro Rangel Henriques</P>

<P>Key = OPHCC2010
Title = VisualLISA: A Visual Environment to Develop Attribute Grammars
Autores = Bastian Cramer, Daniela da Cruz, Maria Joao Varanda Pereira, Nuno Oliveira, Pedro Rangel Henriques</P>

<P>Key = KOMPCCH2010
Title = Comparing General-Purpose and Domain-Specific Languages: An Empirical Study
Autores = Daniela da Cruz, Maria Joao Varanda Pereira, Marjan Mernik, Matej Crepinsek, Nuno Oliveira, Pedro Rangel Henriques, Tomaz Kosar</P>

<H3>book</H3>

<P>Key = RH02
Title = XML \& XSL: da teoria prtica
Autores = Jose Carlos Ramalho, Pedro Rangel Henriques</P>

<H3>incollection</H3>

<P>Key = avalon:jspell
Title = nas Morfolimpadas
Autores = Alberto Manuel Brandao Simoes, Jose Joao Dias de Almeida</P>

<P>Key = avalon:avalinha
Title = Avaliao de alinhadores
Autores = Alberto Manuel Brandao Simoes, Jose Joao Dias de Almeida</P>

<P>Key = harem:rena
Title = Reconhecedor de Entidades
Autores = Jose Joao Dias de Almeida</P>

<P>Key = RRH02
Title = Data Reduction to Improve Knowledge Extraction
Autores = Carlos Ramos, Maria de Fatima Rodrigues, Pedro Rangel Henriques</P>

<P>Key = ORH06
Title = Data Cleaning by Reusing Domain Knowledge
Autores = Maria de Fatima Rodrigues, Paulo Oliveira, Pedro Rangel Henriques</P>

<H3>inproceedings</H3>

<P>Key = graminteractivas1990
Title = Mecanismos para Especificao e Prototipagem de Interfaces Utilizador-Sistema
Autores = F. Mario Martins, Jose Joao Dias de Almeida, Pedro Rangel Henriques</P>

<P>Key = Almeida94b
Title = GPC — a Tool for higher-order grammar specification
Autores = Jose Joao Dias de Almeida</P>

<P>Key = Almeida95a
Title = YaLG — extending DCG for natural language processing
Autores = Jose Joao Dias de Almeida</P>

<P>Key = Almeida94c
Title = Jspell — um m dulo para anlise l xica gen rica de linguagem natural
Autores = Jose Joao Dias de Almeida, Ulisses Pinto</P>

<P>Key = Ramalho95
Title = Algebraic Specification of Documents
Autores = Jose Carlos Ramalho, Jose Joao Dias de Almeida, Pedro Rangel Henriques</P>

<P>Key = Almeida96a
Title = Especificao e tratamento de Dicionrios
Autores = Jose Joao Dias de Almeida</P>

<P>Key = Ulisses96
Title = Tratamento automatico de termos compostos
Autores = Jose Joao Dias de

Almeida, Ulisses Pinto</P>

<P>Key = Almeida96b
Title = YaLG a tool for higher-order grammar specification
Autores = J.B. Barros, Jose Joao Dias de Almeida</P>

<P>Key = Ramalho96
Title = Document Semantics: two approaches
Autores = Jose Carlos Ramalho, Jose Joao Dias de Almeida, Pedro Rangel Henriques</P>

<P>Key = SGML97
Title = SGML Documents: where does quality go?
Autores = Jorge Gustavo Rocha, Jose Carlos Ramalho, Jose Joao Dias de Almeida, Pedro Rangel Henriques</P>

<P>Key = Almeida98
Title = Programa o de dicion rios
Autores = Jose Joao Dias de Almeida</P>

<P>Key = Reis98
Title = Etiquetador morfo-sint ctico para o Portugu s
Autores = Jose Joao Dias de Almeida, Ricardo Reis</P>

<P>Key = ABNO97a
Title = Camila: Formal Software Engineering Supported by Functional Programming
Autores = F.L. Neves, J.N. Oliveira, Jose Joao Dias de Almeida, L.S. Barbosa</P>

<P>Key = ABNO97b
Title = Camila: Prototyping and Refinement of Constructive Specifications
Autores = F.L. Neves, J.N. Oliveira, Jose Joao Dias de Almeida, L.S. Barbosa</P>

<P>Key = AH97
Title = Dynamic Dictionary = cooperative information sources
Autores = Jose Joao Dias de Almeida, Pedro Rangel Henriques</P>

<P>Key = museums98
Title = Adapting Museum Structures for the Web: No Changes Needed!
Autores = J.L. Faria, Jorge Gustavo Rocha, Jose Carlos Ramalho, Jose Joao Dias de Almeida, Mario Ricardo Henriques, Pedro Rangel Henriques</P>

<P>Key = ABBN98
Title = On The Development of Camila
Autores = J.B. Barros, Jose Joao Dias de Almeida, L.F. Neves, L.S. Barbosa</P>

<P>Key = Gis99
Title = Systems Development
Autores = Ana Silva, Jorge Gustavo Rocha, Jose Joao Dias de Almeida, Mario Ricardo Henriques, Pedro Rangel Henriques</P>

<P>Key = RPA99
Title = Maps
Autores = Jorge Gustavo Rocha, Jose Joao Dias de Almeida, Tiago Pedroso</P>

<P>Key = RSea99
Title = SIG
Autores = Ana Silva, Jorge Gustavo Rocha, Jose Joao Dias de Almeida, Mario Ricardo Henriques, Pedro Rangel Henriques</P>

<P>Key = xmldt99
Title = a Perl Down-Translation module
Autores = Jose Carlos Ramalho, Jose Joao Dias de Almeida</P>

<P>Key = Barbosa2000
Title = Polytypic Recursion Patterns
Autores = J.B. Barros, Jose Joao Dias de Almeida, L.S. Barbosa</P>

<P>Key = jj2001x
Title = Smallbook — comando para produ o de livros em pequena escala
Autores =

Jose Joao Dias de Almeida</P>

<P>Key = mp2001
Title = — Arquitectura
Autores = Alberto Manuel Brandao Simoes, Jorge Gustavo Rocha, Jose Joao Dias de Almeida, Pedro Rangel Henriques, Sonia Moreira</P>

<P>Key = alfarrabio2001
Title = Alfarr bio: Adding value to an Heterogeneous Site Collection
Autores = Alberto Manuel Brandao Simoes, Jorge Gustavo Rocha, Jose Joao Dias de Almeida, Pedro Rangel Henriques</P>

<P>Key = freq2002
Title = C lculo de frequ ncias de palavras para entradas de dicion rios atrav s do uso conjunto de analisadores morfol gicos, taggers e corpora
Autores = Alberto Manuel Brandao Simoes, Jose Joao Dias de Almeida, Paulo A. Rocha</P>

<P>Key = jspell2002
Title = Jspell.pm — um m dulo de an lise morfol gica para uso em processamento de linguagem natural
Autores = Alberto Manuel Brandao Simoes, Jose Joao Dias de Almeida</P>

<P>Key = dag2002
Title = Directory Attribute Grammars
Autores = Alberto Manuel Brandao Simoes, Jose Joao Dias de Almeida, Pedro Rangel Henriques</P>

<P>Key = elpub2002
Title = Library::* — a toolkit for digital libraries
Autores = Alberto Manuel Brandao Simoes, Jose Joao Dias de Almeida</P>

<P>Key = APL2k2.Parguess
Title = Extrac o de corpora paralelo a partir da web: constru o e disponibiliza o
Autores = Alberto Manuel Brandao Simoes, J. Alves de Castro, Jose Joao Dias de Almeida</P>

<P>Key = APL2k2.Synthesis
Title = Gera o de voz com sotaque
Autores = Alberto Manuel Brandao Simoes, Jose Joao Dias de Almeida</P>

<P>Key = xata:xmldt
Title = Engenharia reversa de HTML usando tecnologia XML
Autores = Alberto Manuel Brandao Simoes, Jose Joao Dias de Almeida</P>

<P>Key = xata:museudapessoa
Title = essoa
Autores = Alberto Manuel Brandao Simoes, Jose Joao Dias de Almeida</P>

<P>Key = elpub2003
Title = Music publishing
Autores = Alberto Manuel Brandao Simoes, Jose Joao Dias de Almeida</P>

<P>Key = cp3a:terminum2003
Title = Projecto TerminUM
Autores = Alberto Manuel Brandao Simoes, Bruno Martins, J. Alves de Castro, Jose Joao Dias de Almeida, Paulo Silva</P>

<P>Key = cp3a:kvec2003
Title = Lingua-Biterm: um m dulo Perl para extrac o de terminologia bilingue
Autores = Bruno Martins</P>

<P>Key = cp3a:natools2003
Title = Alinhamento de corpora paralelos
Autores = Alberto Manuel Brandao Simoes</P>

<P>Key = xata04:tx
Title = baseada em tipos din micos
Autores = Alberto Manuel Brandao Simoes, Jose Joao Dias de Almeida</P>

<P>Key = xata04:mtd
Title = Memórias de Tradução
Distribuídas
Autores = Alberto Manuel Brandão
Simões, José João Dias de Almeida, Xavier Gomez Guinovart</P>

<P>Key = linguatca
Title = Linguatca: um centro de
recursos distribuído para o processamento computacional da
língua portuguesa
Autores = Alberto Manuel Brandão
Simões, Ana Frankenberg-Garcia, Ana Pinto, Anabela Barreiro,
Belinda Maia, Cristina Mota, Debora Oliveira, Diana Santos,
Eckhard Bick, Elisabete Ranchhod, José João Dias de Almeida,
Luís Cabral, Luís Costa, Luís Sarmento, Marcirio Chaves, Nuno
Cardoso, Paulo A. Rocha, Rachel Aires, Rosario Silva, Rui
Vilela, Susana Afonso</P>

<P>Key = xata05:fs
Title = Representação em XML da
Floresta Sintática
Autores = Alberto Manuel
Brandão Simões, Eckhard Bick, José João Dias de Almeida, Rui
Vilela</P>

<P>Key = xata05:tdt
Title = Inferência de tipos em
documentos XML
Autores = Alberto Manuel Brandão
Simões, José João Dias de Almeida</P>

<P>Key = xata06:navegante
Title = Navegante: um proxy
de ordem superior para navegação intuitiva
Autores
= Alberto Manuel Brandão Simões, José João Dias de Almeida</P>

<P>Key = xata06:xmlauto
Title = XML
Autores =
Alberto Manuel Brandão Simões, José João Dias de Almeida</P>

<P>Key = eamt06
Title = Combinatory Examples Extraction
for Machine Translation
Autores = Alberto Manuel
Brandão Simões, José João Dias de Almeida</P>

<P>Key = lrec06
Title = — Recycling Thesauri into a
Multilingual Ontology
Autores = Alberto Manuel
Brandão Simões, José João Dias de Almeida</P>

<P>Key = elpub06-t2o
Title = Publishing multilingual
ontologies: a quick way of obtaining feedback

Autores = Alberto Manuel Brandão Simões, José João Dias de
Almeida</P>

<P>Key = elpub06-blind
Title = Transcoding for Web
Accessibility for the Blind: Semantics from Structure

Autores = Alberto Manuel Brandão Simões, Alexandre
Carvalho, Antonio R. Fernandes, José João Dias de Almeida</P>

<P>Key = xata07:xlmtmx
Title = — Processamento de
Memórias de Tradução de Grandes Dimensões

Autores = Alberto Manuel Brandão Simões, José João Dias de
Almeida</P>

<P>Key = MP07
Title = Dependency Specification Language

Autores = Alberto Manuel Brandão Simões, José João
Dias de Almeida, Ruben Fonseca</P>

<P>Key = epia-bio-2007
Title = An Ontology-Based
Approach To Systems Biology Literature Retrieval and
Processing
Autores = Alberto Manuel Brandão Simões,
Anália Lourenço, Eugénio Ferreira, Isabel Rocha, José João
Dias de Almeida, Miguel Rocha</P>

<P>Key = epia-music-2007
Title = Using Text Mining
Techniques for Classical Music Scores Analysis

Autores = Alberto Manuel Brandão Simões, Anália Lourenço,

Jose Joao Dias de Almeida</P>

<P>Key = HKMVZ03
Title = Grammatical Approach to
Problem Solving
Autores = Maria Joao Varanda
Pereira, Marjan Mernik, Pedro Rangel Henriques, Tomaz Kosar,
Viljem Zumer</P>

<P>Key = VH01
Title = Visualization / Animation of
Programs based on Abstract Representations and Formal
Mappings
Autores = Maria Joao Varanda Pereira,
Pedro Rangel Henriques</P>

<P>Key = VH02
Title = Automatic Generation of Language-
based Tools
Autores = Maria Joao Varanda Pereira,
Marjan Mernik, Mitja Lenic, Pedro Rangel Henriques</P>

<P>Key = VH03
Title = Visualization / Animation of
Programs in Alma: obtaining different results

Autores = Maria Joao Varanda Pereira, Pedro Rangel Henriques
</P>

<P>Key = RMHVC06
Title = Specifying Languages using
Aspect-oriented Approach: AspectLISA
Autores =
Damijan Rebernak, Daniela da Cruz, Maria Joao Varanda Pereira
, Marjan Mernik, Pedro Rangel Henriques</P>

<P>Key = BHVU07d
Title = PICS una Herramienta para la
Comprensión e Inspección de Programas
Autores =
Maria Joao Varanda Pereira, Mario Beron, Pedro Rangel
Henriques, Roberto Uzal</P>

<P>Key = BHVU07c
Title = Program Inspection to
Interconnect Behavioral and Operational View for Program
Comprehension
Autores = Maria Joao Varanda Pereira,
Mario Beron, Pedro Rangel Henriques, Roberto Uzal</P>

<P>Key = BHVU07b
Title = Comprensión de Programas por
Inspección Visual y Animación
Autores = Maria
Joao Varanda Pereira, Mario Beron, Pedro Rangel Henriques,
Roberto Uzal</P>

<P>Key = BHVU07a
Title = Static and Dynamic Strategies
to Understand C Programs by Code Annotation
Autores
= Maria Joao Varanda Pereira, Mario Beron, Pedro Rangel
Henriques, Roberto Uzal</P>

<P>Key = CHLB07a
Title = O Sitio de Pico, Software
Educativo para Crianças con Paralisia Cerebral

Autores = Elisabete Cunha, Mario Beron, Pedro Rangel
Henriques, Sandra Cristina Lopes</P>

<P>Key = BHVU06a
Title = Herramientas para la
comprensión de programas
Autores = Maria Joao
Varanda Pereira, Mario Beron, Pedro Rangel Henriques, Roberto
Uzal</P>

<P>Key = BHVU06b
Title = Comprensión de Algoritmos de
Ruteo
Autores = Maria Joao Varanda Pereira, Mario
Beron, Pedro Rangel Henriques, Roberto Uzal</P>

<P>Key = BHVUM06
Title = A Language Processing Tool for
Program Comprehension
Autores = G. Montejano,
Maria Joao Varanda Pereira, Mario Beron, Pedro Rangel
Henriques, Roberto Uzal</P>

<P>Key = BHVU08
Title = Simplificando la Comprensión
de Programas a través de la Interconexión de Dominios

Autores = Maria Joao Varanda Pereira, Mario Beron,

Pedro Rangel Henriques, Roberto Uzal</P>

<P>Key = BHV06
Title = A System for Evaluate and Understand Routing Algorithms
Autores = Maria Joao Varanda Pereira, Mario Beron, Pedro Rangel Henriques</P>

<P>Key = BCVHU08
Title = Evaluation Criteria of Software Visualization Systems used for Program Comprehension
Autores = Daniela da Cruz, Maria Joao Varanda Pereira, Mario Beron, Pedro Rangel Henriques, Roberto Uzal</P>

<P>Key = BUHV08
Title = Inspecci n de C digo para relacionar los Dominios del Problema y Programa para la Comprensi n de Programas
Autores = Maria Joao Varanda Pereira, Mario Beron, Pedro Rangel Henriques, Roberto Uzal</P>

<P>Key = OVH05
Title = Compreens o de Aplica es Web : O Processo e as Ferramentas
Autores = Eva Oliveira, Maria Joao Varanda Pereira, Pedro Rangel Henriques</P>

<P>Key = OHV06
Title = Proposta de um Sistema para Compreens o de Aplica es Web
Autores = Eva Oliveira, Maria Joao Varanda Pereira, Pedro Rangel Henriques</P>

<P>Key = GH07b
Title = Analyzing the structure of scientific articles to improve information retrieval
Autores = Alda Lopes Gancarski, Pedro Rangel Henriques</P>

<P>Key = GH07a
Title = Using data together with metadata to improve XML information access
Autores = Alda Lopes Gancarski, Pedro Rangel Henriques</P>

<P>Key = GFH08
Title = Using data together with metadata to improve XML information access
Autores = Alda Lopes Gancarski, Flavio Xavier Ferreira, Pedro Rangel Henriques</P>

<P>Key = FGH08
Title = Information access from XML using semantics and context: application to the Portuguese Emigration Museum
Autores = Alda Lopes Gancarski, Flavio Xavier Ferreira, Pedro Rangel Henriques</P>

<P>Key = FH08
Title = Using OWL to specify and build different views over the Emigration Museum resources
Autores = Flavio Xavier Ferreira, Pedro Rangel Henriques</P>

<P>Key = LPRH07
Title = Navegando na Rede Sem ntica dos Topic Maps com o Ulisses
Autores = Giovanni Rubert Librelotto, Jose Carlos Ramalho, Pedro Rangel Henriques, Renato Preigschadt de Azevedo</P>

<P>Key = LPRH07-TM
Title = Topic Maps Constraint Specification Languages: comparing AsTMa!, OSL, and XTche
Autores = Giovanni Rubert Librelotto, Jose Carlos Ramalho, Pedro Rangel Henriques, Renato Preigschadt de Azevedo</P>

<P>Key = LRHGT08
Title = A Framework to specify, extract and manage Topic Maps driven by ontologie
Autores = Giovanni Rubert Librelotto, Jonas Bulegon Gassen, Jose Carlos Ramalho, Pedro Rangel Henriques, Rogerio Correa Turchetti</P>

<P>Key = LGFSSAH08
Title = Uma Ontologia aplicada a um
 Ambiente Pervasivo Hospitalar
Autores = Fabio L.
 Silva, Giovani Rubert Librelotto, Iara Augustin, Jonas
 Bulegon Gassen, Leandro O. Freitas, Matheus C. Silveira,
 Pedro Rangel Henriques</P>

<P>Key = LMMVRH08
Title = Generating a Semantic Network
 for PubMed
Autores = Giovani Rubert Librelotto,
 Henrique Machado, Jose Carlos Ramalho, Juliana Vizzotto,
 Mirkos Martins, Pedro Rangel Henriques</P>

<P>Key = CPH07f
Title = Pattern-based Program
 Visualization
Autores = Daniela da Cruz, Maria Joao
 Varanda Pereira, Pedro Rangel Henriques</P>

<P>Key = CH07g
Title = Slicing wxHaskell modules to
 derive the User Interface Abstract Model (short paper and
 poster)
Autores = Daniela da Cruz, Pedro Rangel
 Henriques</P>

<P>Key = CH07h
Title = Laboratory Site (poster)

 Autores = Daniela da Cruz, Pedro Rangel Henriques</P>

<P>Key = FCHV08
Title = How to interconnect operational
 and behavioral views of web applications
Autores =
 Daniela da Cruz, Maria Joao Varanda Pereira, Pedro Rangel
 Henriques, Ruben Fonseca</P>

<P>Key = CHP08i
Title = Strategies for Program
 Inspection and Visualization
Autores = Daniela da
 Cruz, Maria Joao Varanda Pereira, Pedro Rangel Henriques</P>

<P>Key = CH07a
Title = anguage
Autores =
 Daniela da Cruz, Pedro Rangel Henriques</P>

<P>Key = CLH07c
Title = Como ensinar com Mapas de
 Conceitos: duas abordagens complementares
Autores =
 Daniela da Cruz, Pedro Rangel Henriques, Sandra Cristina
 Lopes</P>

<P>Key = CH07d
Title = LISS — The language and the
 compiler
Autores = Daniela da Cruz, Pedro Rangel
 Henriques</P>

<P>Key = CFPBH07d
Title = Comparing Generators for
 Language-based Tools
Autores = Daniela da Cruz,
 Maria Joao Varanda Pereira, Mario Beron, Pedro Rangel
 Henriques, Ruben Fonseca</P>

<P>Key = CHP08a
Title = Documents
Autores =
 Daniela da Cruz, Maria Joao Varanda Pereira, Pedro Rangel
 Henriques</P>

<P>Key = CHP08b
Title = DDD
Autores = Daniela
 da Cruz, Maria Joao Varanda Pereira, Pedro Rangel Henriques
 </P>

<P>Key = CPH08c
Title = Properties Preservation during
 Transformation (short paper)
Autores = Daniela da
 Cruz, Jorge Sousa Pinto, Pedro Rangel Henriques</P>

<P>Key = FPCH08d
Title = Language in a Model-Based
 Engineering Environment for Control Systems — An Approach
 for Compiler Implementation
Autores = Daniela da
 Cruz, Elisabete Ferreira, Pedro Rangel Henriques, Rogerio
 Paulo</P>

<P>Key = PMCH08e
Title = : a Visual Interface for an
 Attribute Grammar based Compiler-Compiler (short paper)

Autores = Daniela da Cruz, Maria Joao Varanda Pereira, Marjan Mernik, Pedro Rangel Henriques</P>

<P>Key = PMCH08f
Title = Program Comprehension for Domain-Specific Languages
Autores = Daniela da Cruz, Maria Joao Varanda Pereira, Marjan Mernik, Pedro Rangel Henriques</P>

<P>Key = CHP09a
Title = Code Analysis: Past and Present
Autores = Daniela da Cruz, Jorge Sousa Pinto, Pedro Rangel Henriques</P>

<P>Key = OPCH09a
Title = Visualization of Domain-Specific Programs' Behavior
Autores = Daniela da Cruz, Maria Joao Varanda Pereira, Nuno Oliveira, Pedro Rangel Henriques</P>

<P>Key = oliveira09b
Title = Applying Program Comprehension Techniques to Karel Robot Programs
Autores = Daniela da Cruz, Maria Joao Varanda Pereira, Marjan Mernik, Matej Crepinsek, Nuno Oliveira, Pedro Rangel Henriques, Tomaz Kosar</P>

<P>Key = oliveira09c
Title = VisualLISA: Visual Programming Environment for Attribute Grammars Specification
Autores = Daniela da Cruz, Maria Joao Varanda Pereira, Nuno Oliveira, Pedro Rangel Henriques</P>

<P>Key = kosar09
Title = Influence of domain-specific notation to program understanding
Autores = Daniela da Cruz, Maria Joao Varanda Pereira, Marjan Mernik, Nuno Oliveira, Pedro Rangel Henriques, Tomaz Kosar</P>

<P>Key = FCHGD09a
Title = A Query-by-Example Approach for XML Querying
Autores = Alda Lopes Gancarski, Bruno Defude, Daniela da Cruz, Flavio Xavier Ferreira, Pedro Rangel Henriques</P>

<P>Key = ORH09a
Title = SMARTCLEAN: uma ferramenta para a limpeza incremental de dados
Autores = Maria de Fatima Rodrigues, Paulo Jorge Oliveria, Pedro Rangel Henriques</P>

<P>Key = LPH09a
Title = Uma metodologia para Consultas aos Bancos de Dados do NCBI
Autores = Giovanni Rubert Librelotto, Pedro Rangel Henriques, Rafael Teodosio Pereira</P>

<P>Key = OPHC09a
Title = Domain Specific Languages: A Theoretical Survey
Autores = Daniela da Cruz, Maria Joao Varanda Pereira, Nuno Oliveira, Pedro Rangel Henriques</P>

<P>Key = OPHCC09
Title = : A Domain Specific Visual Language for Attribute Grammars
Autores = Bastian Cramer, Daniela da Cruz, Maria Joao Varanda Pereira, Nuno Oliveira, Pedro Rangel Henriques</P>

<P>Key = MKCHCPO09
Title = Comparison of XAML and C\# Forms using Cognitive Dimensions Framework
Autores = Daniela da Cruz, Maria Joao Varanda Pereira, Marjan Mernik, Nuno Oliveira, Pedro Rangel Henriques, Tomaz Kosar</P>

<P>Key = OHCP09
Title = XAGra - An XML Dialect for Attribute Grammars
Autores = Daniela da Cruz, Maria Joao Varanda Pereira, Nuno Oliveira, Pedro Rangel Henriques</P>

<P>Key = FCHGD09b
Title = GuessXQ, an inference Web-engine for querying XML Documents
Autores = Alda Lopes Gancarski, Bruno Defude, Daniela da Cruz, Flavio Xavier Ferreira, Pedro Rangel Henriques</P>

<P>Key = BHVU09
Title = Instrumentaciones de Programas Escritos en C para Interrelacionar las Vistas Comportamental y Operacional de los Sistemas de Software
Autores = Maria Joao Varanda Pereira, Mario Beron, Pedro Rangel Henriques, Roberto Uzal</P>

<P>Key = CH09d
Title = Assessing Databases in .Net: comparing approaches
Autores = Daniela da Cruz, Pedro Rangel Henriques</P>

<P>Key = CH2010a
Title = Exploring, Visualizing and Slicing the Soul of XML Documents
Autores = Daniela da Cruz, Pedro Rangel Henriques</P>

<H3>mastersthesis</H3>

<P>Key = teseambs
Title = Parallel Corpora word alignment and applications
Autores = Alberto Manuel Brandao Simoes</P>

<H3>misc</H3>

<P>Key = cruz09
Title = GraAL – A Grammar Analyzer
Autores = Daniela da Cruz, Nuno Oliveira, Pedro Rangel Henriques</P>

<H3>phdthesis</H3>

<P>Key = tesejj
Title = Diccion rios din micos multi-fonte
Autores = Jose Joao Dias de Almeida</P>

<H3>techreport</H3>

<P>Key = tlc89
Title = Teoria das Linguagens
Autores = J.B. Barros, Jose Joao Dias de Almeida</P>

<P>Key = estruturasdedados90
Title = Estruturas de Dados
Autores = J.B. Barros, Jose Joao Dias de Almeida</P>

<P>Key = Camila
Title = Camila – A Platform for Software Mathematical Development
Autores = projecto Camila</P>

<P>Key = Natura
Title = Natura – Natural language processing
Autores = Jose Joao Dias de Almeida</P>

<P>Key = jspell1
Title = Manual de Utilizador do JSpell
Autores = Jose Joao Dias de Almeida, Ulisses Pinto</P>

<P>Key = jj95
Title = NLlex — a tool to generate lexical analysers for natural language
Autores = Jose Joao Dias de Almeida</P>

<P>Key = Barbosa95
Title = System Prototyping in Camila
Autores = Jose Joao Dias de Almeida, L.S. Barbosa</P>

<P>Key = Barbosa95a
Title = Camila: A reference Manual
Autores = Jose Joao Dias de Almeida, L.S. Barbosa</P>

<P>Key = BA97a
Title = Systems Prototyping in Camila
Autores = Jose Joao Dias de Almeida, L.S. Barbosa</P>

<P>Key = Barbosa95b
Title = Growing Up With Camila
Autores =

Jose Joao Dias de Almeida, L.S. Barbosa

Key = Almeida96c
Title = From BiTeX to HTML semantic nets
Autores = Jose Carlos Ramalho, Jose Joao Dias de Almeida
Author Index
Afonso, S., linguatca, Aires, R., linguatca
Almeida, J. J. D., linguatca, RPA99, xmldt99, tesejj, speaker:sepln2001, Reis98, RRAH99, jj2001x, mp2001, Almeida94c, ABBN98, dag2002, sepln2003, museums98, graminteractivas1990, BA97a, Natura, Ramalho96, elpub06-blind, Barbosa95b, parguess2002, eamt06, MP07, elpub06-t2o, SGML97, AH97, APL2k2.Parguess, Almeida94b, xata:xmldt, cp3a:terminum2003, Almeida95a, jj96, Ramalho95, xata06:navegante, xata07:xmltmx, freq2002, APL2k2.Synthesis, Almeida96a, xata04:tx, epia-bio-2007, RSea99, xata04:mtd, epia-music-2007, elpub2003, avalon:avalinha, tlc89, xata05:fs, harem:rena, Ramalho98, RAH98, xata:museudapessoa, jj95, Barbosa2000, lrec06, xata06:xmlauto, Barbosa95a, Almeida96b, jspell2002, Almeida98, RARH98, Barbosa95, jspell1, Gis99, ABNO97a, estruturasdedados90, alfarrabio2001, sepln06, ABNO97b, sepln2004, elpub2002, avalon:jspell, Ulisses96, Almeida96c, xata05:tdt
Arnold, G., ALHF02
Augustin, I., LGFSSAH08
Azevedo, R. P., LARH09, LPRH07, LPRH07-TM
Barbosa, L. S., Barbosa95b, Barbosa95, ABNO97a, ABBN98, ABNO97b, Barbosa2000, BA97a, Barbosa95a
Barreiro, A., linguatca
Barroca, L., BH98
Barros, J. B., Almeida96b, estruturasdedados90, ABBN98, tlc89, Barbosa2000
Beron, M., BHVU08, BHVU07d, BCVHU08, BHVU07a, BHVU06, BHVUM06, BHVU09, BHVU06b, BHVU07c, CBHP09, BHVU07b, BUHV08, BHVU06a, CFPBH07d, CHLB07a
Bick, E., linguatca, xata05:fs
Cabral, L., linguatca
Camila, Camila
Cardoso, N., linguatca
Carvalho, A., elpub06-blind
Castro, J. A., cp3a:terminum2003, parguess2002, APL2k2.Parguess
Chaves, M., linguatca
Costa, L., linguatca
Cramer, B., OPHCC09, OPHCC2010
Crepinsek, M., KOMPCCH2010, oliveira09b
Cruz, D., FPCH08d, oliveira09b, KOMPCCH2010, RMHCV06, oliveira09c, CH07d, CHV07, PMCH08j, CLH07c, BCVHU08, FCHGD09a, FCHGD09b, cruz09, CH07h, CHV08ja, CHP09a, CFPBH07d, PMCH08e, CPH08c, CH07g, kosar09, CHP08a, OPHCC2010, CPH07f, RMHVC06, CBHP09, OPCH09a, FPCH08jb, CH2010a, CHP08b, CHP08i, OPHC09a, PMCH08f, MKCHCPO09, OHCP09, OPHCC09, CH09d, CH07a, FCHV08
Cunha, E., CHLB07a
D</H3>
Defude, B., FCHGD09a, FCHGD09b
Doucet, A., GDH06
F</H3>
Faria, J. L., museums98
Fernandes, A. R., elpub06-blind
Ferreira, E., epia-bio-2007, FPCH08d, FPCH08jb
Ferreira, F. X., FH08, FGH08, FCHGD09b, FCHGD09a, GFH08
Fonseca, J., ALHF02
Fonseca, R., MP07, FCHV08, CFPBH07d
Frankenberg-Garcia, A. F., linguatca
Freitas, L. O., LGFSSAH08
G</H3>
Gancarski, A. L., GH07b, GDH06, GH07a, FGH08, FCHGD09a, FCHGD09b, GFH08
Gassen, J. B., LGFSSAH08, LRHGT08
G</H3>
Gray, J., HVMLGW05
Guinovart, X. G., sepln2004, xata04:mtd
H</H3>
Henriques, M. R., RSea99, Gis99, museums98
Henriques, P. R., FPCH08d, ORH09a, HKMVZ03,

JGRH03, cruz09, BH98, kosar09, PMCH08e, CPH08c, RMHVC06,
 RH98a, OPCH09a, BHVU07b, FPCH08jb, CHP08b, CHLB07a, OPHC09a,
 LRHGT08, Gis99, PMCH08f, MKCHCPO09, LMMVRH08, alfarrabio2001,
 LARH09, HVMLGW05, RMHV06, GH07a, oliveira09b, oliveira09c,
 dag2002, museums98, LGFSSAH08, CHV07, CLH07c, BCVHU08, AH97,
 GRH06, BHVU07c, Ramalho95, CFPBH07d, OPHCC2010, BHVU06a,
 GDH06, OHV06, RARH98, OHCP09, OPHCC09, CH09d, CH07a, FCHV08,
 OVH05, GFH08, MSH05, RRAH99, ALHF02, mp2001, CH07d, Ramalho96
 , graminteractivas1990, KMHVZ04, LPRH07, FH08, SGML97, RH02,
 RRH02, FCHGD09a, FCHGD09b, CH07h, BHVU08, GH07b, CH07g, VH01,
 LPH09a, BUHV08, GRH04, VH03, CHP08i, BHVU06b, JGRH04, LPRH07
 –TM, BHVU07d, KOMPCCH2010, GRH05a, RMHCV06, BHVUM06, VH02,
 PMCH08j, ORH06, BHVU09, CHV08ja, CHP09a, CHP08a, FGH08,
 RSea99, BHVU07a, CPH07f, CBHP09, Ramalho98, RAH98, CH2010a,
 BHV06</P><H3>J</H3><P>Jacinto, M., JGRH03, JGRH04</P><H3>K</
 H3><P>Kosar, T., oliveira09b, MKCHCPO09, KOMPCCH2010, HKMVZ03
 , KMHVZ04, kosar09</P><H3>L</H3><P>Lenic, M., VH02, HVMLGW05
 </P><P>Librelotto, G. R., LRHGT08, LPRH07, JGRH03, GRH05a,
 ALHF02, LMMVRH08, LPH09a, GRH06, LARH09, LGFSSAH08, JGRH04,
 GRH04, LPRH07–TM</P><P>Lopes, S. C., CLH07c, CHLB07a</P><P>
 Lourenco, A., epia–bio–2007, epia–music–2007</P><H3>M</H3><P>
 Machado, H., LMMVRH08</P><P>Maia, B., linguatca</P><P>
 Martins, B., cp3a:terminum2003, cp3a:kvec2003</P><P>Martins,
 F. M., graminteractivas1990</P><P>Martins, M., LMMVRH08</P><P>
 >Mendes, G., MSH05</P><P>Mernik, M., PMCH08j, oliveira09b,
 PMCH08f, MKCHCPO09, KOMPCCH2010, RMHCV06, RMHVC06, HKMVZ03,
 HVMLGW05, VH02, RMHV06, KMHVZ04, kosar09, PMCH08e</P><P>
 Montejano, G., BHVUM06</P><P>Moreira, S., mp2001</P><P>Mota,
 C., linguatca</P><H3>N</H3><P>Neves, F. L., ABNO97b, ABNO97a
 </P><P>Neves, L. F., ABBN98</P><H3>O</H3><P>Oliveira, D.,
 linguatca</P><P>Oliveira, E., OHV06, OVH05</P><P>Oliveira, J
 . N., ABNO97b, ABNO97a</P><P>Oliveira, N., OPHC09a,
 oliveira09b, MKCHCPO09, OHCP09, OPHCC2010, KOMPCCH2010,
 OPHCC09, oliveira09c, cruz09, OPCH09a, kosar09</P><P>Oliveira
 , P., ORH06</P><P>Oliveria, P. J., ORH09a</P><H3>P</H3><P>
 Paulo, R., FPCH08d, FPCH08jb</P><P>Pedroso, T., RPA99</P><P>
 Pereira, M. J. V., oliveira09b, BHVU07d, KOMPCCH2010, RMHCV06
 , oliveira09c, BHVUM06, HKMVZ03, VH02, KMHVZ04, CHV07,
 PMCH08j, BCVHU08, BHVU07c, BHVU09, CHV08ja, CFPBH07d, PMCH08e
 , BHVU08, kosar09, CHP08a, OPHCC2010, CPH07f, RMHVC06, VH01,
 BHVU07a, CBHP09, OPCH09a, BHVU07b, BUHV08, CHP08b, CHP08i,
 BHVU06a, VH03, OPHC09a, OHV06, PMCH08f, MKCHCPO09, OHCP09,
 OPHCC09, BHV06, FCHV08, BHVU06b, HVMLGW05, OVH05, RMHV06</P><
 P>Pereira, R. T., LPH09a</P><P>Pinto, A., linguatca</P><P>
 Pinto, J. S., CPH08c, CHP09a</P><P>Pinto, U., Ulisses96,
 Almeida94c, jspell11</P><H3>R</H3><P>Ramalho, J. C., xmldt99,
 RRAH99, GRH05a, Almeida96c, museums98, Ramalho96, LPRH07,
 JGRH03, SGML97, RH02, GRH06, Ramalho95, RH98a, Ramalho98,
 RAH98, GRH04, LRHGT08, RARH98, LMMVRH08, LARH09, JGRH04,
 LPRH07–TM</P><P>Ramos, C., RRH02</P><P>Ranchhod, E.,
 linguatca</P><P>Rebernak, D., RMHVC06, RMHV06, RMHCV06</P><P>
 >Reis, R., Reis98</P><P>Rocha, I., epia–bio–2007</P><P>Rocha,
 J. G., RPA99, RARH98, Gis99, RRAH99, RSea99, SGML97, mp2001,
 alfarrabio2001, museums98</P><P>Rocha, M., epia–bio–2007</P>

<P>Rocha, P. A., linguatca, freq2002</P><P>Rodrigues, M. F
 ., ORH09a, ORH06, RRH02</P><H3>S</H3><P>Santos, D.,
 linguatca</P><P>Sarmiento, L., linguatca</P><P>Silva, A.,
 RSea99, Gis99</P><P>Silva, F. L., LGFSSAH08</P><P>Silva, N. A
 ., MSH05</P><P>Silva, P., cp3a:terminum2003</P><P>Silva, R.,
 linguatca</P><P>Silveira, M. C., LGFSSAH08</P><P>Simoes, A.
 M. B., xmldt2, linguatca, speaker:sepln2001, mp2001,
 teseambs, dag2002, sepln2003, elpub06-blind, parguess2002,
 eamt06, MP07, elpub06-t2o, APL2k2.Parguess, xata:xmldt, cp3a:
 terminum2003, xata06:navegante, xata07:xmldmx, freq2002,
 APL2k2.Synthesis, xata04:tx, epia-bio-2007, xata04:mtd, epia-
 music-2007, cp3a:natools2003, elpub2003, avalon:avalinha,
 xata05:fs, xata:museudapessoa, xata06:xmldauto, jspell2002,
 alfarrabio2001, sepln06, sepln2004, elpub2002, avalon:jspell,
 lrec06, xata05:tdt</P><H3>T</H3><P>Turchetti, R. C., LRHGT08
 </P><H3>U</H3><P>Uzal, R., BHVU08, BHVU07d, BCVHU08, BHVU07a,
 BHVUM06, BHVU09, BHVU06b, BHVU07c, BHVU07b, BUHV08, BHVU06a
 </P><H3>V</H3><P>Vilela, R., linguatca, xata05:fs</P><P>
 Vizzotto, J., LMMVRH08</P><H3>W</H3><P>Wu, H., HVMLGW05</P><
 H3>Z</H3><P>Zumer, V., HKMVZ03, KMHVZ04</P></BODY>
 </HTML>