

Projeto de LI-3

Martins, José(a78821) Costa, Mariana(a78824)
Quaresma, Miguel(a77049)

June 7, 2017

Contents

1	Introdução	2
2	Desenvolvimento	2
2.1	Bibliotecas	2
2.2	Classes	2
2.2.1	Article.java	2
2.2.2	Contributor.java	3
2.2.3	ArtCompareText.java	3
2.2.4	ArtCompareWords.java	3
2.2.5	ComparatorContributorRevs.java	3
2.2.6	QueryEngineImpl.java	3
3	Conclusão	3

1 Introdução

A **Wikipédia** é uma das muitas fontes de informação disponíveis na Web, possuindo uma quantidade bastante considerável de dados/informação. No entanto para que esta informação seja útil é necessário saber processá-la e acima de tudo fazê-lo em tempo útil, sendo para isso necessário recorrer a estruturas de dados que permitam uma pesquisa rápida mas sem uso excessivo de memória. O objetivo deste projeto é, por isso, implementar uma aplicação (em *Java*) que responde a um conjunto de *queries* relativas a *snapshots* da **Wikipédia** em tempo reduzido e de uma forma correta, recorrendo para isso a um conjunto de recursos já existentes e a estruturas de dados que permitam atingir este objetivo.

2 Desenvolvimento

2.1 Bibliotecas

Neste trabalho usamos várias bibliotecas disponibilizadas pelo Java, sendo de destacar as bibliotecas pertencentes ao StAX no qual nos permitiu realizar o parse dos ficheiros XML percorrendo-os aos poucos em vez de realizar o parse através de um ficheiro guardado em memória com toda a informação importante do ficheiro XML poupando assim tanto em recursos como em tempo de execução.

2.2 Classes

Sentimos a necessidade de criar duas classes principais, sendo que uma representa artigos e outra representa contribuidores. Portanto destas classes podemos construir quantas instancias artigos ou contribuintes quisermos.

2.2.1 Article.java

A classe article, seguindo a linha do trabalho anterior e de modo a responder às queries propostas possui as seguintes variáveis de instancia:

```
private long id; //id do artigo
private String title; //título do artigo
private Map<Long,String> revisions; // revisões do artigo
                                   // chave: id da revisão
                                   // valor: timestamp da revisão
private long nRev; //número de revisões
private long len; //das revisões, o número de caracteres
                //da revisão com mais caracteres
private long words; //das revisões, o número de palavras
                //da revisão com mais palavras
```

Nesta classe é implementado gets, sets, equals, toString, clone e compareTo, sendo de destacar o método:

```
public void setNewLenghtWords(String text)
```

visto ser o método que calcula o tamanho e o número de palavras de um texto passado como argumento e caso os valores sejam maiores que os existentes nas variáveis de instancia `len` e `words` atualiza-os.

2.2.2 Contributor.java

Para além disso temos mais 3 classes nas quais implementamos comparadores necessários para implementar algumas queries.

2.2.3 ArtCompareText.java

2.2.4 ArtCompareWords.java

2.2.5 ComparatorContributorRevs.java

Por fim a classe "mãe".

2.2.6 QueryEngineImpl.java

Possui um HashMap de artigos e um TreeMap de contribuintes de modo a agregar artigos num conjunto e contribuidores também num conjunto, sobre os quais podemos aplicar métodos. Ainda nessa classe possuímos dois longs de modo a guardar os artigos únicos(`artUn`) e os artigos totais(`artTot`).

3 Conclusão