



Universidade do Porto

FEUP Faculdade de Engenharia

Concepção e Análise de Algoritmos
Projeto: Comparação de ficheiros

29 de Maio de 2012

André Freitas - ei10053@fe.up.pt

Olivier Alves - ei10069@fe.up.pt

Tiago Tavares - ei10140@fe.up.pt

Introdução

A sociedade atual cada vez mais depende de informação e conhecimento que é registado em formato de texto. Ora estes textos são alterados e melhorados havendo assim uma necessidade de assinalar de um modo claro as suas mudanças. Assim sendo, no âmbito da Unidade Curricular de Concepção e Análise de Algoritmos da Faculdade de Engenharia da Universidade do Porto, foi realizado um relatório que visa documentar a implementação de um programa simples que assinala de um modo intuitivo as diferenças entre dois ficheiros.

A implementação passará por usar algoritmos de comparação de strings que são da autoria dos autores desse relatório, sendo uma solução *cross-plataform* que usa a consola (standard output) para representar as diferenças entre dois ficheiros com cores.

A linguagem de programação usada é o C++, que pelas suas características de ser baixo/médio nível, permite manipular facilmente as cores da consola e definir parâmetros para a compilação nos ambiente de Windows, Linux e Mac OS X.

Implementação

Tal como referido anteriormente, a implementação é feita em C++ usando algoritmos próprios para comparação de *strings*, usando a consola de output.

Descrição do problema

O utilizador ao introduzir dois ficheiros, sendo o primeiro o original e o segundo o resultado das alterações, obtém a comparação entre esses dois ficheiros. Assinala-se assim a partir do primeiro em relação ao segundo o que foi removido, adicionado e/ou o que ficou igual, contando o número de palavras removidas e adicionadas.

Objetivos

Existem essencialmente dois tipos de implementações já feitas deste problema, que é o comando “*diff*” que está disponível em sistemas *UNIX* e o do *git* (controlo de versões), mas que apenas assinalam as diferenças em termos de linhas e não entre palavras. Ora, é esta a lacuna que se pretende resolver, implementando algo que assinala as diferenças em palavras dando uma representação mais clara ao utilizador das modificações.

Esta solução deverá ser:

1. *user-friendly*;
2. simples em código e estruturação;
3. possível executar em várias plataformas;
4. representar de uma forma mais apelativa as diferenças.

Implementação em C++

Com a implementação em linguagem “c++” foram desenvolvido os ficheiros:

- “*diff.h*”
- “*color.h*”

O primeiro contém os métodos para calcular e representar as diferenças usando strings.

O segundo é utilizado para manipular as cores do texto da consola, para mostrar as diferenças.

Todos os métodos foram associados a classes como membros estáticos, criando assim um contexto de algoritmos e não de programação orientada a objectos.

Algoritmos implementados

Ora, as diferenças entre ficheiros irão articular-se sobre as palavras e não sobre as linhas ou caracteres. Assim sendo, o algoritmo chave deste programa é o seguinte: A cada linha de ambos os ficheiros obtém-se as suas palavras e insere-se num vector de palavras para cada ficheiro. Compara-se assim a igualdade entre posições, se a palavra se encontra na mesma posição, se foi deslocada ou simplesmente se foi removida.

Exemplificando, para as seguintes linhas:

- linha do ficheiro original: “O João corre muito rápido”
- linha do ficheiro final: “O Roberto corre todos os dias muito rápido”

Teria-se as seguintes iterações:

1. A palavra “O” existe na mesma posição nos dois ficheiros;
2. A palavra “João” não existe na posição original e não foi deslocada, por isso foi

- removida e substituída por “Roberto”;
3. A palavra “corre” mantém-se na mesma posição;
 4. A palavra “muito” não existe na mesma posição, porém, foi deslocada mais para a frente, por isso foram adicionadas as palavras “todos”, “os” e “dias” ficando finalmente “muito”. Vamos então alinhar o índice para coincidir entre linhas a ordem de iteração;
 5. A palavra “rápido” em termos de ordem, mantém-se a última.

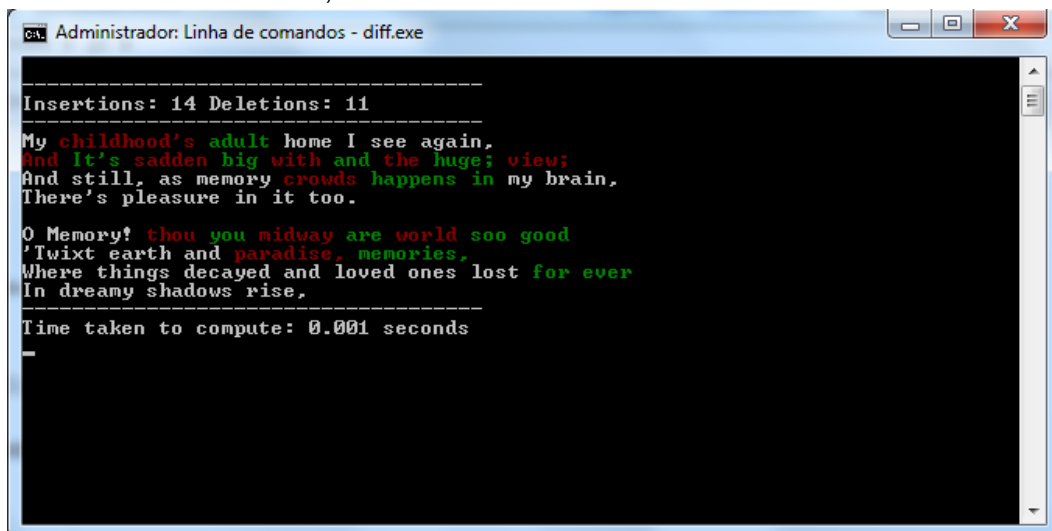
A diferença é então: “=O _Roberto +João =corre +todos +os +dias =muito =rápido”. A função que implementa esta lógica é :

- *string Diff::computeLineDiff(string line1,string line2)*

A complexidade temporal deste algoritmo é $O(n)$ dado que o número de iterações será tanto quanto o número de linhas de cada ficheiro, dada que é uma comparação linha a linha.

Execução do Programa

Para executar o programa basta escrever na linha de comandos “diff a.txt b.txt” que irá dar o seguinte resultado em Windows, Linux e Mac OS X:



```
ca. Administrador: Linha de comandos - diff.exe

-----
Insertions: 14 Deletions: 11
-----
My childhood's adult home I see again,
And it's sadden big with and the huge; view;
And still, as memory crowds happens in my brain,
There's pleasure in it too.

O Memory! thou you midway are world soo good
'Twixt earth and paradise, memories,
Where things decayed and loved ones lost for ever
In dreamy shadows rise,
-----
Time taken to compute: 0.001 seconds
-
```

```
andre@pandora: ~/git/cal-diff/code
-----
Insertions: 14 Deletions: 12
-----
My childhood's adult home I see again,
And It's sadden big with and the huge; view;
And still, as memory crowds happens in my brain,
There's pleasure in it too.

O Memory! thou you midway are world soo good
'Twixt earth and paradise, memories,
Where things decayed and loved ones lost for ever
In dreamy shadows rise,
-----
Time taken to compute: 0 seconds
```

```
bin — bash — 80x24
-----
Insertions: 14 Deletions: 12
-----
My childhood's adult home I see again,
And It's sadden big with and the huge; view;
And still, as memory crowds happens in my brain,
There's pleasure in it too.

O Memory! thou you midway are world soo good
'Twixt earth and paradise, memories,
Where things decayed and loved ones lost for ever
In dreamy shadows rise,
-----
Time taken to compute: 0.0007 seconds
spartacus:bin andre$
```

Desenvolvimento do Projecto

O projeto em causa é de pequena dimensão, precisando apenas de um simples algoritmo que calcule as diferenças entre ficheiros. Assim sendo a divisão de tarefas assentou-se sobre as componentes de manipulação de ficheiros, cálculos das diferenças e das suas representações, que foi atribuído respectivamente a Olivier Alves, André Freitas e Tiago Tavares.

Todos os membros contribuíram de forma semelhante para o sucesso do projeto, usando-se ferramentas colaborativas (Git e Google Hangouts).

Conclusão

O principal problema colocado era como comparar ficheiro textuais. A solução proposta permite a um utilizador uma detecção simples, rápida e eficiente das possíveis diferenças entre dois ficheiros tendo para o efeito um ficheiro base e um novo. Cumprindo todos os objectivos pedidos na formulação do enunciado dado da forma mais eficiente possível, o programa desenvolvido resolve o problema dado e preenche uma lacuna das anteriores soluções que existem no mercado, fazendo uma comparação de palavras e não de linhas sendo multi-plataforma.

Bibliografia

Não foram usadas quaisquer referências bibliográficas para este projeto.