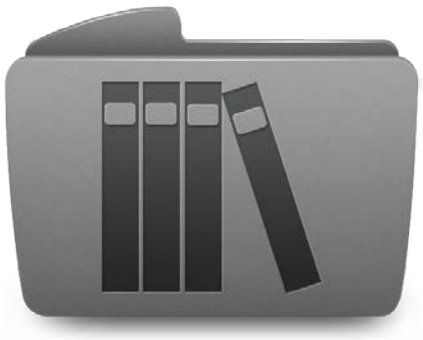
Departamento de Informática
Licenciatura em Engenharia Informática

Ano lectivo 2012/2013

Estruturas de Dados e Algoritmos II



Biblioteca de fotografias

Docente:

Vasco Pedro

Discente:
Marcus Santos, nº 29764

Daniel Ramos, nº29423

ÍNDICE

ÍNDICE DE FIGURAS

ÌNDICE DE QUADROS

INTRODUÇÃO

No âmbito da disciplina de Estruturas de Dados e Algoritmos II, pretende-se a implementação de uma aplicação que auxilie na escolha de fotografias com o intuito de usar as mesmas em artigos de revista, jornais, cartazes, livros, sites Web, etc. O programa servirá como um stock de fotos na escolha de uma fotografia para um determinado tema, sejam eles temas quaisquer.

O programa permitira dois tipos de comandos, um comando que permite inserir fotografias e outro comando que permite a pesquisa de fotografias dado uma lista de temas.

Para auxiliar o desenvolvimento desta aplicação, foram usados dois tipos de TADs que serão explicados mais a frente neste relatório.

DESENVOLVIMENTO

ORGANIZAÇÃO DO PROGRAMA

Com o objectivo de facilitar a organização do código, o código foi dividido em vários ficheiros onde as funções que são comuns estão agrupadas.

FUNCIONAMENTO DO PROGRAMA

FUNCIONAMENTO DAS OPERAÇÕES PERMITIDAS NO PROGRAMA

Comando de inserção das fotografias "I" – O comando de inserção das fotografias verifica as seguintes restrições:

- Verifica se o tamanho do *input* (nome da fotografia) não ultrapassa o máximo permitido para tamanho do nome da fotografia, ou seja, 50 caracteres.
- Verifica se o comando "I" é seguido de qualquer outro caractere que não o caractere espaço.

O utilizador ao introduzir o comando de inserção é alocado espaço em memória para a nova fotografia, de seguida o programa aguarda que o utilizador insira os temas referentes à fotografia.

Inseridos os temas o programa irá guardar a foto em uma LinkedList que contem todas as fotografias e os temas em uma *Trie* com a respectiva foto associada ao tema que ficará guardada numa outra LinkedList.

Comando de pesquisa dos temas "P" – O comando de pesquisa dos temas verifica a seguinte restrição:

- Verifica se o comando "P" é seguido de qualquer outro caractere que não o caractere espaço.

O utilizador ao introduzir o comando para pesquisar uma fotografia, é pedido que o utilizador insira os temas a pesquisar.

Inseridos os temas a pesquisar, o programa irá procurar todas as fotos que contem os temas introduzidos.

Na procura de temas múltiplos, é usado uma FilterTrie que é uma trie que contem os ids das fotos que já foram previamente pesquisadas pelo tema anterior. Sempre que uma fotografia esteja na última FilterTrie, está e adicionada a uma nova FilterTrie que será usada como base para o tema seguinte. As fotos que estiverem sempre na FilterTrie ao longo da pesquisa de todos os temas serão adicionadas a uma LinkedList e será devolvida a mesma como resultado da pesquisa.

Os resultados serão então mostrados no ecrã.

ESTRUCTURA INTERNA DO FICHEIRO DE DADOS

A leitura e inserção de dados na base de dados é feita a partir do ficheiro data.dat que contem os dados relativamente aos temas e as fotografias que pertencem a cada tema.

A estrutura interna do ficheiro que está representado no quadro 1, contem para cada fotografia o tamanho da fotografia seguido do nome da fotografia e contem o tamanho total de todos os temas para essa fotografia, seguido do nome de cada tema.

INT	Tamanho da fotografia
String	Nome da fotografia
INT	Tamanho total dos temas
String	Nome dos temas

Quadro 1: Estrutura interna do ficheiro

Cada tema esta separado por uma barra vertical para diferenciar um tema do seguinte. Um exemplo em modo hexadecimal dos dados é demonstrado na figura 1.

```
0C 00 00 00 70 65 69 78  65 20 61 6F 20 73 61 6C     peixe ao sal
00 09 00 00 00 70 65 69  78 65 7C 73 61 6C 00         peixe|sal
```

Figura 1: Estrutura do ficheiro de dados em modo hexadecimal

TIPOS ABSTRACTOS DE DADOS USADOS

Decidiu-se usar Tries com auxílio de LinkedLists (figura 2) para implementar a base de dados do programa.

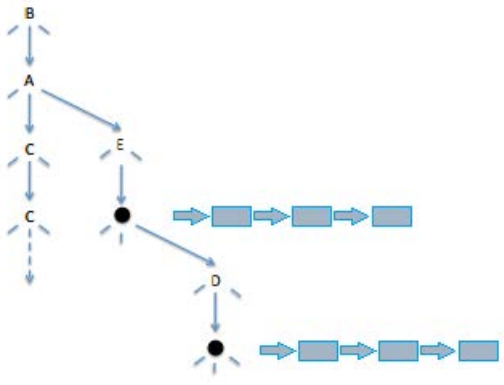


Figura 2: Trie com listas ligadas

As Tries são usadas para implementar a pesquisa por temas em que cada tema tem uma lista ligada de fotos. Como uma fotografia pode conter mais que um tema, as

fotos são guardadas numa lista ligada a parte evitando assim a necessidade de desalocar o espaço alocado em memória para essa fotografia a partir da Trie, o que invalidaria o endereço de memória dessa fotografia nos outros temas.

Decidiu-se usar tries porque funcionam bem para pesquisar dados baseados em strings com um número limitado de caracteres em que apenas os nós que contem os caracteres do tema são acedidos.

Podia-se ter usado Hashtables em vez de tries para indexar os temas mas as vantagens das tries a longo prazo poderão superar as vantagens de ter uma Hashtable, em que numa Hashtable poderão haver colisões e o tempo que leva a gerar o hash de uma string poderá até ser mais lento do que utilizar uma trie.

Usar estritamente listas ligadas está fora de questão porque a procura de um tema ou vários por fotografia implicaria o varrimento total do conteúdo da lista ligada, o que tornaria a pesquisa muito lenta.

COMPLEXIDADE DAS ESTRUCTURAS USADAS

Complexidade da inserção de um tema e de uma fotografia:

A inserção de um tema tem a mesma complexidade da inserção de uma palavra numa Trie porque apenas está dependente do tamanho do alfabeto (d) e do tamanho da palavra (n), isto é O(dn).

A inserção da fotografia tem a mesma complexidade da pesquisa numa trie, isto é O (dn).

Como a inserção do tema acontece primeiro e só depois é inserido a fotografia, a complexidade de inserir uma fotografia é:

$$O(dn) + O(dn) = O(2dn) = O(dn)$$

Complexidade da pesquisa de uma fotografia:

A complexidade de pesquisa de uma fotografia irá depender do número de temas (t) a pesquisar.

Por cada tema a pesquisar a complexidade será a complexidade de uma pesquisa na trie, isto é O (dn). Mais a complexidade de pesquisar uma linkedList O(m), sendo que m é o numero de fotografias na LinkedList associadas ao tema. Mais a complexidade de pesquisar numa FilterTrie pelo id da foto, isto é O (dn). Mais a complexidade de inserir o id da fotografia na FilterTrie, isto é O (dn).

E finalmente a complexidade de inserir um resultado na LinkedList que é constante, isto é O (1).

Logo a complexidade será $O(t * dn * m * dn * dn * 1) = O(t\ (dn)^3 m)$

Caso a pesquisa seja só de um tema, neste caso não é aplicado filtros e a complexidade passa a ser de O(dn * m * 1) = *O(dnm)*

Acessos ao disco

Leitura: Todo o conteúdo da base de dados é lido quando o programa inicia. Por cada fotografia é feito quatro acessos ao disco rígido.

Escrita: A escrita no disco só é feita quando se insere uma nova fotografia. Por cada fotografia é feito quatro escritas no disco rígido.

CONCLUSÃO

A necessidade de encontrar uma estrutura viável para implementar a pesquisa de fotos por tema de acordo com o esperado levou a que o grau de dificuldade subisse ligeiramente devido a alguma dificuldade em encontrar uma forma viável de pesquisar uma fotografia com vários temas ao mesmo tempo. A solução foi uma segunda implementação da Trie apenas com números que actuasse como filtro para verificar se as fotos estavam em todos os temas.

Depois já da finalização e entrega do trabalho, descobriu-se que a inserção dos temas e das fotos podia ter sido simplificado para que a fotografia fosse directamente inserida quando a inserção do tema ocorresse, em vez de inserir o tema e só depois a fotografia.