Universidade Federal Fluminense

Instituto de Ciência e Tecnologia – Departamento de Computação

Trabalho de Programação com Arquivos (primeira parte)

Indexação

(2 pontos. Data de entrega 03/12/2021) A indexação é um processo computacional de se construir índices sobre uma base de dados. Índices são estrutura de dados que permitem a realização mais eficiente de operações como consulta (busca) por meio de um "resumo" dos dados existentes na base.

Na primeira parte do trabalho, vocês devem fazer um programa que tenha três opções principais: 1) criar um índice para um arquivo texto; 2) utilizar um índice existente para realizar buscas por palavras; 3) encerrar o programa.

1) Caso o usuário escolha criar um índice, o programa deve seguir as etapas abaixo:
    a. Perguntar ao usuário o nome do arquivo texto.
    b. Abrir o arquivo texto e ler cada palavra existente. Cada palavra distinta deve ser representada pelo struct Palavra (abaixo).
    c. O programa deve ter uma lista de palavras chamada Lista (pode ser lista encadeada, vetor ou qualquer outra estrutura de dados que mantenha as palavras ordenadas alfabeticamente). Lista deve possuir um conjunto de informações do tipo struct Palavra.
    d. Sempre que uma palavra for lida do arquivo, o programa deve checar se ela já existe em Lista. Se não existir, deve ser inserida em ordem alfabética, a quantidade de ocorrências dessa palavra deve ser definida com o valor 1 e a linha onde a palavra está deve ser registrada também. Se a palavra já existir, o programa deve apenas aumentar q quantidade de ocorrências dessa palavra acrescentar a linha onde ela foi encontrada novamente no struct correspondente.
    e. Depois de ler todas as palavras do arquivo, o programa deve gerar o índice correspondente a esse arquivo texto, por meio do arquivo binário "índice.dat".
    f. A organização dos dados nesse arquivo binário será o seguinte:
        i. Número inteiro contendo o total de palavras do índice
        ii. Cada palavra será representada por: um número inteiro contendo a quantidade de caracteres da palavra **+ 1**, os caracteres da palavra em si junto com o **'\0'**, um número inteiro indicando a quantidade de vezes em que essa palavra foi encontrada no arquivo texto, e, finalmente as linhas onde a palavra foi encontrada (representada por um número inteiro).
        iii. Um esquema visual está apresentado abaixo.
    g. Após criar o arquivo binário, o programa deve voltar ao menu principal.
2) Caso o usuário escolha utilizar um índice, o programa deve seguir as etapas abaixo:
    a. Liberar a memória alocada para a lista atual (se tiver sido alocada) e abrir o arquivo "índice.dat";

b. Ler a quantidade de palavras (primeiro número inteiro do arquivo binário);
c. Ler e mostrar na tela cada um dos registros subsequentes, contendo: um número inteiro que indica a quantidade de caracteres da palavra em questão **(contando com o \0),** os caracteres em si **(incluindo o \0),** um número inteiro que indica a quantidade de ocorrências dessa palavra no arquivo texto original e, finalmente, os números inteiros que representam as linhas onde a palavra foi encontrada. Esses dados também devem ser inseridos numa Lista de palavras.
d. O programa deve então perguntar ao usuário qual palavra ele quer pesquisar;
e. Se a palavra for encontrada, o programa deve mostrar na tela a quantidade de vezes em que ela existia no arquivo texto original e em quais linhas. Se não for encontrada, o programa deve avisar ao usuário.
f. Após mostrar o resultado da busca, o programa deve voltar ao menu principal;

3) Caso o usuário escolha encerrar o programa, toda a memória (dinamicamente alocada) deve ser liberada. O arquivo "índice.dat" NÃO deve ser deletado.

```
struct Palavra{

char letras[50];    //caracteres que formam a palavra (pode ser utilizado std::string de C++)

int qtdOcorrencias; //quantidade de vezes que essa palavra apareceu no arquivo

int *linhas; //vetor dinâmico (pode ser utilizado std::vector de C++ em substituição a esses 2 últimos campos)
}
```

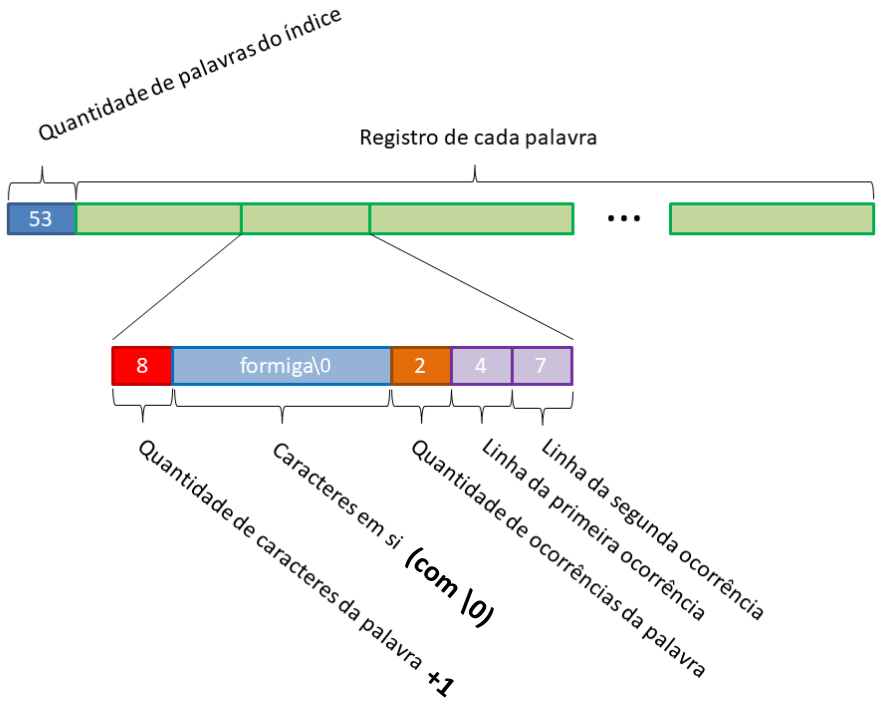

Figura 1: Esquema representativo do arquivo "índice.dat"