

# Trabalho Prático 2: Índice Invertido

## Algoritmos e Estruturas de Dados III – 2017/1

Jéssica Taís Carvalho Rodrigues

19 de Junho de 2017

### 1 Introdução

Em aplicações de bate papo, acontecem muitas conversas ao mesmo tempo e realizar busca de alguma palavra não é uma tarefa fácil. Considerando que já existe um algoritmo de busca, o ponto central desse problema é criar um índice invertido para usar nesta busca.

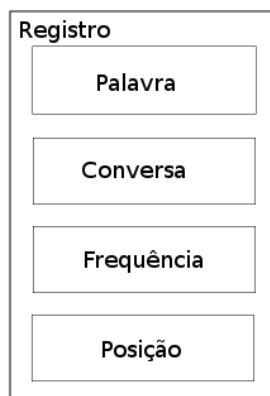
Este índice invertido precisa cumprir alguns requisitos, como estar ordenada de forma lexicográfica, porém a quantidade de memória do celular é limitada e a organização precisa ser feita usando ordenação externa.

Outras condições de formato também precisam ser satisfeitas e estas serão melhor descritos e explicados no decorrer deste documento.

### 2 Solução do problema

O programa conta com o programa principal mais três módulos, sendo eles: resolução, arquivo e registro.

Para solução do problema, inicialmente cada conversa é estruturada no formato que consiste em [Palavra], [Documento que a palavra está], [Frequência da palavra] e a [posição da palavra na memória] e este formato compõe a estrutura de Registro. A estrutura pode melhor ser entendida na imagem abaixo:



Após esta organização é já inserindo ordenada no arquivo fazendo uso de ordenação externa pelo algoritmo do QuickSort externo.

Para o cálculo da frequência foi usada a mesma ideia de ponteiros para leitura e escrita que é utilizada no QuickSort externo, sendo que um ponteiro lê e calcula o total da frequência de uma determinada palavra e no final o outro vai preenchendo todos com esta soma.

Após cada uma das conversas estruturadas, para se criar o índice invertido em seu resultado final, os arquivos são unidos dois a dois fazendo um novo arquivo e este salvo no caminho de saída dado inicialmente pela entrada.

### 3 Análise de complexidade

Variáveis de relevância para a análise:

Variável	Significado
R	Número de registros num documento
F	Frequência dos registros num documento

A função main tem complexidade  $O(R^2)$ , por ser o maior tempo dentre as funções executadas no programa.

#### 3.1 Tempo

##### 3.1.1 Registro

Função	Complexidade
comparaRegistro	$O(1)$
imprimeRegistro	$O(1)$
criaBufferOrdenado	$O(1)$
bufferVazio	$O(1)$
insereBuffer	$O(R)$
removeMaxBuffer	$O(R)$
removeMinBuffer	$O(R)$
liberaBuffer	$O(R)$

##### 3.1.2 Arquivo

Função	Complexidade
estruturaArquivo	$O(R)$
computaFrequencia	$O(R^2)$
ordenaArquivo	$O(R \log R)$
uneArquivos	$O(\log R)$
transformaArquivo	$O(R)$

##### 3.1.3 Resolução

Função	Complexidade
resolve	$O(R^2)$

#### 3.2 Espaço

O programa é estruturado em registros (R) que possuem 4 atributos e somam 32 bytes na memória. Considerando isso e que há manipulações nesses registros a complexidade de espaço do programa é dada por  $O(R \cdot F)$ .

## 4 Avaliação experimental

Para a avaliação experimental, foram realizados testes simples, chamados de testes toys. Os resultados estão logo abaixo.

### 4.1 Testes toys

Nestes testes, foram 10 entradas diferentes e a quantidade de documentos (conversas) estava entre 3 a 6 e a quantidade de memória primária disponível dada estava entre 160 e 640 bytes. Todos estes executaram em menos de 1 segundo.

### 4.2 Análise

De acordo com a análise de complexidade e pela percepção dos testes, o algoritmo é polinomial quadrático e seu uso do espaço ocorre de acordo com a memória primária disponível que é dada na entrada.

## 5 Conclusão

Realizar esse trabalho foi um desafio interessante. Apesar do foco ser ordenação externa, minha maior dificuldade foi na manipulação de arquivos, mas após intensa pesquisa e estudos sobre, tive um grande aprendizado. Foi de grande proveito pois pude reforçar meus conhecimentos sobre o assunto e ratificá-los.

## 6 Referências

- Projeto de Algoritmos - Nivio Ziviani
- Manipulando arquivos em C, disponível em
  1. [https://www.vivaolinux.com.br/artigo/Manipulando-arquivos-em-C-\(parte-1\)](https://www.vivaolinux.com.br/artigo/Manipulando-arquivos-em-C-(parte-1))
  2. [https://www.vivaolinux.com.br/artigo/Manipulando-arquivos-em-C-\(parte-2\)](https://www.vivaolinux.com.br/artigo/Manipulando-arquivos-em-C-(parte-2))