
Índice Remissivo com Árvores AVL

Murielly Oliveira Nascimento



UNIVERSIDADE FEDERAL DE UBERLÂNDIA
FACULDADE DE COMPUTAÇÃO
TRABALHO DE ALGORITMOS E ESTRUTURA DE DADOS 2.

Uberlândia
2022

Murielly Oliveira Nascimento

**Índice Remissivo
com Árvores AVL**

Trabalho apresentado e desenvolvido durante a
matéria de Algoritmos e Estrutura de Dados 2
no ano de 2022/1 sobre o tema Árvores AVL.

Área de concentração: Sistemas de Informação

Uberlândia
2022

Resumo

Um índice remissivo de um livro, lista os termos e tópicos que são abordados num documento, em ordem alfabética, juntamente com as páginas em que aparecem no livro. Seguindo a ideia de criar um índice remissivo para um documento texto, considerando apenas os termos (palavras), cada termo encontrado no documento teria uma lista de linhas em que foi encontrado. Devem ser desconsideradas diferenças entre letras maiúsculas e minúsculas, e se a palavra aparece mais de uma vez numa determinada linha, esta linha aparece apenas uma vez no índice.

Este projeto envolve a criação de um índice remissivo para um documento utilizando árvore binária AVL. Cada nó da árvore AVL deve conter uma palavra e um vetor para armazenar as linhas em que aquela palavra aparece no documento.

Apenas as funções de busca e inserção serão utilizadas neste projeto. Não haverá remoção de palavras do índice e pontuações poderão ser utilizadas como delimitadores de palavras, mas não constam do índice.

O índice deve ser retornado em um arquivo texto (.txt) com as três linhas finais (total de palavras, total de palavras distintas, tempo de construção), e o documento de entrada também deve ser do tipo texto simples (.txt)

Palavras-chave: Árvores AVL.

Solução

Repositório: <<https://replit.com/@Murielly/Trabalho-2#main.c>>

Foi usada a linguagem C para a solução do problema. O código é dividido em arquivos para as TADs e para a função main. A estrutura da árvore é como segue:

```
1
2 #define TAM 50
3
4 struct NO
5 {
6     char palavra[TAM];
7     int linha[TAM];
8     int pos;
9     int altura;
10    struct NO *esq;
11    struct NO *dir;
12 };
13
14 typedef struct NO *ArvAVL;
15
16 ArvAVL *cria_ArvAVL();
17 void libera_ArvAVL(ArvAVL *raiz);
18 int insere_ArvAVL(ArvAVL *raiz, char *valor, int linha);
19 int estaVazia_ArvAVL(ArvAVL *raiz);
20 int altura_ArvAVL(ArvAVL *raiz);
21 int totalNO_ArvAVL(ArvAVL *raiz);
22 int consulta_ArvAVL(ArvAVL *raiz, char *valor);
23 void emOrdem_ArvAVL(ArvAVL *raiz);
```

No arquivo main os nomes dos arquivos de leitura e escrita são, poema.txt e indice.txt, respectivamente. A função insere, abre o arquivo com o poema e o lê. As suas palavras são inseridas na estrutura árvore.

```
1 int insere(ArvAVL *avl)
2 {
```

```

3 // Abre arquivo
4 FILE *arq = fopen(filename , "r");
5 char *conteudo_linha = malloc(MAX);
6 char *palavra = malloc(MAX);
7 int linha = 1;
8 int Nropalavra = 0;
9
10 // Le arquivo e preenche arvore
11 while (fscanf(arq, "%[^\n] ", conteudo_linha) != EOF)
12 {
13     palavra = strtok(conteudo_linha, " ,.?!?/");
14     while (palavra != NULL)
15     {
16         insere_ArvAVL(avl, palavra, linha);
17         palavra = strtok(NULL, " ,.?!?/");
18         Nropalavra++;
19     }
20     linha++;
21 }
22 fclose(arq);
23
24 return Nropalavra;
25 }

```

A função `arqIndice` passa as informações da árvore para um arquivo texto. Criando um índice remissivo.

```

1 int arqIndice(ArvAVL *avl, int NroPalavra, clock_t time)
2 {
3
4     char total[MAX] = {'\0'};
5     char distintas[MAX] = {'\0'};
6
7     // Abre arquivo indice
8     FILE *indice = fopen(fileIndice, "w");
9
10    // Cabecalho do arquivo
11    fprintf(indice, "_____\n");
12    fprintf(indice, "Indice:\n");
13
14    // Escreve dados da arvore para o arquivo
15    escreve(avl, indice);
16
17    // Numero de palavras total
18    snprintf(total, MAX, "%d", NroPalavra);
19    fprintf(indice, "Numero total de palavras:%s\n", total);
20
21    // Numero de palavras distintas

```



```

22     NroPalavra = totalNO_ArvAVL(avl);
23     snprintf(distintas, MAX, "%d", NroPalavra);
24     fprintf(indice, "Numero de palavras distintas:%s\n", distintas);
25
26     // Calcula o tempo levado para realizar esse programa
27     clock_t tempo = clock() - time;
28     double tempo_total = ((double)tempo) / CLOCKS_PER_SEC;
29     char segundos[MAX] = {'\0'};
30     snprintf(segundos, MAX, "%lf", tempo_total);
31
32     fprintf(indice, "Tempo de construcao do indice usando arvore AVL: %s\n", segundos);
33
34     // Fechamos o arquivo
35     fclose(indice);
36 }

```

Por fim, a função escreve que é chamada pela função arqIndice faz a escrita dos dados no arquivo indice.txt

```

1 void escreve(ArvAVL *raiz, FILE *arq)
2 {
3     if (raiz == NULL)
4         return;
5     if (*raiz != NULL)
6     {
7         escreve(&((*raiz)->esq), arq);
8
9         // Escreve a palavra no arquivo
10        fprintf(arq, "%s ", (*raiz)->palavra);
11
12        // Percorre a lista de linhas em que a palavra aparece
13        for (int i = 0; i < (*raiz)->pos; i++)
14        {
15            char linha[MAX] = {'\0'};
16            snprintf(linha, MAX, "%d", (*raiz)->linha[i]);
17            fprintf(arq, "%s ", linha);
18        }
19        fprintf(arq, "\n");
20        escreve(&((*raiz)->dir), arq);
21    }
22 }

```

A função main é como segue:

```

1
2 #define MAX 50
3
4 const char *filename = "poema.txt";

```

```

5 const char *fileIndice = "indice.txt";
6
7 int main(int argc, char *argv[])
8 {
9     clock_t time;
10    time = clock();
11
12    // Cria arvore
13    ArvAVL *avl;
14    avl = cria_ArvAVL();
15
16    // Inserimos dados do arquivo na arvore
17    int Nropalavra = insere(avl);
18
19    // Criamos o arquivo indice
20    arqIndice(avl, Nropalavra, time);
21
22    // Finalizamos o programa
23    libera_ArvAVL(avl);
24    printf("Indice Remissivo criado com sucesso!\n");
25    return 0;
26 }

```

1.1 Testes

A Rua dos Cataventos

Da vez primeira em que me assassinaram,
 Perdi um jeito de sorrir que eu tinha.
 Depois, a cada vez que me mataram,
 Foram levando qualquer coisa minha.

Hoje, dos meu cadáveres eu sou
 O mais desnudo, o que não tem mais nada.
 Arde um toco de Vela amarelada,
 Como único bem que me ficou.

Vinde! Corvos, chacais, ladrões de estrada!
 Pois dessa mão avaramente adunca
 Não haverão de arrancar a luz sagrada!

Aves da noite! Asas do horror! Voejai!

Que a luz trêmula e triste como um ai,
A luz de um morto não se apaga nunca!

O arquivo indice.txt gerado a partir desse poema é como segue:
Índice Remissivo:
A 1 15
Arde 8
Asas 13
Aves 13
Cataventos 1
...
Número total de palavras:101
Número de palavras distintas:78
Tempo de construção do índice usando árvore AVL: 0.000588 segundos

A Dream Within a Dream

Take this kiss upon the brow!
And, in parting from you now,
Thus much let me avow —
You are not wrong, who deem
That my days have been a dream;
Yet if hope has flown away
In a night, or in a day,
In a vision, or in none,
Is it therefore the less gone?
All that we see or seem
Is but a dream within a dream.

I stand amid the roar
Of a surf-tormented shore,
And I hold within my hand
Grains of the golden sand —
How few! yet how they creep
Through my fingers to the deep,
While I weep — while I weep!
O God! Can I not grasp
Them with a tighter clasp?
O God! can I not save

One from the pitiless wave?
Is all that we see or seem
But a dream within a dream?

O arquivo indice.txt gerado a partir desse poema é como segue:

Índice:

A 4

All 11

And 2 15

But 25

Can 20

Dream 4 4

God 20 22

Grains 16

How 17

I 13 15 19 19 20 22

..

Número total de palavras:154

Número de palavras distintas:100

Tempo de construção do índice usando árvore AVL: 0.000466 segundos

Le Chat

Viens, mon beau chat, sur mon coeur amoureux;
Retiens les griffes de ta patte,
Et laisse-moi plonger dans tes beaux yeux,
Mêlés de métal et d'agate.

Lorsque mes doigts caressent à loisir
Ta tête et ton dos élastique,
Et que ma main s'enivre du plaisir
De palper ton corps électrique,

Je vois ma femme en esprit. Son regard,
Comme le tien, aimable bête
Profond et froid, coupe et fend comme un dard,

Et, des pieds jusques à la tête,
Un air subtil, un dangereux parfum
Nagent autour de son corps brun.

O arquivo indice.txt gerado a partir desse poema é como segue:

Índice:

Chat 1

Comme 11

De 9

Et 4 8 13

Je 10

Le 1

Lorsque 6

Mêlés 5

Nagent 15

..

Número total de palavras:93

Número de palavras distintas:79

Tempo de construção do índice usando árvore AVL: 0.000449 segundos