



Trabalho 3 – Índice Remissivo

SCC-502 Algoritmos e Estruturas de Dados 1 M. Cristina/Jorge

1 Introdução

O trabalho deverá ser feito **individualmente** e submetido para o sistema SSP (Sistema de Submissão de Programas), do ICMC.

Data de entrega: 12 de dezembro de 2013.

1.1 Instruções de submissão

Para entrar no sistema e submeter os trabalhos, seguir os passos:

1. Entre no site <<https://ssp.icmc.usp.br/>>;
2. Faça login/cadastro no site;
3. Matricule-se na disciplina SCC0502 - Algoritmos e Estruturas de dados (M. Cristina) (2013/2-Turma A)
4. Clique em Submeter exercicio -> "Trabalho 3. Índice Remissivo";
5. Submeta seu exercício;
6. O sistema fará a correção automática do seu programa, seguindo alguns casos de teste (padrões de entrada e saída) e apresentará o número de acertos e erros. Você pode submeter o programa quantas vezes desejar (até o prazo de entrega).

1.2 Critério de Avaliação

Dados:

$N_{acertos}$ = Número de casos de teste corretos;

N_{casos} = Número total de casos de teste;

$T3$ = Nota do trabalho 2.

$$T3 = \frac{N_{acertos} * 10}{N_{casos}}$$

2 Descrição do Trabalho

Várias aplicações necessitam de um relatório de referências cruzadas. Por exemplo, a maioria dos livros apresenta um índice remissivo que corresponde a uma lista alfabética de palavras chave ou palavras relevantes do texto com a indicação dos locais no texto onde cada palavra chave ocorre.

Seu programa deve, a partir de um conjunto de palavras e a página onde elas aparecem, fazer um índice remissivo.

Implemente o índice como uma árvore de busca binária.

3 Entrada

A entrada deve ser lida do teclado (`stdin`).

A entrada contém apenas um caso de teste, composto por várias linhas. Cada linha tem início com um comando: **inserir**, **buscar** **remissivo** **sair**.

3.1 inserir

O comando `inserir` tem o seguinte formato:

i <palavra> <pagina>

em que <palavra> é uma string que contém apenas os caracteres [a ... z] (não é sensível a caixa | "a" == "A") e <pagina> é o número da página em que a palavra aparece.

<pagina > ∈ {1, 2, 3, 4, 5, ...1000}.

| <palavra > | ≤ 50

3.2 buscar

O comando `buscar` tem o seguinte formato:

b <palavra>

E deve imprimir os números das páginas em que a palavra <palavra > aparece separados apenas por vírgula.

3.3 remissivo

O comando `remissivo` tem o seguinte formato:

r

e deve imprimir o índice remissivo ordenado por ordem alfabética (**todos os caracteres em caixa baixa (lowercase)**).

3.4 sair

O comando tem o formato:

s

e deve terminar o programa (e retornar 0).

4 Saída

A saída deve ser apresentada na saída padrão, através do comando `printf`

Apenas os comandos **buscar** e **remissivo** devem apresentar algo na tela.

4.1 buscar

Suponhamos que a palavra "palavra" aparece nas páginas 1, 7 e 10.

O programa deve imprimir uma linha com os números em ordem crescente separados apenas por vírgula, na forma

```
//-----exemplo.c-----  
printf("%d,%d,%d\n",1,7,10);
```

Caso a palavra não exista no texto, deve-se imprimir o número 0.

4.2 remissivo

O comando **remissivo** deve imprimir o índice remissivo em ordem alfabética. Como a estrutura utilizada é uma árvore binária de busca, isto equivale a realizar um percurso em-ordem na árvore.

```
void emOrdem(No *n){  
    if (n != NULL){  
        emOrdem(n->esq);  
        printf("%s\n",n->palavra);  
        emOrdem(n->dir);  
    }  
}
```

Deve-se imprimir uma palavra por linha, na forma: $\langle \textit{palavra} \rangle \langle \textit{pagina}_1 \rangle, \langle \textit{pagina}_2 \rangle, \dots, \langle \textit{pagina}_n \rangle$

As páginas devem ser apresentadas em ordem crescente.

Por exemplo:

```
abacaxi 2  
amor 3  
amortece 8,9,10  
casa 9  
casado 11  
palavra 1,7,10
```

5 Exemplo de execução

Entrada	Saída
i abacate 1	2,3
i banana 3	1,2,5
i fruta 5	0
i fruta 1	abacate 1
i fruta 2	banana 2,3
i melancia 4	fruta 1,2,5
i banana 2	melancia 4
b banana	
b fruta	
b elefante	
r	
s	
i jorge 1	1,2,3,4,5
i henrique 2	1,2
i ana 3	2
i beatriz 4	ana 3,5
i ana 5	beatriz 4
i julio 2	henrique 2
i julio 1	jorge 1
i nomes 1	julio 1,2
i nomes 2	nomes 1,2,3,4,5
i nomes 3	
i nomes 4	
i nomes 5	
b nomes	
b julio	
b henrique	
r	
s	

6 Plágio

ATENÇÃO: O sistema SSP utiliza a ferramenta MOSS, de Stanford, para detecção de plágio. Para mais informações, visite: <http://theory.stanford.edu/aiken/moss/>.