

MAC 121 – Algoritmos e Estruturas de Dados I**Segundo semestre de 2018****Tabelas de símbolos – Entrega: 25 de novembro de 2018**

O objetivo deste exercício-programa é testar diferentes implementações para a estrutura de dados **tabela de símbolos**. Nesta estrutura fazemos as operações de inserção e busca de elementos. A estrutura pode ser implementada utilizando, entre outros,

- vetor desordenado;
- vetor ordenado;
- lista ligada desordenada;
- lista ligada ordenada;
- árvore de busca binária.

Neste exercício você deverá implementar de todas as formas acima, e testar empiricamente as implementações com relação ao tempo que as operações de inserção e busca demoram.

Para fazer seus testes, você deverá implementar uma tabela de símbolos para calcular a frequência que ocorrem palavras em um texto muito grande. Assim, o programa deverá receber na linha de comando o nome de um arquivo de entrada (em formato UTF 8) e produzir uma lista de palavras, uma por linha, com a palavra e o número de ocorrências no texto. Esta lista poderá ser apresentada ordenada em ordem alfabética ou pelo número de ocorrências (na entrada é especificada a ordem desejada: “A” para alfabética ou “O” para ocorrências). E, para implementar a tabela de símbolos o usuário poderá escolher uma das 5 estruturas listadas acima:

- VD: vetor desordenado
- VO: vetor ordenado
- LD: lista ligada desordenada
- LO: lista ligada ordenada
- AB: árvore de busca binária

Você poderá utilizar textos grandes, por exemplo, do Projeto Gutenberg:

<http://www.gutenberg.org>.

Neste site você pode fazer o *download* de vários livros que poderá utilizar como entrada em seu EP.

Junto com seu EP você deverá entregar mais uma vez um **relatório** mostrando os testes que você fez, e as conclusões que obteve. Faça testes com textos de diferentes tamanhos para poder ter uma ideia da complexidade assintótica de cada implementação. Faça testes com textos em diferentes línguas. Será que faz diferença?

Exemplo de entrada

A chamada do seu programa usará a linha de comando para os 3 parâmetros: nome do arquivo a ser lido, tipo de estrutura usada e tipo de ordenação desejada. Assim, por exemplo, se seu programa se chama **ep4**, a chamada para implementar uma tabela de símbolos usando uma lista ligada ordenada e ordenando as palavras pelo número de ocorrências ficaria:

```
./ep4 OsLusiadas.txt L0 0
```

Exemplo de saída

```
a 1278
o 987
e 812
de 801
as 732
.
.
.
```

Detalhes da entrega e correção

- No paca vocês poderão entregar dois arquivos: o código fonte `.c` do seu programa e o relatório (pdf);
- Sigam estritamente o formato descrito no exemplo;
- Na correção seu programa será compilado em linux usando
`gcc -Wall <arquivo>.c -o ep3`