MAC122/BMAC SEGUNDO EXERCÍCIO-PROGRAMA

OCORRÊNCIAS PRÓXIMAS

Y. KOHAYAKAWA

Data de entrega. Você deve entregar este EP até o dia 5/11/2003, pelo Panda.

1. Introdução

Neste segundo exercício-programa, você verá como usar *tabelas de símbolos* em seus programas. Neste EP, você também precisará resolver um pequeno problema de otimização.

2. O problema a ser resolvido

O programa que você escreverá neste exercício deve resolver o problema que descrevemos a seguir.

O usuário dará inicialmente um texto T, possivelmente longo. O usuário irá então fornecer uma seqüência $C_1, C_2, \ldots$ de conjuntos de palavras. Para cada conjunto C_i, o seu programa deve imprimir um trecho do texto T que contém todas as palavras em C_i. Caso o texto não contenha todas as palavras em C_i, seu programa deve imprimir uma mensagem dizendo isso. (O programa pode devolver o resultado para C_i logo após ler C_i.) O problema de otimização vem aqui: adicionalmente, vamos impor a condição de que

(*) *o trecho devolvido como resposta para a entrada C_i deve ter o menor número de caracteres possível.*

Observações. Uma *palavra* é uma seqüência maximal de letras. Você deve distinguir letras maiúsculas de minúsculas, de forma que `Vitória` e `vitória` devem ser consideradas palavras diferentes.

Execução. O texto T de entrada deve vir em um arquivo, cujo nome será especificado pelo usuário. O seu programa deve receber os conjuntos C_i no `stdin`, com os C_i separados por linhas em branco. Assim, seu programa poderia ser executado, por exemplo, como segue:

```
meu_prompt $ ep2 bible.txt < conjuntos
```

onde `conjuntos` é o arquivo de conteúdo

```
In created earth

divided sea Moses
```

As respostas poderiam ser[1]

Date: Versão de 14 de outubro de 2003.

[1] Eu não verifiquei!

```
In the beginning God created the heaven and the earth
```

e

```
Moses stretched out his hand over the sea; and the LORD
caused the sea to go back by a strong east wind all that
night, and made the sea dry land, and the waters were divided
```

3. Estrutura geral do programa

Você deve estruturar o seu programa da seguinte forma.

3.1. Objetos das tabelas de símbolos. Os objetos a serem armazenados em sua tabela de símbolos devem ser do tipo `Item`, implementados em `Item.c` e manipulados através da interface `Item.h`.

3.2. Tabela de símbolos. A sua tabela de símbolos deve ser implementada em `ST.c` e o acesso a ela deve ser *estritamente através da interface* `ST.h`. Projete a interface `ST.h` de forma que ao se alterar a implementação da tabela de símbolos, não será necessário se alterar outras partes do programa.

Nesta etapa, procure implementar sua tabela de símbolos de forma que o seu programa possa manipular arquivos de tamanho razoável, mas não necessariamente tão grandes como um livro. Com o que vimos em sala até o momento, provavelmente não sabemos como escrever um programa que possa realmente processar um arquivo como o *King James' Bible*, que contém algo como 800.000 palavras.

3.3. Arquivos de teste. Uma boa fonte de arquivos para testar o seu programa é o Projeto Gutenberg.

4. Observações

1. *Este EP é individual.*
2. Preste atenção na modularização descrita neste enunciado. Se você seguir estas instruções, será mais fácil você fazer o EP3.
3. Seja cuidadoso com sua programação (correção, documentação, apresentação, clareza do código, etc), dando especial atenção a suas estruturas de dados. A correção será feita levando isso em conta.
4. *Não imponha restrições arbitrárias sobre a entrada.* Quando usado forma inesperada, o seu programa deve parar de forma 'graciosa'.
5. Organizem-se na turma para fazer bons testes. Comparem entre vocês o desempenho de seus programas.
6. Seja criativo. Se seu programa fizer algo a mais do que foi pedido você poderá ganhar algum bônus na nota. Por outro lado, você *deve* respeitar as especificações dadas neste enunciado.
7. Entregue seu EP seguindo os moldes usuais: cabeçalho claro, instruções claras para o monitor poder compilar e executar seu programa, e eventuais arquivos de teste (não muito grandes).

Observação final. Tragam dúvidas para as nossas aulas. Eventualmente, ajustes no enunciado poderão ocorrer ao discutirmos este EP em sala.