MAC323 EXERCÍCIO-PROGRAMA 1

PALAVRAS E SUAS FREQÜÊNCIAS

Y. KOHAYAKAWA

Data de entrega: 19/3/2012 (23:55)

Introdução. Este EP supõe que você está familiarizado com a manipulação de palavras em um arquivo-texto. Será mais simples, neste EP, supor que os textos a serem manipulados são textos em inglês, e portanto não haverá caracteres acentuados.

Descrição do problema. Dado um texto, queremos montar a lista de palavras que ocorrem no texto, com suas respectivas freqüências, em ordem decrescente de freqüência. Por exemplo, suponha que a entrada é o texto abaixo:

```
We must not underrate the gravity of the task which lies before us or the
temerity of the ordeal, to which we shall not be found unequal.  We must
expect many disappointments, and many unpleasant surprises, but we may be sure
that the task which we have freely accepted is one not beyond the compass and
the strength of the British Empire and the French Republic.
```

A saída deve então ser

```
the 9
and 3
not 3
of 3
we 3
which 3
We 2
be 2
many 2
must 2
...
```

Note que, por exemplo, *the* aparece 9 vezes no texto e é a palavra mais freqüente. No caso de empate, as palavras devem ser listadas em ordem alfabética (note que *and*, *not*, *of* etc ocorrem todas 3 vezes e que esta lista está ordenada).

Seu programa. Seu programa deve receber a entrada no `stdin` e a saída deve ser enviada para o `stdout`. Assim, uma execução de seu programa poderia ser

```
prompt$ ep1 < Churchill.txt > out.txt
```

Versão de 26 de maio de 2012, 18:33.

Opções. Implemente, pelo menos, a opção de linha de comando `-n`, que diz ao seu programa quantas palavras devem ser impressas. Por exemplo,

```
prompt$ ep1 -n7 < Churchill.txt
```

Deve imprimir as 7 palavras mais freqüentes em `Churchill.txt` (mais precisamente, as 7 primeiras linhas da saída sem a opção `-n7`). Caso o argumento de `-n` for maior que o número de linhas da saída sem a opção, a opção não deve ter nenhum efeito.

Idealmente, o usuário deve poder escolher entre os modos *case insensitive* e *case sensitive*: isto é, ignorando a diferença entre maiúsculas e minúsculas ou não. Se você implementar a escolha entre tais modos, você ganhará nota extra. O padrão deve ser *case sensitive* e o modo *case insensitive* (caso você implemente) deve ser especificado com o argumento de linha de comando `-I`.

Finalmente, como funciona seu programa com palavras com caracteres acentuados?

Observações

1. *Este EP é estritamente individual.* Programas semelhantes receberão nota 0.
2. 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.
3. Comparem entre vocês o desempenho de seus programas.
4. Entregue seu EP no Paca.
5. Não deixe de incluir em seu código um *relatório* para discutir seu EP: discuta as estruturas de dados usadas, os algoritmos usados, etc. *Se você escrever claramente como funciona seu EP, o monitor terá pouca dificuldade em corrigi-lo, e assim você terá uma nota mais alta.* (Se o monitor sofrer para entender seu código, sua nota será baixa.)

Observação final. Enviem dúvidas para a lista de discussão da disciplina.

Instituto de Matemática e Estatística, Universidade de São Paulo, Rua do Matão 1010, 05508–090 São Paulo, SP

Endereço eletrônico: `yoshi@ime.usp.br`