

Professor:	Ciro Cirne Trindade
Disciplina:	Linguagem de Programação
Curso:	Análise e Desenvolvimento de Sistemas
Grupos:	3 alunos
Data de entrega:	26/09/2014 (Manhã) – 01/10/2014 (Noite)

1º Trabalho Prático – Letras mais Frequentes

Neste problema estamos interessados na frequência das letras em uma determinada linha de texto. Especificamente, nós queremos saber a(s) letra(s) que ocorre(m) mais frequentemente, ignorando a caixa (maiúsculas e minúsculas devem ser consideradas iguais). Considere que as letras são as 26 letras do alfabeto sem acentuação.

Entrada

A entrada contém vários casos de teste. A primeira linha de um caso de teste contém um inteiro N indicando o número de casos de teste. Cada caso de teste contém uma única linha de texto. A linha pode conter caracteres que não sejam letras, mas é garantido que ela contém pelo menos uma letra e menos de 200 caracteres no total.

Saída

Para cada caso de teste, imprima uma linha contendo a(s) letra(s) que ocorre(m) mais frequentemente no texto em minúsculo (se houver empate, imprima todas as letras com a maior frequência em ordem alfabética).

Exemplo

Entrada	Saída
3	acfil
Facil	e
programar bem exige 10% de talento e 90% de suor	ae
Frequencia das letras	

Para resolver este problema você deve obrigatoriamente implementar um módulo contendo as funções definidas no arquivo *frequencia.h* mostrado a seguir. As funções deste módulo devem ser utilizadas na solução do problema. A função *main()* da sua aplicação deve ficar em um arquivo fonte que contenha apenas esta função.

```
/* frequencia.h */
```

```
#ifndef _FREQUENCIA_H  
#define _FREQUENCIA_H
```

```
#define COMPRIMENTO_MAXIMO 200 // comprimento máximo do texto  
#define LETRAS_ALFABETO 26 // número de letras do alfabeto
```

```
/* recebe uma string como 1º argumento e um caractere com 2º  
 * e devolve a frequência do caractere na string */  
int frequencia(char *, char);
```

```
/* recebe um vetor de inteiros como 1º argumento e o tamanho
 * deste vetor como 2º argumento e devolve o maior valor
 * contido no vetor. O vetor representa a frequência de
 * cada letra do alfabeto, sendo assim, o elemento da posição
 * 0 do vetor é a frequência da letra 'a', o da posição 2 a
 * frequência da letra 'b', ..., o da posição 25 a frequência
 * a letra 'z'. */
int maior_frequencia(int [], int);

/* recebe um vetor de inteiros como 1º argumento, o tamanho
 * deste vetor como 2º argumento e a frequência da(s) letra(s)
 * que ocorre(m) mais frequentemente no texto como 3º argumento.
 * O vetor representa a frequência de cada letra do alfabeto
 * no texto e a função imprime a(s) letra(s) de maior frequência.
 * Se houver empate, imprime as letras em ordem alfabética */
void imprime_mais_frequente(int [], int, int);

#endif
```

Informações importantes sobre Trabalho

1. Critérios de avaliação:
 - a) Corretude: 90%
 - b) Legibilidade: 10%
2. Todos os trabalhos devem possuir no cabeçalho a identificação dos autores.
3. Trabalhos copiados (com ou sem eventuais disfarces) terão a nota dividida pelo número de cópias (inclusive o original).
4. Trabalhos atrasados não serão aceitos.
5. Trabalhos com erros de sintaxe (ou seja, erros de compilação) receberão nota ZERO.
6. É muito importante que seu programa tenha comentários e esteja bem indentado, ou seja, digitado de maneira a ressaltar a estrutura de subordinação dos comandos do programa. A legibilidade do código será levada em consideração pelo critério de avaliação do trabalho.
7. Você deve enviar os fontes e o *makefile* da aplicação para o seguinte e-mail: ciroct@gmail.com
8. Guarde uma cópia do seu trabalho pelo menos até o final do semestre. Este material pode ser solicitado pelo professor em caso de extravio.