

Ortografia

Nome do arquivo fonte: `ortografia.c`, `ortografia.cpp` ou `ortografia.pas`

Um serviço de busca na Internet está preocupado com a crescente taxa de erros de ortografia de seus usuários, tornando mais difíceis as buscas por palavras chaves, que constantemente contêm erros de algumas letras, devidos a má digitação ou má ortografia.

O serviço funciona com base num dicionário de palavras. O usuário deve inserir uma palavra num campo de um formulário; o serviço então procura esta palavra no dicionário e retorna conteúdo que tenha relação com a palavra.

Para contornar o problema de ortografia, você foi contratado para fazer um programa que tenta adivinhar qual palavra o usuário pretendia procurar, independente de haver erros de ortografia nela.

Para este problema vamos definir a *distância* entre duas palavras A e B como sendo o número de operações, descritas abaixo, necessárias para transformar A em B :

1. Retirar uma letra de A .
2. Adicionar uma letra a A , em qualquer posição.
3. Trocar qualquer letra de A por outra letra, na mesma posição.

O serviço de busca definiu que a palavra P fornecida pelo usuário pode se referir a uma palavra D do dicionário se está a uma distância de no máximo 2 de D .

Exemplos:

- A palavra ‘tu’ pode se referir à palavra do dicionário ‘tubo’, realizando duas vezes a operação 2.
- A palavra ‘crto’ pode se referir à palavra do dicionário ‘corte’, realizando uma vez a operação 2 e uma vez a operação 3.
- A palavra ‘crto’ pode se referir à palavra do dicionário ‘curto’, realizando uma vez a operação 2.
- A palavra ‘hortgrafea’ não pode se referir à palavra do dicionário ‘ortografia’.

Você deve escrever um programa que, dado um dicionário de palavras, descubra para cada palavra fornecida pelo usuário a quais palavras do dicionário ela pode se referir, nas condições descritas acima.

Entrada

A entrada contém um único conjunto de testes, que deve ser lido do *dispositivo de entrada padrão* (normalmente o teclado). A primeira linha contém 2 inteiros N , M , representando respectivamente o número de palavras contidas no dicionário ($1 \leq N \leq 1000$) e o número de palavras a serem analisadas ($1 \leq M \leq 100$). Cada uma das N linhas seguintes conterá uma palavra pertencente ao dicionário. Cada uma das M linhas seguintes conterá uma palavra a ser analisada, fornecida pelo usuário. Cada palavra pode ter de 1 a 20 letras, contendo apenas letras de ‘a’ a ‘z’, minúsculas.

Saída

Seu programa deve imprimir, na *saída padrão*, M linhas, sendo uma linha para cada palavra fornecido pelo usuário. Cada linha deve conter todas palavras do dicionário às quais a palavra fornecida pode se referir. No caso de haver mais de uma palavra em uma linha da resposta, elas devem ser separadas por um espaço em branco, aparecendo na ordem que elas foram dadas na entrada, como pode ser visto no exemplo de saída abaixo. No caso de não haver nenhuma palavra em uma linha da resposta, deixe-a em branco.

Exemplo de entrada	Exemplo de saída
<pre>3 3 pato pateta caneca pat ccanecos pata</pre>	<pre>pato pato pateta</pre>

Informações sobre a pontuação

- Em um conjunto de casos de teste que totaliza 30 pontos, $N \leq 10$ e $M \leq 10$.
- Em um conjunto de casos de teste que totaliza 55 pontos, $N \leq 50$ e $M \leq 500$.