

## Práctica 1

### Problema 1. Reordenando letras en palabras

El párrafo siguiente es un extracto de un texto desconocido pero de fuente confiable (en inglés):

*“Accdrnig to a rscheearch at Cmabrigde Uinervtisy, it deosn’t mttar in waht order the ltteers in a wrod are, the olny iprmoetnt tihng is taht the frist and lsat ltteer be at the rghit pclae. The rset can be a toatl mses and you can sitll raed it wouthit porbelm. Tihs is bcuseae the huamn mnid deos not raed ervey lteter by istlef, but the wrod as a wlohe.”*

En este problema, se le proporcionará una serie de palabras y un texto. Las letras dentro de las palabras del texto estarán desordenadas, pero teniendo en cuenta que la primera y la última letra están en su posición original. Su tarea es ordenar el texto colocando las letras en sus posiciones originales con su versión correcta que se encuentra en el diccionario.

#### Entrada (reordenando.in)

La primera línea contendrá un entero  $N$  indicando el número de casos de prueba ( $1 \leq N$ ), seguidamente los  $N$  casos de prueba. Cada caso de prueba está compuesto de exactamente dos líneas: la primera línea contendrá la lista de palabras del diccionario en orden lexicográfico ascendente, la segunda línea contiene la lista de palabras a ser ordenadas. Las palabras en ambas listas están separadas por espacios en blanco y son palabras del alfabeto inglés en minúsculas. Usted puede asumir que cada lista no será vacía y tendrán a lo más 200 caracteres, y el diccionario no tendrá palabras duplicadas.

#### Salida (Pantalla)

Para cada caso de prueba, imprimir una línea con el texto ordenado: cada palabra  $w$  en el texto de entrada deberá ser reemplazada por una palabra  $d$  en el diccionario tal que  $d$  pueda ser convertida a  $w$  reordenando las letras a excepción de la primera y última letra. Si hay más de una palabra  $d$  en el diccionario que podría ser utilizada por  $w$ , reemplázela con la primera que se encuentra en el orden lexicográfico; si no hay palabras en el diccionario para  $w$ , entonces  $d=w$ .

#### Ejemplo de Entrada

2

```
make me programming rich will  
pagmrnmorig will mkae me rcih  
dreaming drinaemg yeah  
yaeh right keep drinaemg
```

Práctica 1

**Ejemplo de Salida**

```
programming will make me rich  
yeah right keep dreaming
```

**Problema 2: Triangulo de pascal**

Se usa para encontrar los coeficientes numéricos en una fórmula de binomios.

Entrada(teclado): Número de filas a generar

Salida(pantalla): Filas generadas

Ejemplo de entrada

7

Ejemplo de salida

```
1  
1 1  
1 2 1  
1 3 3 1  
1 4 6 4 1  
1 5 10 10 5 1  
1 6 15 20 15 6 1
```