

Preparando Recetas

Contribución de Guillermo García

Descripción del problema

Ricardo es un excelente cocinero. No hay receta que no sea capaz de preparar. ¡El problema es que a menudo olvida comprar los ingredientes!

Por esta razón, decidió preparar un listado de los ingredientes que tiene en la heladera. La heladera contiene en total H ingredientes. Además, organizando un baúl en su casa, encontró un libro que le regaló su abuela con P páginas, que tiene exactamente una receta en cada una de ellas. En particular, la i -ésima página, contiene un listado con los R_i ingredientes que necesita para llevar a cabo la receta.

Como tanto Ricardo como su abuela son algo desorganizados al redactar, **a veces hay ingredientes repetidos (de forma innecesaria)** en estos listados.

Debes ayudar a Ricardo escribiendo una función que determine cuántas recetas del libro que le regaló su abuela puede realizar con los ingredientes que tiene en la heladera, y además (para puntaje completo) en qué páginas del libro se encuentran dichas recetas.

Detalles de implementación

Debes implementar la función `recetas(heladera, libro, realizables)`, que dado un arreglo de strings `heladera` con los ingredientes presentes en la heladera de Ricardo, y un arreglo de arreglos de strings `libro` con todos los listados de ingredientes presentes en cada página del libro que le regaló su abuela; retorne la cantidad de recetas que puede realizar Ricardo con los ingredientes que tiene en la heladera.

Además, la función debe almacenar un listado con las páginas del libro de dichas recetas (sin repetir, en el orden de la entrada, comenzando a contar desde 1) en el parámetro `realizables`. Cualquier orden en el que se listen dichas páginas es considerado como válido.

Evaluador local

El evaluador local lee de la entrada estándar:

- Una primera línea con dos enteros, correspondientes a H y P : la cantidad de ingredientes en la heladera, y la cantidad de páginas en el libro de recetas, en ese orden.
- Una segunda línea con los H nombres de los ingredientes de la heladera, separados por espacios.
- P pares de líneas, uno por cada página del libro. En el i -ésimo par, de la primera línea lee un número R_i (que indica la cantidad de ingredientes en el listado de la i -ésima receta). A continuación, de la segunda línea del par lee los R_i nombres de los ingredientes del listado, separados por espacios.

Devuelve dos líneas por la salida estándar. El resultado de llamar a la función `recetas` en la primera línea, y el listado almacenado en el parámetro `realizables` en la segunda línea, con las palabras separadas por espacio.

Cotas

- $0 \leq H \leq 50.000$
- $1 \leq P \leq 50.000$
- $R_i \geq 1$ para todo $1 \leq i \leq P$
- Nunca se recibirán más de 100.000 ingredientes en total entre todos los listados de la entrada.
- Los nombres de los ingredientes tendrán a lo sumo 8 letras, y utilizarán únicamente letras minúsculas del alfabeto inglés (compuesto por 26 letras desde 'a' hasta 'z').

Ejemplo

Si el evaluador local recibe la siguiente entrada:

```
4 3
limon limon pan panceta
5
limon batata limon cordero batata
3
pan pan panceta
2
limon panceta
```

Ante una correcta implementación, el evaluador podría imprimir:

```
2
2 3
```

Porque con los ingredientes que tiene en su heladera, Ricardo puede cocinar la segunda y la tercera receta, pero no la primera.

Notar que en algunos casos puede haber ingredientes repetidos en la receta y/o en la heladera (pero el resultado deberá ser el mismo que si aparecieran una única vez).

Puntajes

Se recibe el 50% del puntaje por calcular correctamente el valor de retorno de la función (cantidad de recetas realizables por Ricardo), y el 50% restante si además el listado de páginas de libro indicado es correcto.

Subtareas

1. $H = 0$ (2 puntos)
2. $H = 1$, y $R_i = 1$ para todo $1 \leq i \leq P$ (6 puntos)
3. **La cantidad de ingredientes** entre todos los listados **será a lo sumo 2.000** (14 puntos)
4. $R_i = 1$ para todo $1 \leq i \leq P$ (22 puntos)
5. **Los ingredientes dentro de cada listado son todos distintos.** (44 puntos)
6. Sin más restricciones (12 puntos)