
Problema de demostración de PRO2**X83851_es**

Consideremos una secuencia de palabras donde junto a cada palabra aparece un entero que representa una categoría semántica de la palabra. La secuencia termina cuando se lea una palabra igual que la marca (un punto) seguida de un entero cualquiera; este último par no pertenece a la secuencia. Los enteros que representan las categorías poseen valores entre 1 y un cierto valor entero n , que también es un dato del programa. Dicho valor se lee al principio y lo supondremos pequeño. Encontraréis un ejemplo hacia la línea 70 de este fichero.

Puede ocurrir que haya categorías sin ninguna palabra y palabras que aparezcan asociadas a diferentes categorías a lo largo de la secuencia. El número de palabras distintas de la secuencia está acotado por la constante `@MAXNUMPAL=20@`, pero la longitud de la secuencia es desconocida, pues cada palabra puede aparecer varias veces. La longitud de cada palabra está acotada por la constante `@MAXLONG=20@`.

Después de la secuencia de pares (palabra, entero) se leerá una secuencia de enteros, que representan categorías entre 1 y n , acabada en una marca igual a cero y de longitud desconocida, pues cada categoría puede aparecer varias veces. Para cada aparición de cada categoría de esa secuencia, obtened y escribid la suma de las frecuencias de las palabras que han aparecido alguna vez asociadas a dicha categoría en la primera secuencia. Para cada palabra se ha de considerar el número total de veces que ha aparecido en la primera secuencia, independientemente de la categoría que la haya acompañado en cada aparición.

Para resolver el problema se debe producir una nueva especificación y una nueva implementación de la clase `ListaPalabras`, que realice las tareas pedidas de la manera más sencilla y eficiente posible. En particular, no se debe guardar información redundante.

No se debe definir ninguna clase nueva, pero será necesario modificar las cabeceras de alguna operaciones e incluso los campos de la clase original. Habrá que entregar el programa principal, llamado `pro2.cpp`, y los archivos `ListaPalabras.hpp` (para el que os proporcionamos una plantilla, con algunos cambios ya aplicados) y `ListaPalabras.cpp`. Los nombres son *obligatorios*.

No hay que generar ninguna documentación con el doxygen.

Requisitos adicionales:

- el main debe realizar los siguientes pasos:
 1. leer `@n@`;
 2. inicializar la lista de palabras (usando un constructor de la clase);
 3. leer la secuencia de pares (palabra, entero) actualizando la información de la lista de palabras;
 4. leer la secuencia de categorías y, para cada categoría leída,
 - (a) obtener la suma total de frecuencias de las palabras asociadas alguna vez a esa categoría;
 - (b) escribir dicha suma.

- El paso 3. se ha de hacer en un bucle directamente en el @main@ (tratamiento de una secuencia) que llame a las operaciones necesarias de las clases @Palabra@ y @ListaPalabras@.
- El paso 4. se ha de hacer en otro bucle en el @main@ que llame a una *nueva* operación de la clase @ListaPalabras@ para el paso 4.1. Todos los cout del programa han de aparecer en el main, nunca en las operaciones.
- Para construir la nueva clase @ListaPalabras@ no se pueden usar pilas ni colas, ni cualquier otra estructura que no haya aparecido en las sesiones 1–5 de laboratorio.

Una mica de codi:

```
for (int i=0; i<n; ++i)
```

Entrada

Falta escriure.

Salida

Falta escriure.

Ejemplo de entrada 1

```
10
saxo 7
libertad 3
roca 1
gato 2
saxo 1
coche 7
vaso 6
fuerza 3
coche 7
coche 8
saxo 10
gato 4
saxo 10
. 1
7
1
5
0
```

Ejemplo de salida 1

```
cat 7: suma frecuencias 7
cat 1: suma frecuencias 5
cat 5: suma frecuencias 0
```

Observación

Aquest problema requereix el compilador PRO2. Hi ha alguns fitxers públics disponibles.

Información del problema

Autoría: Borja Valles

Generación: 2026-01-27T18:56:33.092Z

© Jutge.org, 2006–2026.

<https://jutge.org>