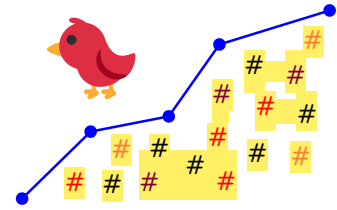


Tridente de temas candentes

Corría el año 2006 cuando el joven Jack Dorsey inventó, no se sabe si en un garaje, en un observatorio ornitológico o en la consulta del otorrino-laringólogo, un servicio de mensajes cortos públicos de no más de 140 caracteres. Reducto de un puñado de internautas curiosos durante sus primeros años, ahora es raro encontrar a alguien que no conozca esta red social. Unos cuantos de esos 140 caracteres se pueden emplear en citar a otros usuarios escribiendo @nombre o hacer referencia a temas con #etiqueta donde etiqueta es una cadena de texto sin espacios. La plataforma ofrece en directo una lista de los temas más candentes, aquellos que han sido citados más veces en los últimos minutos. Sin embargo, recientemente, un extravagante multimillonario ha tirado de chequera para comprar la red social, cambiarla de nombre a X (no se sabe si porque hizo su fortuna gracias a las quinielas o por otro tipo de evocaciones) y manejarla a su antojo. Entre otros cambios, ha empezado a cobrar por todo, incluso por el servicio que ofrecía los tres temas más citados.



Con el propósito de seguir enterado de las tendencias sin contribuir a engrosar la ya abultada faltriquera del magnate, estás elaborando un sistema para calcular los temas más candentes por ti mismo. Has encontrado un servicio gratuito que proporciona periódicamente actualizaciones del número de citas de cada tema y con ella quieres calcular el podio de los tres temas más citados en los últimos 15 minutos. Estas actualizaciones pueden ser C tema n cuando el tema *tema* ha recibido n citas desde la última actualización o E tema n cuando han expirado n citas del tema *tema* de las comunicadas anteriormente por haber pasado los 15 minutos. Cuando se reciba el comando TC se desea obtener el podio de las tres palabras más citadas en la última franja. En caso de empate, queremos priorizar las tendencias más recientes, así que quedarán primero los temas cuyo último evento C haya llegado más tarde. Los temas sin ninguna cita no aparecerán en ningún caso.

Entrada

La entrada está formada por diversos casos de prueba. Para cada caso, primero aparece en una línea el número n de eventos (como mucho 1.000.000) y a continuación aparecen n líneas que describen cada uno de estos eventos. La descripción de un evento comienza por una palabra que indica su tipo (TC, C o E), que en el caso de los eventos C y E el tipo viene seguido de dos argumentos: el tema citado (una palabra) y el número de citas que aumentan o disminuyen (un entero positivo).

La entrada termina con el último caso.

Salida

Para cada consulta TC se escribirá el podio de temas más citados de mayor a menor. Cada entrada del podio se escribirá en una línea formada por el número de orden (1, 2 o 3), un espacio y el tema correspondiente. Si hubieran aparecido menos de tres temas, se escribirán solo los primeros puestos (que podrían ser ninguno). Cada caso de prueba terminará con tres guiones ---.

Les vamos asignando prioridad. Tenemos un entero que sea prio
=1000000

Entrada de ejemplo

```
9
C quickSort 31
C mergeSort 40
C burbuja 17
C quickSort 8
TC
C heapSort 19
C burbuja 1
E mergeSort 6
TC
5
C Hulk 1000
C Ironman 3000
E Hulk 1000
C Superman 3000
TC
```

Salida de ejemplo

```
1 mergeSort
2 quickSort
3 burbuja
1 quickSort
2 mergeSort
3 heapSort
---
1 Superman
2 Ironman
---
```