

Tieni aggiornato il catalogo (catalogo)

Devi tenere aggiornato il catalogo di una libreria.

In particolare le azioni della libreria saranno quelle di aggiungere e togliere una copia di un libro con un certo *id* e di contare quante copie di un libro con un certo *id* sono presenti.

Implementazione

Dovrai sottoporre un unico file, con estensione `.c` o `.cpp`.

📖 Tra gli allegati a questo task troverai un template `catalogo.c` o `catalogo.cpp` con un esempio di implementazione.

Dovrai implementare le seguenti funzioni:

```
C/C++ | void aggiungi(long long int id);
```

- L'intero *id* rappresenta l'id del libro di cui è stata aggiunta una copia.

```
C/C++ | void toglì(long long int id);
```

- L'intero *id* rappresenta l'identificativo del libro di cui è stata tolta una copia.

```
C/C++ | int conta(long long int id);
```

- L'intero *id* rappresenta l'identificativo del libro di cui si vuole conoscere il numero di copie.
- La funzione deve restituire il numero di copie presenti del libro con identificativo *id*.

Grader di prova

Nella directory relativa a questo problema è presente una versione semplificata del grader usato durante la correzione, che potete usare per testare le vostre soluzioni in locale. Il grader di esempio legge i dati da `stdin`, chiama le funzioni che dovete implementare e scrive su `stdout`, secondo il seguente formato.

Il file di input è composto da $Q + 1$ righe, dove Q rappresenta il numero totale di chiamate alle funzioni `aggiungi`, `togli` e `conta`, contenenti:

- Riga 1: l'unico intero Q .
- Righe $2 \dots Q + 1$: la descrizione di un'operazione, che può quindi essere:
 - a *id*: aggiungere una copia del libro *id*;
 - t *id*: togliere una copia del libro *id*;
 - c *id*: viene richiesto il numero di copie del libro *id*.

Il file di output è composto da C righe, dove C è il numero di chiamate alla funzione `conta`:

- Riga i : il valore restituito dall' i -esima chiamata alla funzione `conta`.

Assunzioni

- $1 \leq Q \leq 200\,000$.
- $0 \leq id \leq 10^{18}$ per ogni chiamata alle funzioni `aggiungi`, `togli` e `conta`.
- Inizialmente il catalogo è vuoto.
- È garantito che ad ogni chiamata alla funzione `togli`, esiste almeno una copia del libro.

Assegnazione del punteggio

Il tuo programma verrà verificato su diversi test case raggruppati in un unico subtask. Per ottenere il punteggio relativo al problema, è necessario risolvere correttamente tutti i test che compongono il subtask.

- **Subtask 1 [0 punti]:** Casi d'esempio.
- **Subtask 2 [10 punti]:** $id \leq 1\,000$ per ogni chiamata.
- **Subtask 3 [40 punti]:** $Q \leq 1\,000$.
- **Subtask 4 [50 punti]:** Nessuna limitazione aggiuntiva.

Esempi di input/output

stdin	stdout
5 a 5 c 7 a 10 a 5 c 5	0 2
11 a 11 a 11 a 10 a 10 c 13 c 10 a 10 a 10 t 11 c 11 c 10	0 2 1 4

Spiegazione

Nel **primo caso di esempio**, al primo conteggio il catalogo è il seguente:



Nel secondo conteggio il catalogo è il seguente:



Nel **secondo caso di esempio**, nel primo e secondo conteggio il catalogo è il seguente:



Nel terzo e quarto conteggio il catalogo è il seguente:

