

Sushi variegato (yutaka)

Yuta Takaya deve servire il sushi ai suoi commensali. In questo momento ha un lungo tagliere che contiene, disposti in fila, N pezzi di sushi. Esistono differenti varietà di sushi: l' i -esimo pezzo è di tipo $V[i]$.

Yuta ha intenzione di apportare al tagliere alcuni tagli con la sua katana (eventualmente 0, eventualmente $N - 1$) tra pezzi di sushi adiacenti, al fine di ottenere un certo numero di “taglierini” da portare in tavola: ogni taglierino conterrà un certo numero (non nullo) di pezzi di sushi adiacenti. Yuta ha deciso che, affinché gli ospiti non si annoino, nessun taglierino dovrà contenere due pezzi di sushi dello stesso tipo.

In quanti modi diversi Yuta Takaya può partizionare il tagliere? Poiché questo numero può essere molto grande, si chiede di restituirlo modulo $10^9 + 7$ (1 000 000 007).

Implementazione

Dovrai sottoporre esattamente un file con estensione `.cpp` o `.c`.

📖 Tra gli allegati a questo task troverai un template (`yutaka.cpp` e `yutaka.c`) con un esempio di implementazione.

Dovrai implementare la seguente funzione:

■ Funzione `taglia`

C/C++ | `int taglia(int N, int V[]);`

- L'intero N rappresenta il numero di pezzi di sushi.
- L'array $V[]$, indicizzato da 0 a $N - 1$, contiene i tipi di sushi. L' i -esimo pezzo è di tipo $V[i]$.
- La funzione dovrà restituire un unico intero: il numero di modi in cui Yuta Takaya può effettuare i tagli, modulo $10^9 + 7$.

Nella directory relativa a questo problema è presente una versione semplificata del grader usato durante la valutazione, che potete usare per testare le vostre soluzioni in locale. Il grader di esempio legge i dati di input da `stdin`, chiama la funzione `taglia()` e scrive il suo valore di ritorno su `stdout`.

Nel caso vogliate generare un file di input, `stdin` deve avere questo formato:

- Riga 1: contiene l'intero N .
- Riga 2: contiene gli interi $V[0]$, $V[1]$, \dots , $V[N - 1]$, separati da uno spazio.

`stdout` avrà il seguente formato:

- Riga 1: contiene l'intero ritornato da `taglia()`.

Assunzioni

- $1 \leq N \leq 1\,000\,000$
- $1 \leq v[i] \leq 1\,000\,000$ per $i = 0, 1, \dots, N - 1$

Assegnazione del punteggio

Il tuo programma verrà testato su diversi test case raggruppati in subtask. Per ottenere il punteggio relativo ad un subtask, è necessario risolvere correttamente tutti i test relativi ad esso.

- **Subtask 1 [0 punti]**: Casi d'esempio.
- **Subtask 2 [17 punti]**: $N \leq 15$.
- **Subtask 3 [10 punti]**: $N \leq 2\,000$ e, per ogni tipo di sushi, vi sono al più due pezzi di quel tipo.
- **Subtask 4 [22 punti]**: $N \leq 2\,000$.
- **Subtask 5 [15 punti]**: $N \leq 100\,000$.
- **Subtask 6 [36 punti]**: Nessuna limitazione specifica.

Esempi di input/output

5 3 1 1 2 2	4
7 1 1 1 1 1 1 1	1

Spiegazione

Nel **primo caso di esempio** vi sono 4 insiemi di tagli che soddisfano le condizioni:

3	1	1	2	2
---	---	---	---	---

3	1	1	2	2
---	---	---	---	---

3	1	1	2	2
---	---	---	---	---

3	1	1	2	2
---	---	---	---	---

Nel **secondo caso di esempio** tutti i pezzi di sushi sono dello stesso tipo, ergo vi è un solo modo di effettuare i tagli in modo tale che ogni taglierino contenga pezzi di tipo diverso.