Zająknięcia

XXIII Olimpiada Informatyczna (2015/2016), etap II
Autor zadania: Tomasz Syposz
Opis rozwiązania: Jakub Radoszewski
Limit pamięci: 32 MB
https://oi.edu.pl/pl/archive/oi/23/zaj

Bitek zapadł ostatnio na dziwną chorobę: strasznie się jąka, a przy tym jedyne słowa, które wypowiada, to liczby. Jego starszy brat, Bajtek, zauważył jednak dziwną powtarzalność w zająknięciach Bitka. Podejrzewa, że Bitek tak naprawdę udaje, żeby nie chodzić do szkoły i móc więcej grać na komputerze. Bajtek nie może przez to uczyć się programowania i jest z tego powodu bardzo smutny. Postanowił więc zdemaskować młodszego brata i liczy, że w nagrodę będzie miał tyle czasu na programowanie, ile dusza zapragnie.

Opiszmy formalnie podejrzenia Bajtka. Załóżmy, że mamy dany ciąg liczb A.

- *Podciągiem* ciągu A nazywamy ciąg powstały przez wyrzucenie z A dowolnych wyrazów, np. $1, 1, 7, 5$ jest podciągiem ciągu $1, 3, 1, 7, 6, 6, 5, 5$, natomiast $1, 5, 7$ nim nie jest.
- *Zająknięciem* ciągu A nazywamy podciąg A, który składa się z ustawionych po kolei par takich samych wyrazów, np. $1, 1, 1, 1, 3, 3$ oraz $2, 2$ są zająknięciami ciągu $1, 2, 1, 2, 1, 2, 1, 3, 3$, natomiast $1, 2, 1, 2$ oraz $2, 2, 2$ nimi nie są.

  Formalnie zająknięcie jest podciągiem postaci $a_1, a_2, a_3, \ldots, a_{2k-1}, a_{2k}$, gdzie $a_{2i-1} = a_{2i}$ dla $1 \leqslant i \leqslant k$.

Mając dane dwie wypowiedzi Bitka jako ciągi liczb, pomóż Bajtkowi stwierdzić, jaka jest długość najdłuższego zająknięcia, które występuje w każdym z tych ciągów, a nagroda cię nie ominie.

Wejście

Pierwszy wiersz wejścia zawiera dwie liczby całkowite n oraz m ($n, m \geqslant 2$) oznaczające długości ciągów A i B, które reprezentują wypowiedzi Bitka. W drugim wierszu wejścia znajduje się n liczb całkowitych $a_1, a_2, \ldots, a_n$, czyli kolejne wyrazy ciągu A ($1 \leqslant a_i \leqslant 10^9$). W trzecim wierszu wejścia znajduje się m liczb całkowitych $b_1, b_2, \ldots, b_m$, czyli kolejne wyrazy ciągu B ($1 \leqslant b_i \leqslant 10^9$).

Wyjście

Twój program powinien wypisać na wyjście jedną nieujemną liczbę całkowitą oznaczającą długość najdłuższego wspólnego zająknięcia ciągów A i B. Jeśli ciągi nie mają żadnego wspólnego zająknięcia, poprawnym wynikiem jest 0.

Przykład

Dla danych wejściowych:	poprawnym wynikiem jest:

`7 9`
`1 2 2 3 1 1 1`
`2 4 2 3 1 2 4 1 1`
	`4`

Wyjaśnienie przykładu: Szukanym ciągiem jest 2, 2, 1, 1.

Ocenianie

Zestaw testów dzieli się na podzadania spełniające poniższe warunki. Testy do każdego podzadania składają się z jednej lub większej liczby osobnych grup testów.

Podzadanie	Warunki	Liczba punktów
1	$n, m \leqslant 2000$	30
2	$n, m \leqslant 15\,000$ i każda liczba w każdym ciągu występuje co najwyżej dwa razy	28
3	$n, m \leqslant 15\,000$	42

Rozwiązanie

W zadaniu dane są dwa ciągi liczb $A = (a_1, \ldots, a_n)$ i $B = (b_1, \ldots, b_m)$. W opisie rozwiązania zamiast o wartościach elementów ciągów wygodniej nam będzie mówić o ich kolorach, tak więc np. a_i będzie oznaczać kolor i-tego elementu ciągu A. Naszym celem jest znaleźć najdłuższe wspólne zająknięcie ciągów A i B, czyli najdłuższy ciąg kolorów złożony z parami powtarzających się elementów, który jest podciągiem każdego z ciągów A i B. Co istotne, w odpowiedzi wystarczy podać długość najdłuższego wspólnego zająknięcia.

Najdłuższy wspólny podciąg

Nasze zadanie ewidentnie ma związek z problemem znajdowania najdłuższego wspólnego podciągu dwóch ciągów. Rozważania zacznijmy więc od przypomnienia klasycznego rozwiązania tego problemu za pomocą programowania dynamicznego (patrz np. książka *Wprowadzenie do algorytmów* [6]). Wyznacza się w nim dwuwymiarową tablicę NWP rozmiaru $(n+1) \times (m+1)$, taką że $NWP[i, j]$ oznacza długość najdłuższego wspólnego podciągu ciągów $a_1, \ldots, a_i$ oraz $b_1, \ldots, b_j$. Szukanym wynikiem jest oczywiście $NWP[n, m]$. Mamy następującą zależność rekurencyjną:

$$NWP[i,j] = \begin{cases} 0 & \text{jeżeli } i = 0 \text{ lub } j = 0, \\ NWP[i-1, j-1] + 1 & \text{jeżeli } a_i = b_j, \\ \max(NWP[i-1, j],\ NWP[i, j-1]) & \text{w przeciwnym przypadku.} \end{cases}$$