Zadanie: TES
Test na inteligencję

XVII OI, etap I. Plik źródłowy `tes.*` Dostępna pamięć: 64 MB. *19.10–16.11.2009*

Jedno z zadań w bajtockim teście na inteligencję polega na wykreślaniu liczb z zadanego początkowego ciągu tak, aby otrzymywać w ten sposób różne inne zadane sekwencje. Bajtazar chciałby zostać bajtockim mistrzem IQ, ale wyjątkowo kiepsko radzi sobie z zadaniami tego typu. Zamierza dużo ćwiczyć i poprosił Cię o napisanie programu, który pomoże mu szybko sprawdzać odpowiedzi.

Wejście

Pierwszy wiersz standardowego wejścia zawiera jedną liczbę całkowitą m ($1 \leqslant m \leqslant 1\,000\,000$). Drugi wiersz zawiera ciąg m liczb całkowitych $a_1, a_2, \ldots, a_m$ ($1 \leqslant a_i \leqslant 1\,000\,000$ dla $1 \leqslant i \leqslant m$) pooddzielanych pojedynczymi odstępami, tworzących początkowy ciąg w zadaniu z testu. Trzeci wiersz wejścia zawiera jedną dodatnią liczbę całkowitą n. Kolejne $2n$ wierszy zawiera opisy ciągów, które mają powstać w wyniku wykreślania różnych liczb z początkowego ciągu. Opis każdego z tych ciągów zajmuje po dwa kolejne wiersze. W pierwszym wierszu każdego opisu znajduje się liczba całkowita m_i ($1 \leqslant m_i \leqslant 1\,000\,000$). Drugi wiersz zawiera m_i-elementowy ciąg liczb całkowitych $b_{i,1}, b_{i,2}, \ldots, b_{i,m_i}$ ($1 \leqslant b_{i,j} \leqslant 1\,000\,000$ dla $1 \leqslant j \leqslant m_i$) pooddzielanych pojedynczymi odstępami. Możesz założyć, że suma długości podanych n sekwencji nie przekroczy $1\,000\,000$.

Wyjście

Twój program powinien wypisać na standardowe wyjście n wierszy. Wiersz o numerze i (dla $1 \leqslant i \leqslant n$) powinien zawierać jedno słowo „`TAK`” lub „`NIE`” (bez cudzysłowów), w zależności od tego, czy i-ta sekwencja z wejścia może powstać w wyniku wykreślenia (tj. usunięcia) pewnych (niekoniecznie kolejnych) liczb z początkowego ciągu. Oczywiście, kolejność liczb w powstałym po wykreśleniach ciągu ma znaczenie (patrz przykład).

Przykład

Dla danych wejściowych:

```
7
1 5 4 5 7 8 6
4
5
1 5 5 8 6
3
2 2 2
3
5 7 8
4
1 5 7 4
```

poprawnym wynikiem jest:

```
TAK
NIE
TAK
NIE
```