

# Wycieczki

Letni obóz treningowy OIJ, dzień 3.  
20 sierpnia 2020

Kod zadania: **wyc**  
Limit czasu: **10 s**  
Limit pamięci: **512 MB**



Bajtosią prowadzi biuro podróży. Nie jest to łatwy biznes, szczególnie w dzisiejszych czasach, dlatego trzeba wprowadzać nowe akcje i promocje. Bajtosią zdecydowała się zorganizować serię  $N$  jednodniowych wycieczek, po jednej na każdy z  $N$  dni wakacji. Przygotowana wycieczka na  $i$ -ty dzień ma koszt  $A_i$  (dla  $i = 1, 2, \dots, N$ ).

Bajtosią zauważyła, że wszyscy klienci mają bardzo podobne potrzeby. Wszyscy klienci decydują się na kupno dokładnie jednej wycieczki. Każdy klient ma pewien przedział czasu, kiedy jest na urlopie i chciałby kupić wycieczkę pomiędzy pewnymi dniami  $L_j$  a  $R_j$  (włącznie). Każdy klient ma także bon turystyczny o pewnym koszcie  $V_j$ , który pozwala mu pokryć koszt tej wycieczki. Aby bon można było wykorzystać w całości (i nic się nie zmarnowało), klient chciałby kupić wycieczkę która jest warta **więcej** niż  $V_j$ .

Bajtosią także podzieliła swoich klientów na dwie kategorie:

- Klientów, którzy chcą wybrać się na wakacje **jak najszybciej**. Oznacza to, że wykupią oni pierwszą wycieczkę, która będzie dostępna podczas ich urlopu i kosztowała więcej niż wartość ich bonu.
- Klientów, którzy chcą wyjechać na wakacje **jak najtaniej**. Oznacza to, że wykupią oni najtańszą wycieczkę, która jest dostępna podczas ich urlopu, o ile będzie kosztowała więcej niż wartość ich bonu. W przypadku kilku wycieczek spełniających to kryterium, klienci zawsze wybierają tę wycieczkę, która będzie najszybciej.

Bajtosią teraz chciałaby przyspieszyć obsługę klientów i stworzyć system, który pomoże obsługiwać zapytania. Dodatkowo, czasami koszty wycieczek się zmieniają (z przyczyn niezależnych od Bajtosi) i jej system musi obsługiwać także zmiany kosztów wycieczek.

Napisz program, który wczyta początkowe ceny wycieczek, zapytania klientów oraz zmiany cen, obliczy najlepszy dzień na wycieczkę dla każdego klienta i wypisze wyniki na standardowe wyjście.

## Wejście

W pierwszym wierszu wejścia znajdują się dwie liczby naturalne  $N$  oraz  $Q$  ( $1 \leq N, Q \leq 200\,000$ ), określające kolejno: liczbę wycieczek będącą jednocześnie liczbą dni wakacji oraz liczbę zapytań klientów wraz ze zmianami cen. W drugim wierszu wejścia znajduje się ciąg  $N$  liczb naturalnych  $A_i$  ( $0 \leq A_i \leq 10^9$ ), gdzie  $A_i$  oznacza początkową cenę wycieczki zaplanowanej na  $i$ -ty dzień.

W kolejnych  $Q$  wierszach znajdują się kolejne zdarzenia.

- Jeżeli chcemy obsłużyć klienta, który chce wybrać się na wakacje jak najszybciej, na początku wiersza znajdzie się słowo **najszybciej**, a po nim trzy liczby całkowite  $L_j$ ,  $R_j$  oraz  $V_j$  ( $1 \leq L \leq R_j \leq N$ ,  $0 \leq V_j \leq 10^9$ ) oznaczające kolejno pierwszy i ostatni dzień urlopu danego klienta oraz wartość jego bonu.
- Jeżeli chcemy obsłużyć klienta, który chce wybrać się na wakacje jak najtaniej, na początku wiersza znajdzie się słowo **najtaniej**, a po nim trzy liczby całkowite  $L_j$ ,  $R_j$  oraz  $V_j$  ze znaczeniem oraz ograniczeniami jak wyżej.
- Jeżeli cenę którejś wycieczki należy zmodyfikować, na początku wiersza znajdzie się słowo **zmiana**, a po nim dwie liczby całkowite  $D_j$  oraz  $C_j$  ( $1 \leq D_j \leq N$ ,  $0 \leq C_j \leq 10^9$ ), które oznaczają, że cenę wycieczki dnia  $D_j$  należy zmienić na  $C_j$ .

## Wyjście

Twój program powinien wypisać odpowiedzi dla zdarzeń typu **najszybciej** oraz **najtaniej** zgodnie z kolejnością ich występowania na wejściu w osobnym wierszach.

Jeżeli nie istnieje żadna wycieczka spełniająca warunki klienta, należy zamiast tego wypisać **NIE**.



## Ocenianie

Możesz rozwiązać zadanie w kilku prostszych wariantach – niektóre grupy testów spełniają pewne dodatkowe ograniczenia. Poniższa tabela pokazuje, ile punktów otrzyma Twój program, jeśli przejdzie testy z takim ograniczeniem.

Dodatkowe ograniczenia	Liczba punktów
nie ma żadnych zmian	30
nie ma klientów, którzy chcą się wybrać na wakacje jak najszybciej	35
nie ma klientów, którzy chcą się wybrać na wakacje jak najtaniej	35
$N, Q \leq 40\,000$	67

## Przykłady

Wejście dla testu wyc0a:

```
6 5
3 2 4 2 9 1
najtaniej 2 5 3
najszybciej 3 4 3
najtaniej 1 6 9
zmiana 4 10
najtaniej 1 6 9
```

Wyjście dla testu wyc0a:

```
3
3
NIE
4
```

**Wyjaśnienie do przykładu:** Początkowo ceny wycieczek w kolejnych dniach to  $[3, 2, 4, 2, 9, 1]$ , mamy  $Q = 6$  zdarzeń.

- Szukamy najtańszej wycieczki o koszcie większym niż 3 wśród wycieczek między drugim a piątym dniem. Najtańszą taką wycieczką jest wycieczka w trzecim dniu o koszcie 4. Wypisujemy numer dnia – 3.
- Szukamy pierwszej wycieczki o koszcie większym niż 3 pomiędzy drugim a czwartym dniem. Jest to ponownie wycieczka w trzecim dniu.
- Pomiedzy pierwszym a szóstym dniem nie mamy żadnej wycieczki droższej niż 9, więc należy wypisać NIE.
- Zmieniamy cenę wycieczki czwartego dnia na 10. Teraz ceny wycieczek w kolejnych dniach to  $[3, 2, 4, 10, 9, 1]$ .
- Pomiedzy pierwszym a szóstym dniem mamy już wycieczkę droższą niż 9 (jest to wycieczka zmodyfikowana w poprzednim zdarzeniu). Wypisujemy zatem 4.

Wejście dla testu wyc0b:

```
4 6
7 3 1 2
najtaniej 1 2 0
najtaniej 2 3 0
najtaniej 3 4 0
najszybciej 1 2 0
najszybciej 2 3 0
najszybciej 3 4 0
```

Wyjście dla testu wyc0b:

```
2
3
3
1
2
3
```

## Pozostałe testy przykładowe

- test wyc0c:  $N = 20$ ,  $Q = 100$ , nie ma klientów szukających wycieczki jak najszybciej.
- test wyc0d:  $N = 20$ ,  $Q = 100$ , nie ma klientów szukających wycieczki jak najtaniej.
- test wyc0e:  $N = Q = 40000$ , test losowy.