

Zadanie Dziwny Generator

Oryginał zadania pochodzi z Codeforces. [\[link\]](#)

Treść

Bajtazar otrzymał od bitysława tajemniczy przedmiot — generator permutacji losowych. Zafascynowany jego działaniem Bajtazar, postanowił dokładnie przeanalizować algorytm, według którego urządzenie tworzy permutację n elementów.

Proces składa się z n kroków. W i -tym kroku generator wybiera pozycję w permutacji dla liczby i . Pozycja ta jest wyznaczana w następujący sposób:

1. Dla każdego indeksu $j \in \{1, \dots, n\}$ wyznaczana jest wartość r_j — najmniejszy indeks taki, że $j \leq r_j \leq n$ oraz pozycja r_j nie jest jeszcze zajęta. Jeśli taka pozycja nie istnieje, r_j jest niezdefiniowane.
2. Dla każdego indeksu $t \in \{1, \dots, n\}$ wyznaczana jest wartość $count_t$ — liczba takich indeksów j , dla których $r_j = t$.
3. Generator rozważa tylko te pozycje, które są jeszcze wolne. Spośród nich wybiera taką pozycję t , dla której wartość $count_t$ jest maksymalna.
4. Jeśli istnieje więcej niż jedna pozycja z maksymalną wartością $count_t$, generator może wybrać **dowolną z nich**.

Rozważmy działanie generatora na przykładzie z $n = 5$ i z początkowym rozmieszczeniem

$[2, 3, _, _, 1]$.

Wyznaczając pozycję dla liczby 4 generator:

- wyznacza wartości r ($r_1 = 3, r_2 = 3, r_3 = 3, r_4 = 4, r_5$ jest niezdefiniowane),
- wyznacza $count$ ($count_1 = 0, count_2 = 0, count_3 = 3, count_4 = 1, count_5 = 0$),
- Wolne pozycje to 3 i 4. Wartości $count$ dla nich to odpowiednio 3 oraz 1.
- Maksymalna wartość $count$ wynosi 3 i jest osiągnięta dla pozycji 3. Zatem liczba 4 musi zostać umieszczona na pozycji 3.

Bajtazar wygenerował wiele permutacji, a następnie zapisał jedną, która przysłała mu do głowy: p_1, p_2, \dots, p_n . Chciałby wiedzieć, czy taka permutacja mogła zostać wytworzona przez opisany wyżej generator. Pomóż mu to rozstrzygnąć.

Wejście

Pierwszy wiersz wejścia zawiera jedną liczbę całkowitą t ($1 \leq t \leq 10^5$), oznaczającą liczbę zestawów danych. W kolejnych wierszach znajdują się opisy poszczególnych zestawów.

Pierwszy wiersz każdego zestawu danych zawiera jedną liczbę całkowitą n ($1 \leq n \leq 10^5$) — długość permutacji. Drugi wiersz zestawu zawiera n różnych liczb całkowitych p_1, p_2, \dots, p_n ($1 \leq p_i \leq n$), tworzących permutację, którą zapisał Denis.

Suma wartości n dla wszystkich zestawów danych nie przekracza 10^5 .

Zadanie Dziwny Generator

Oryginał zadania pochodzi z Codeforces. [\[link\]](#)

Wyjście

Dla każdego zestawu danych wypisz w oddzielnym wierszu słowo TAK, jeżeli podana permutacja mogła zostać wytworzona przez generator, lub NIE w przeciwnym przypadku.

Przykłady

Wejście (przykład 1):

```
5
5
2 3 4 5 1
1
1
3
1 3 2
4
4 2 3 1
5
1 5 2 4 3
```

Wyjście (przykład 1):

```
TAK
TAK
NIE
TAK
NIE
```

Wyjaśnienie: Przeanalizujemy działanie generatora dla pierwszego zestawu danych:

- Dla liczby 1: Wartości $r = [1, 2, 3, 4, 5]$, a tablica liczników $count = [1, 1, 1, 1, 1]$. Wartość maksymalna (1) występuje na każdej wolnej pozycji. Generator może wybrać dowolną z nich. Zgodnie z przykładem, wybiera pozycję 5.
- Dla liczby 2: Wartości $r = [1, 2, 3, 4, \times]$, tablica $count = [1, 1, 1, 1, 0]$. Wartość maksymalna (1) występuje na pozycjach 1, 2, 3 i 4. Generator może wybrać dowolną z nich. Wybiera pozycję 1.
- Dla liczby 3: Wartości $r = [2, 2, 3, 4, \times]$, tablica $count = [0, 2, 1, 1, 0]$. Wartość maksymalna (2) jest osiągnięta **wyłącznie** na pozycji 2. Generator musi wybrać pozycję 2.
- Dla liczby 4: Wartości $r = [3, 3, 3, 4, \times]$, tablica $count = [0, 0, 3, 1, 0]$. Wartość maksymalna (3) jest osiągnięta **wyłącznie** na pozycji 3. Generator musi wybrać pozycję 3.
- Dla liczby 5: Wartości $r = [4, 4, 4, 4, \times]$, tablica $count = [0, 0, 0, 4, 0]$. Wartość maksymalna (4) jest osiągnięta na jedynej wolnej pozycji 4. Generator wybiera pozycję 4.

Ostatecznie otrzymaliśmy permutację '2 3 4 5 1', co oznacza, że mogła ona zostać stworzona przez generator, więc odpowiedź to TAK