

Dysk twardy (dysk-twardy)

Memory limit: 32 MB

Time limit: 1.00 s

Producent bajtockich dysków twardych BajtoDisk wypuszcza właśnie na rynek nową serię dysków twardych HugeDisk o pojemnościach dochodzących do kilkuset bajtobajtów. Nowością w tych dyskach ma być specjalne oprogramowanie, które walczy ze zjawiskiem fragmentacji danych – czyli sytuacją, w której jeden plik zapisany jest w wielu częściach na dysku twardym.

Twoim zadaniem jako pracownika BajtoDisku, jest implementacja oprogramowania, które zajmuje kolejne sektory dysku. Dokładniej – należy zaimplementować funkcję, która zajmie jeden wybrany (podany= jako argument) sektor dysku oraz wyznaczy numer następnego wolnego sektora.

Napisz program, który wczyta z wejścia operacje zajęcia sektora, których należy dokonać na dysku, dla każdej operacji wyznaczy numer następnego wolnego sektora po aktualnie zajęтым oraz wypisze wyniki na standardowe wyjście.

Wejście

W pierwszym wierszu wejścia znajduje się jedna liczba naturalna N , określająca liczbę sektorów na dysku. W kolejnych N wierszach znajdują się kolejne operacje dyskowe – aż do wypełnienia całego dysku. W $i + 1$ -szym wierszu znajduje się i -ta operacja. Opis każdej operacji składa się z jednej liczby naturalnej A_i , $1 \leq A_i \leq N$, określającej numer zajmowanego sektora w i -tej operacji. Gwarantowane jest, że wszystkie liczby A_i są parami różne.

Wyjście

Twój program powinien wypisać na wyjście dokładnie N wierszy. W i -tym wierszu wyjścia powinna się znaleźć odpowiedź na i -te zapytanie. Odpowiedź na każde z zapytań powinna się składać z jednej liczby naturalnej określającej numer następnego wolnego sektora (po zajęтым w i -tej operacji). Jeśli po danym sektorze nie ma już wolnych sektorów – odpowiedzią na dane zapytanie jest jedno słowo NIE.

Ograniczenia

$1 \leq N \leq 500\,000$.

W testach wartych łącznie 20% maksymalnej punktacji zachodzi dodatkowy warunek: $N \leq 5\,000$.

Przykład

Input	Output
5	5
4	3
2	5
3	NIE
5	NIE
1	