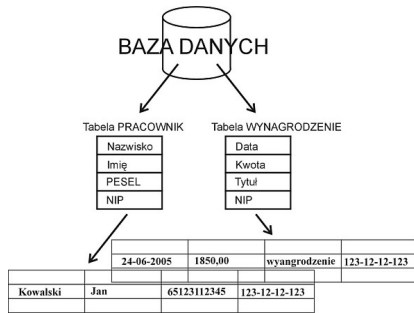


Baza danych



Od dziś Jasiu został pracownikiem firmy "Fraktax". Jego zadaniem na dziś jest wprowadzanie danych do bazy. Jako nowy pracownik, Jasio mógł tylko wprowadzać dane, wyszukiwać je oraz cofać i ponawiać operację wstawienia elementu do bazy. Jak się okazało system bazodanowy uległ awarii. Młody pracownik chciał zrobić dobre wrażenie i postanowił naprawić system lub napisać go od nowa. Niestety nie potrafi programować, więc poprosił ciebie o pomoc. Napisz program, który będzie symulował opisaną sytuację.

Poniżej znajdują się niezbędne informacje:

- system działa na algorytmie binarnego drzewa wyszukiwawczego,
- wprowadzane wartości są unikatowymi liczbami całkowitymi z zakresu $[0..2^{32}-1]$,
- początkowa postać bazy danych jest już wypełniona liczbami zgodnie z algorytmem BST i tych liczb **nie wolno** modyfikować
- Jasiu może dodawać nowy element oraz przeszukiwać bazę a także cofać operację wstawienia i ponawiać cofniętą operację
- cofać można operacje maksymalnie do postaci początkowej bazy danych, kolejna próba cofnięcia nic nie zmienia w bazie
- ponawiać można do momentu ostatnio wstawionej wartości (jeśli do pierwotnej postaci bazy wstawimy 3 liczby, następnie cofniemy wstawienie dwa razy, i wstawimy jedną liczbę, to możemy cofnąć się maksymalnie dwa razy, a po tej operacji ponowić operację wstawienia także dwa razy), każda następna próba ponowienia nic nie zmienia w bazie.

Wejście

W pierwszym wierszu jedna liczba całkowita dodatnia n określająca początkową liczbę danych w bazie ($n \leq 10^5$).

W drugim wierszu n unikatowych liczb naturalnych, które należy wstawić do drzewa BST.

Następnie jedna liczba q określająca liczbę zapytań ($q \leq 200\,000$). Zapytanie może przyjmować jedną z czterech postaci:

- **i [liczba]** - wstaw do bazy liczbę [liczba]
- **f [liczba]** - wyszukaj w bazie liczbę [liczba]
- **z** - cofnij operację wstawienia
- **y** - ponów operację

Wyjście

Dla każdego polecenia **f [liczba]** w danym wierszu wypisujemy kolejne wartości drzewa, przez jakie przechodzimy wyszukując liczbę wypisując na końcu **ok** jeśli liczba została znaleziona lub **brak** jeśli liczby nie ma w bazie.

Przykład

Wejście:

8
5 1 6 4 9 2 8 0
17
f 6
f 11
f 3
i 11
i 3
f 11
f 3
z
f 11
f 3
z
f 11
f 3
y
y
f 11
f 3

Wyjście:

5 6 ok
5 6 9 brak
5 1 4 2 brak
5 6 9 11 ok
5 1 4 2 3 ok
5 6 9 11 ok
5 1 4 2 brak
5 6 9 brak
5 1 4 2 brak
5 6 9 11 ok
5 1 4 2 3 ok