



Zadanie V: Kolejka

Twoim zadaniem jest efektywna implementacja interfejsu kolejki (FIFO) z ograniczoną pojemnością. Interfejs zawiera operację sprawdzania wartości i usuwania pierwszego elementu, dodawania elementu na końcu. Zakładamy, że na samym początku kolejka jest pusta.

Uwaga! Nie korzystaj z kontenera `std::queue` lub `std::deque`. Twoja implementacja powinna używać tablicy a nie listy. Wykorzystaj ideę cyklicznego bufora.

Format wejścia

Pierwsza linia wejścia dwie liczby całkowite n i k , $0 \leq n, k \leq 10^6$ - ilość poleceń. n definiuje pojemność kolejki, tzn. maksymalną ilość elementów. W kolejnych k liniach znajduje się jedno polecenie z poniższych:

- *PUSH* x - dodaje element x na koniec kolejki,
- *POP* - usuwa pierwszy element z kolejki,
- *FRONT* - zwraca wartość pierwszego elementu w kolejce.

Wartości elementów są ograniczone: $0 \leq x \leq 10^6$.

Format wyjścia

Na wyjściu dla każdej komendy powinien zostać wypisany wynik jej działania:

- *PUSH* - napis *OK* lub *FULL* jeśli kolejka jest pełna,
- *POP* - napis *OK* lub *EMPTY* jeśli kolejka jest pusta,
- *FRONT* - wartość pierwszego elementu w kolejce lub napis *NONE* jeśli kolejka jest pusta.



Przykład

Dla danych wejściowych:

3 11
FRONT
PUSH 4
PUSH 3
PUSH 9
PUSH 8
FRONT
POP
POP
FRONT
POP
POP

Poprawną odpowiedzią jest

NONE
OK
OK
OK
FULL
4
OK
OK
9
OK
EMPTY