**Ví dụ**

**Input**

Case 1:

3 6

N

N

E 1

N

N

N

queue = 1 2 3

* Thao tác 1: N

**→ 1**

queue = 2 3 1

* Thao tác 2: N

**→ 2**

queue = 3 1 2

* Thao tác 3: E 1

queue = 1 3 2

* Thao tác 4: N

**→ 1**

queue = 3 2 1

* Thao tác 5: N

**→ 3**

queue = 2 1 3

* Thao tác 6: N

**→ 2**

queue = 1 3 2

Case 2:

10 2

N

N

queue = 1 2 3 4 5 6 7 8 9 10

* Thao tác 1: N

**→ 1**

queue = 2 3 4 5 6 7 8 9 10 1

* Thao tác 2: N

**→ 2**

queue = 3 4 5 6 7 8 9 10 1 2

Cho chúng ta một cái queue, ban đầu chúng ta có P người theo thứ tự 1 2 3 … P ở trong queue

Thao tác **‘N’** → Chỉ cần đưa người đứng đầu queue ra, rồi chuyển về cuối queue → O(1)

Thao tác **‘E’ x**:

* Gọi n là số lượng phần tử trong queue
* Thêm số x vào cuối queue
* Lặp lại n lần thao tác: chuyển những giá trị khác x từ đầu queue về cuối queue.

→ O(n) // n là số phần tử trong queue

P <= 10^9

Giả sử C lần E x → Thời gian: **O(P \* C)**

Không gian: **O(P)**

**Giả sử muốn đưa số 4 về đầu của queue**

**queue = 1 2 3 4 5**

**B1: n = 5**

**B2: 1 2 3 4 5 4**

**B3:**

**Lần 1: 2 3 4 5 4 1**

**Lần 2: 3 4 5 4 1 2**

**Lần 3: 4 5 4 1 2 3**

**Lần 4: 5 4 1 2 3**

**Lần 5: 4 1 2 3 5**

**P** rất lớn, nhưng **C** rất nhỏ

→ Chỉ có tối đa là **C** người được khám bệnh

→ Chỉ cần quan tâm **C** người đầu tiên **(1 <= id <= C)**

Nếu không có **E x** → In theo lần lượt từ **1** tới **C**

Nếu có **E x**: Chỉ cần thêm x vào cuối, và chuyển n lần những giá trị khác x từ đầu về cuối queue.

**→ O(min(P, C) \* C)**

**Giải thuật**

* **Bước 1:** Khởi tạo hàng đợi khám bệnh. Đưa vào hàng đợi lần lượt các giá trị từ **1** đến **min(P, C)**.
* **Bước 2:** Duyệt qua từng truy vấn:
  + Truy vấn **‘N’**: In ra phần tử đầu tiên ở đầu hàng đợi, xóa phần tử đó và đưa nó lại vào cuối hàng.
  + Truy vấn **‘E’ x**:
    - Gán **n** là kích thước hiện tại của hàng đợi.
    - Đưa **x** vào cuối hàng đợi.
    - Lặp **n** lần, đưa lần lượt các phần tử khác **x** vào cuối hàng.
* **Bước 3:** Reset các biến, chuẩn bị cho bộ test tiếp theo.

**Mã giả**

**tc = 0;**

**while true: {**

**read(P, C);**

**if P == 0 and C == 0:**

**break;**

**tc += 1;**

**q = queue();**

**for i = 1 to min(P, C):**

**q.push(i);**

**print(‘Case ’, tc, ‘:’);**

**for i = 1 to C: {**

**read(c); // đọc vào 1 ký tự**

**if c == ‘N’: {**

**print(q.front());**

**q.push(q.front());**

**q.pop();**

**} else {**

**read(x);**

**n = q.size();**

**q.push(x);**

**for j = 1 to n {**

**temp = q.front();**

**q.pop();**

**if temp != x:**

**q.push(temp);**

**}**

**}**

**}**

**}**

**Độ phức tạp:**

Không gian: **O(min(P, C))**

Thời gian: **O(min(P, C) \* C)**