

Street Parade

Big-O Blue - Lecture 04: Stack & Queue

Tóm tắt đề bài

Tóm tắt đề bài

- Ban tổ chức muốn sắp xếp các chiếc xe đang đỗ trong *bãi* ra trình diễn theo *thứ tự từ nhỏ đến lớn*.
- Một chiếc xe có quyền đỗ tạm vào một *con hẻm* trên đường đi nếu chưa tới lượt nó và *con hẻm* chỉ đủ rộng để một chiếc xe di chuyển tại một thời điểm.

Yêu cầu: xác định xem ban tổ chức có thể thực hiện được màn trình diễn đúng như dự định hay không.

Mô tả Input/Output

Input

Gồm một hoặc nhiều bộ test. Mỗi bộ test có định dạng như sau:

- Dòng đầu tiên: số nguyên dương n số lượng xe đang đỗ trong bãi ($n \leq 1000$)
- Dòng thứ hai: n số nguyên từ 1 đến n với thứ tự ngẫu nhiên đại diện cho số thứ tự của các chiếc xe đang đỗ trong bãi.

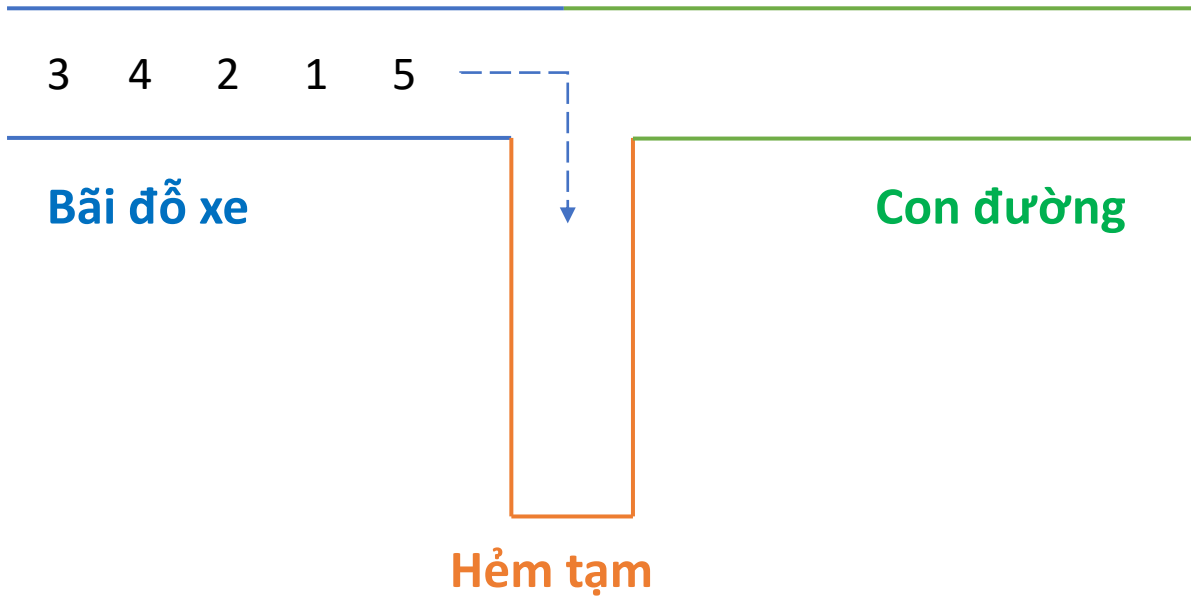
Kết thúc toàn bộ bộ test là số 0.

Output

Yes/No tương ứng với màn trình diễn thực hiện được/không thực hiện được

Giải thích ví dụ

Ví dụ

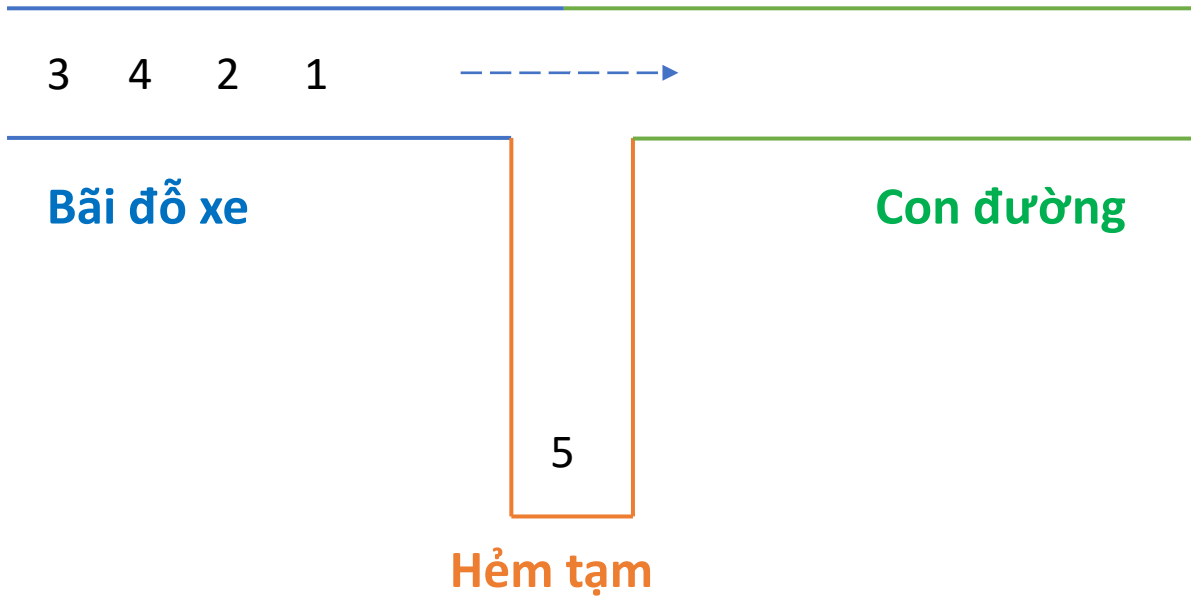


Input	Output
5 5 1 2 4 3 0	yes

Thứ tự xe được ra đường: 1
Không có xe trong hẻm
Xe lấy ra trong bãi đỗ: 5
→ Đưa xe 5 vào hẻm tạm



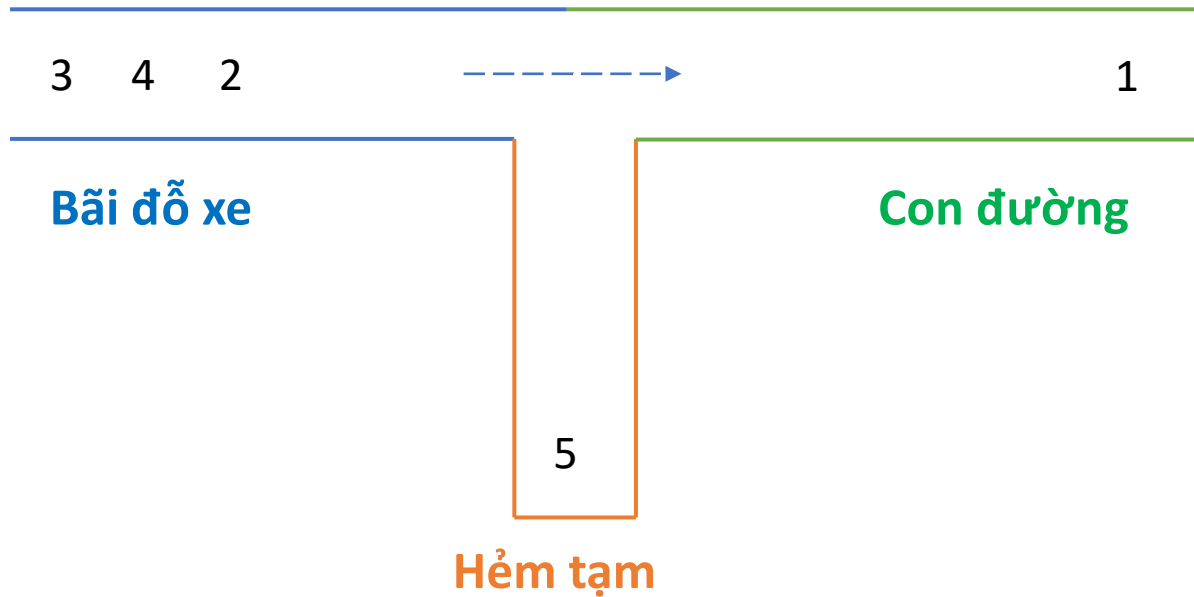
Ví dụ



Input	Output
5 5 1 2 4 3 0	yes

Thứ tự xe được ra đường: 1
 Xe được lấy ra trong hẻm: 1 ✓
 → Đưa xe 1 ra trình diễn

Ví dụ

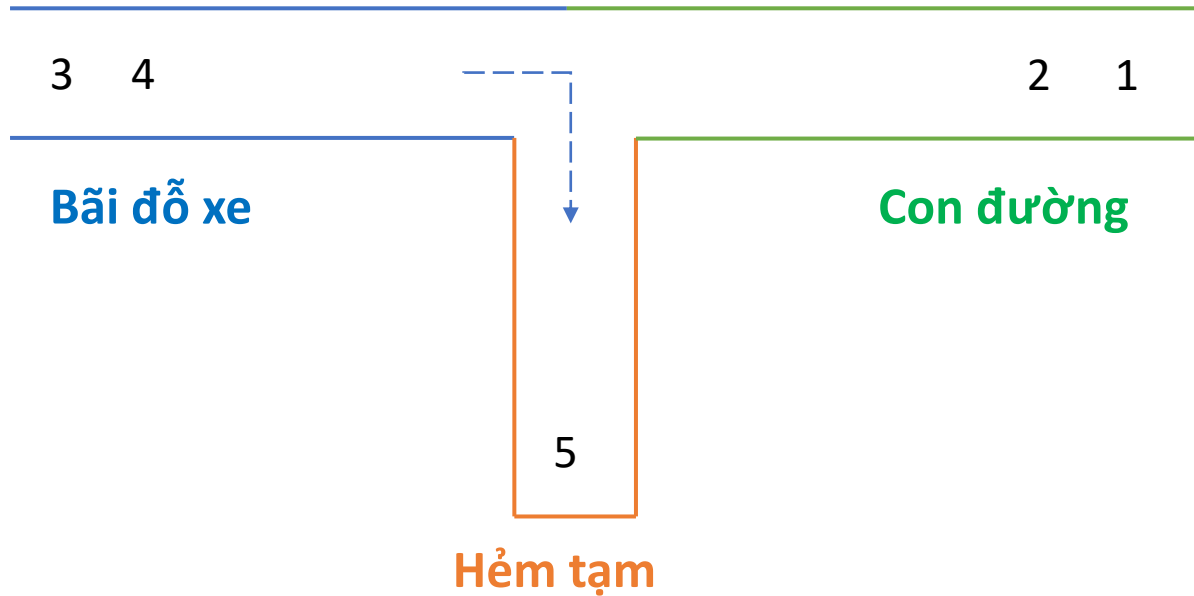


Input	Output
5 5 1 2 4 3 0	yes

Thứ tự xe được ra đường: 2
Xe lấy ra trong hẻm: 2
→ Đưa xe 2 ra đường



Ví dụ

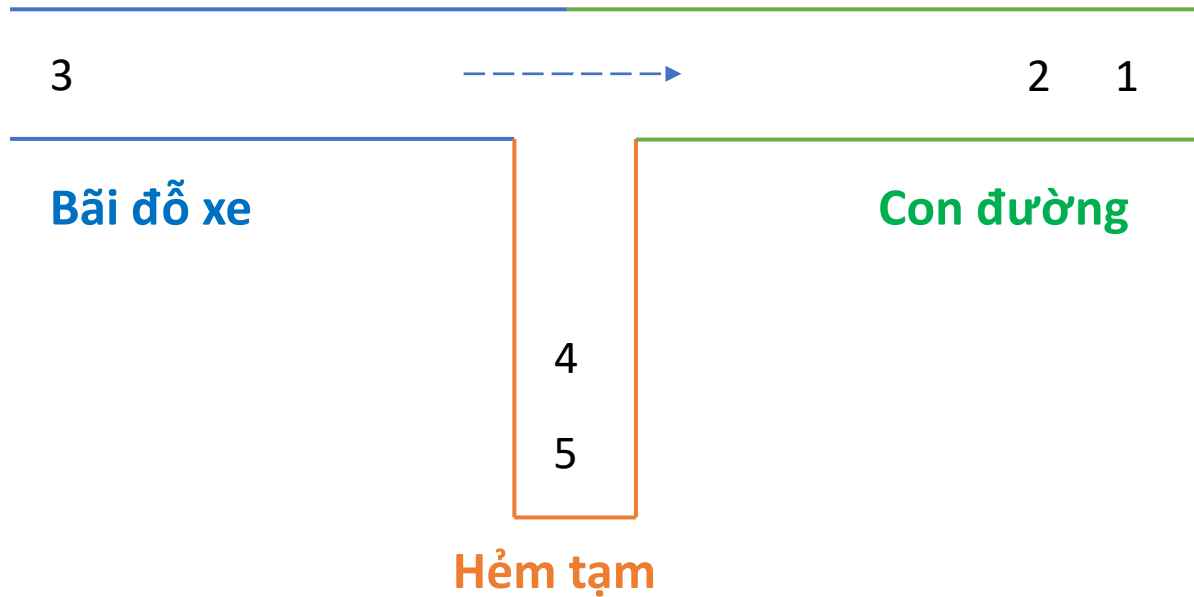


Input	Output
5	yes
5 1 2 4 3	
0	

Thứ tự xe được ra đường: 3
 Xe được lấy ra trong bãi: 4
 Xe được lấy ra trong hẻm: 5
 → Đưa xe 4 vô hẻm



Ví dụ

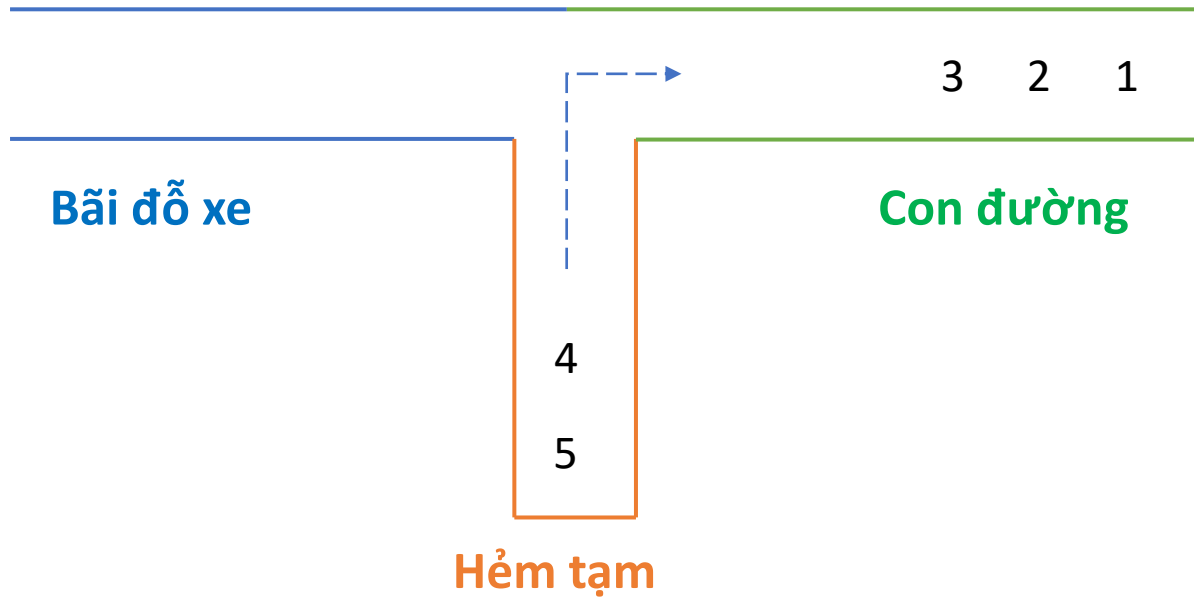


Input	Output
5 5 1 2 4 3 0	yes

Thứ tự xe được ra đường: 3
 Xe lấy ra trong bãi đỗ: 3
 → Đưa xe 3 ra đường



Ví dụ

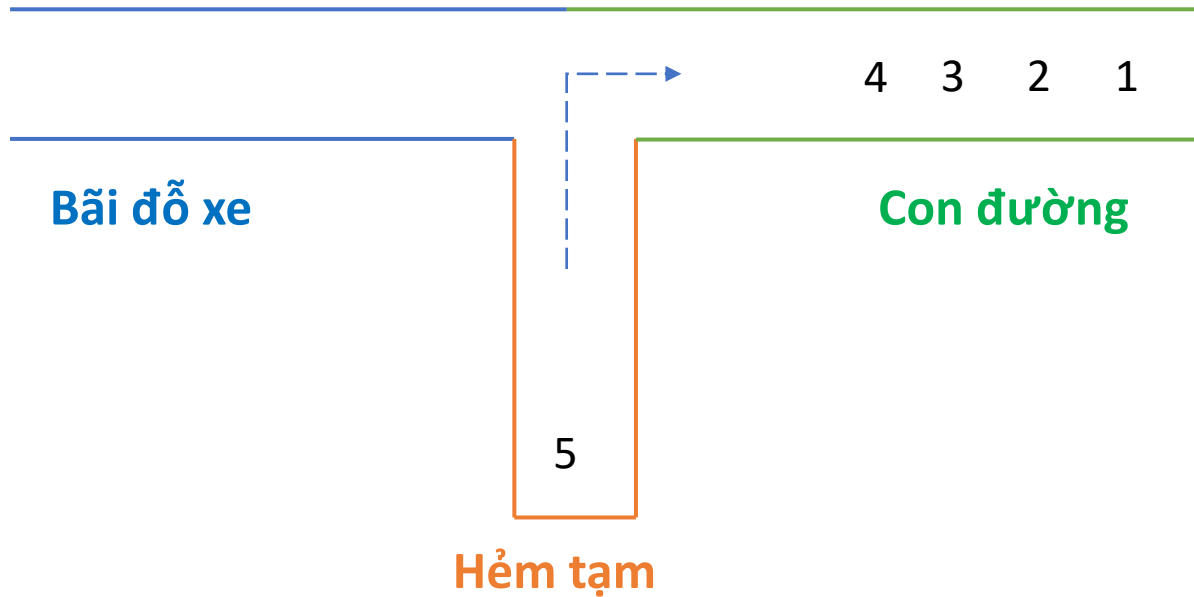


Input	Output
5	yes
5 1 2 4 3	
0	

Thứ tự xe được ra đường: 4
 Trong bãi hết xe
 Xe lấy ra trong hẻm: 4
 → Đưa xe 4 ra đường



Ví dụ

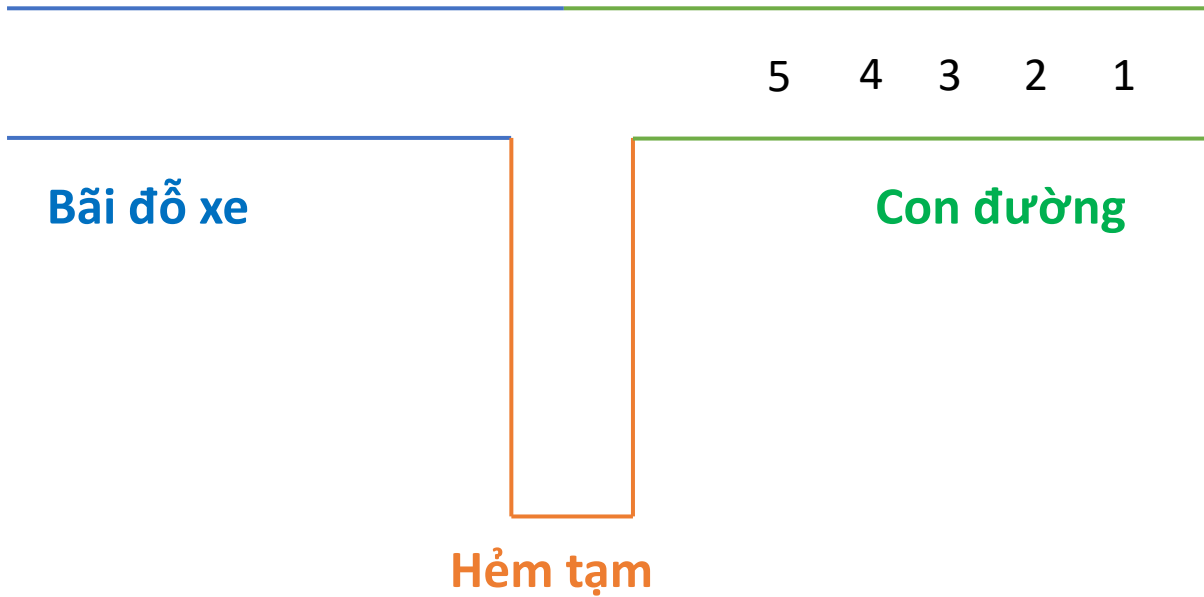


Input	Output
5 5 1 2 4 3 0	yes

Thứ tự xe được ra đường: 5
Trong bãi hết xe
Xe lấy ra trong hẻm: 5
→ Đưa xe 5 ra đường



Ví dụ



Input	Output
5 5 1 2 4 3 0	yes

Đủ 5 xe ra đường theo thứ tự từ nhỏ đến lớn → YES



Hướng dẫn giải

Nhận xét

- Xe nào vào hẻm trước thì sẽ ra sau
→ hẻm hoạt động như CTDL Stack
- Dùng 1 biến nextCar để kiểm soát thứ tự xe được ra ở thời điểm hiện tại.
- Với mỗi lần cần đưa một chiếc xe ra trình diễn:
 - Đối chiếu nextCar với xe ở bãi đỗ và xe ở hẻm để đưa các xe đến lượt ra trình diễn
 - Nếu không có xe nào được đưa ra, ta đưa xe đang đỗ trong bãi vào hẻm để xét xe kế tiếp nằm trong bãi.

Các bước giải

Bước 1: Đọc thông tin các xe hiện có trong bãi vào một mảng

Bước 2: Khởi tạo stack rỗng (hẻm) và biến đếm nextCar = 1 (thứ tự trình diễn)

Bước 3: Duyệt qua từng chiếc xe trong bãi:

- Nếu nextCar đúng bằng số thứ tự của xe trong bãi đỗ → đưa ra trình diễn, tăng nextCar
- Ngược lại, xét xem xe trong hẻm có phù hợp không:
 - Nếu có, đem xe trong hẻm ra trình diễn, tăng nextCar
 - Nếu không, nghĩa là cả xe đang nằm trong bãi và xe đầu hẻm đều không đúng thứ tự → đưa xe hiện tại trong bãi vào hẻm để xét xe kế tiếp

Bước 4: Nếu hết xe trong bãi đỗ, kiểm tra các xe còn lại trong hẻm có thỏa thứ tự còn lại hay không và đưa ra trình diễn nếu đúng

Bước 5: Kiểm nếu nextCar đạt tới thứ tự $(n + 1)$ → đủ n chiếc xe ra trình diễn → in “yes”. Ngược lại, in “no”

Độ phức tạp: $O(T * n)$ với T là số lượng bộ test và n là số lượng xe hiện có trong bãi đỗ

Mã giả

Mã giả (lặp n test case)

```
While True:
    read n
    if n == 0:
        break
    read trucks
    side_truck = []
    nextCar = 1
    I = 0
```

Mã giả (xử lý từng test case)

```
While I < n:
    if trucks[i] == nextCar:
        nextCar += 1
        I += 1
    else if side_truck is not empty and top of side_truck == nextCar:
        pop side_truck
        nextCar += 1
    else
        push trucks[i] to side_truck
        I += 1
While side_truck is not empty and top of side_truck == nextCar
    nextCar += 1
    pop side_truck
Print('yes' if nextCar == n+1 else 'no')
```

Thank you