**Đề bài**: Số mà có các chữ số hai bên tương xứng mà bằng nhau là số đối xứng

Ví dụ: 121, 212, 12321, 123321

Đầu vào là 1 số bất kỳ, yêu cầu:

A, Kiểm tra số đó có phải là số đối xứng không

B, Nếu không là đối xứng thì có thể biến đổi số đó thành đối xứng được không?

Với điều kiện số xếp số từ giữa trở đi bé dần về hai bên.

Ví du:

123123 không là đối xứng nhưng có thể thành => 123321

1234512345 không là đối xứng nhưng có thể thành => 123454321

123121212 không đối xứng có thể thành => 112232211

1231212 không đối xứng và không biến đổi được.

Không cần lập trình, chỉ cần viết thuật toán

**BÀI LÀM**

Flow bên dưới:

Lưu số được nhập vào 1 biến a

Nhập 1 số

Đảo ngược a và lưu vào biến b

Tìm chữ số lớn nhất trong số a.   
Đồng thời đếm số lần xuất hiện của chữ số lớn nhất: countMax

a == b?

Print: a là số

đối xứng

số a có 1 chữ số?

số a có 2 chữ số?

Print: a không phải là số

đối xứng

Yes

No

Yes

No

Yes

countMax == số chữ số của a - 1?

Đếm số lần xuất hiện của các chữ số khác chữ số max

Tất cả các chữ số khác chữ số max đều có số lần xuất hiện là 1 số chẵn?

No

Yes

No

Yes

No