

# TÌM MA TRẬN NGHỊCH ĐẢO BẰNG PHƯƠNG PHÁP GAUSS – JORDAN

## THUẬT TOÁN

**INPUT:** Nhập ma trận vuông từ file “input.txt”

**OUTPUT:** Ma trận nghịch đảo của ma trận đã cho nếu ma trận đã cho khả nghịch.  
Ngược lại, in ra dòng chữ “**Ma trận không khả nghịch!!!**”

## MỘT SỐ HÀM SỬ DỤNG TRONG THUẬT TOÁN

- **TimPhanTuGiai():** *Tìm phần tử giải của ma trận A*

Duyệt từng hàng của ma trận A từ trên xuống dưới

B1: Bỏ qua những hàng có phần tử giải

B2: Xét từ trái qua phải:

- Nếu thấy có giá trị **1** hoặc **-1**, chọn đó làm phần tử giải

- Nếu không có 1 hoặc -1, chọn phần tử khác 0 có giá trị tuyệt đối lớn nhất

B3: Nếu tìm thấy phần tử giải, lưu giá trị và lưu vị trí hàng cột của nó.

Ngược lại, “**Ma trận không khả nghịch!!!**” → Dừng chương trình

- **GaussJordan():** *biến đổi tương đương trên hàng của ma trận bổ sung  $[A|E]$*

B1: Dùng hàm TimPhanTuGiai() để tìm phần tử giải  $A_{pq}$

B2: Biến đổi sơ cấp trên  $[A|E]$ :

- Giữ nguyên hàng p:  $(\text{hàng } p)^{(1)} = (\text{hàng } p)$

- Các phần tử khác tính theo công thức:

$$(\text{hàng } i)^{(1)} = (\text{hàng } i) - [A|E]_{ip} * (\text{hàng } p)$$

=> Ta thu được ma trận  $[A|E]^{(1)}$

B3: Lập lại bước 2 để tìm  $[A|E]^{(2)}$ ,  $[A|E]^{(3)}$ , ... cho đến khi không thể chọn được phần tử giải nữa

- **Chuanhoaheso():** để chuẩn hoá hệ số 1

Ở mỗi hàng của  $[A|E]^{(n)}$ , mọi phần tử của hàng đó chia cho phần tử giải của hàng đó

- **SoSanh(a, b, n):** kiểm tra xem 2 hàng đã sắp xếp đúng chưa

Xét 2 (hàng a) và (hàng b) của ma trận bên trái của  $[A|E]^{(n)}$  sau khi chuẩn hóa hệ số 1, xét các phần tử trên cùng 1 cột

nếu ở hàng a là 1 và ở hàng b là 0 => trả về giá trị 1

nếu ở hàng a là 0 và ở hàng b là 1 => trả về giá trị -1

trường hợp còn lại là cả 2 hàng đều bằng 0 => trả về giá trị 0

- **SapXepHang():** chuẩn hóa về ma trận  $[E|A^{-1}]$

Lập lại n lần:

Dùng hàm SoSanh(hàng i, hàng i+1, n) kiểm tra (hàng i) với các (hàng i+1) đã xếp đúng chưa

Nếu SoSanh(hàng i, hàng i+1, n) trả về giá trị 1 thì đổi chỗ 2 hàng đó trên ma trận  $[A|E]^{(n)}$  sau khi chuẩn hóa hệ số 1

**\*Main():** Chương trình chính

B1: Nhập input ma trận A vuông từ file "input.txt"

B2: Tính kích cỡ ma trận A là: n

B3: Tạo ma trận đơn vị E của A

B4: Với mỗi  $i = 0, 1, \dots, (n-1)$

Dùng hàm GaussJordan() để biến đổi tương đương trên hàng của ma trận bổ sung  $[A|E]$

B5: Dùng hàm Chuanhoaheso() để chuẩn hoá hệ số 1

B6: Dùng hàm SapXepHang() để chuẩn hóa về ma trận  $[E|A^{-1}]$

B7: Xuất ra ma trận  $A^{-1}$

## VÍ DỤ

Ma trận đã cho

```
[[ 3.  5.  7.]  
 [ 2.  1.  4.]  
 [-1. -3. -7.]]
```

=====

Ma trận nghịch đảo là:

```
[[ 0.16666667  0.46666667  0.43333333]  
 [ 0.33333333 -0.46666667  0.06666667]  
 [-0.16666667  0.13333333 -0.23333333]]
```

Ma trận đã cho

```
[[ 3.  5.  7.]  
 [ 2.  1.  4.]  
 [-3. -5. -7.]]
```

Ma trận không khả nghịch!!!!