

Thuật toán Gauss Jordan giải hệ phương trình $Ax = B$

Input: Ma trận hệ số mở rộng M , số cột n , số hàng m

Output: Kết luận nghiệm của hệ phương trình

Bước 1: Đưa ma trận về dạng bậc thang rút gọn bằng cách đưa các phần tử trên đường chéo chính thành 1, những phần tử không nằm trên đường chéo chính thành 0.

Bước 2: Xác định hạng của ma trận A (rA), hạng của ma trận bổ sung M (rM).

Bước 3: Kết luận nghiệm của hệ phương trình:

$rA = rM = n - 1$ thì hệ phương trình có nghiệm duy nhất.

$rA = rM < n - 1$ thì hệ phương trình có vô số nghiệm.

$rA < rM$ thì hệ phương trình vô nghiệm.

Gói rút gọn

Input: Ma trận hệ số mở rộng M , số cột n , số hàng m

Output: Ma trận rút gọn

Bước 1: Đổi chỗ dòng có phần tử nằm trên đường chéo chính bằng 0

for $i = 0$ to $n - 2$

 if ($M[i][i] == 0$)

 for $k = i + 1$ to m

 if ($M[k][i] != 0$)

 for $l = 0$ to $l < n$

$tg = M[i][l];$

$M[i][l] = M[k][l];$

$M[k][l] = tg;$

Bước 2: Xác định loại cần chuyển về 0 ($M[j][i]$ không nằm trên đường chéo chính và khác 0) hay về 1 ($M[j][i]$ nằm trên đường chéo chính và khác 1)

for $j = 0$ to m

```
if ((j != i) && (M[j][i] != 0))  
    loai = 0 rồi chuyển đến bước 3
```

```
if ((j == i) && (M[j][i] != 1))  
    loai = 1 rồi chuyển đến bước 4
```

Bước 3: Chuyển các phần tử không nằm trên đường chéo chính thành 0

```
for k = 0 to m - 1
```

```
    if (k != j)
```

```
        HeSoNhan = 0;
```

```
        kt = 0;
```

```
        for l = 0 to i - 1
```

```
            if (M[k][l] != 0)
```

```
                kt = 1;
```

```
            if ((kt == 0) && (M[k][i] != 0))
```

```
                dong = k;
```

```
                HeSoNhan = (-1) * M[j][i] / M[dong][i];
```

```
if (HeSoNhan != 0)
```

```
    for k = 0 to n - 1
```

```
        M[j][k] = M[j][k] + HeSoNhan * M[dong][k];
```

Bước 4: Chuyển các phần tử trên đường chéo chính thành 1

```
kt = 0;
```

```
for l = 0 to i - 1
```

```
    if (M[j][l] != 0)
```

```
        kt = 1;
```

```
HeSoNhan = M[j][i];
```

```
if ((HeSoNhan != 0) && (kt == 0))
```

```
    for l = 0 to n - 1
```

```
        M[j][l] = M[j][l] / HeSoNhan;
```

Gợi biện luận

Input: Ma trận hệ số mở rộng M, số cột n, số hàng m

Output: Kết luận về nghiệm của hệ

Bước 1: Tìm hạng của ma trận A

```
int rA = 0;
```

```
int tangHangA;
```

```
for i = 0 to m - 1
```

```
    tangHangA = 0;
```

```
    for j = 0 to n - 2
```

```
        if (M[i][j] != 0)
```

```
            tangHangA = 1;
```

```
    rA += tangHangA;
```

Bước 2: Tìm hạng của ma trận M

```
int rM = 0;
```

```
int tangHangM;
```

```
for i = 0 to m - 1
```

```
    tangHangM = 0;
```

```
    for j = 0 to n - 1
```

```
        if ((int)M[i][j] != 0)
```

```
            tangHangM = 1;
```

```
    rM += tangHangM;
```

Bước 3: Biện luận về nghiệm

- Trường hợp hệ có nghiệm duy nhất

```
if (rA == rM && rA == n - 1)
```

```
    printf("\n-> Hệ phương trình có nghiệm duy nhất là:");
```

```
    for i = 0 to n - 2
```

```
        printf("\n X%d = %lf", i + 1, M[i][n - 1]);
```

- Trường hợp hệ có vô số nghiệm

```
if (rA == rM && rA < n - 1)
```

```
    printf("\n-> He phuong trinh co vo so nghiem.\n");
```

```
    for i = n - 2 to 0
```

```
        printf("\n X%d = ", i + 1);
```

```
        printf("%lf", M[i][n - 1]);
```

```
        for j = i + 1 to n - 2
```

```
            if (M[i][j] != 0)
```

```
                printf(" %lf X%d", -M[i][j], j + 1);
```

- Trường hợp hệ vô nghiệm

```
if (rA < rM)
```

```
    printf("\n-> He phuong trinh vo nghiem.\n");
```