

```

int i,j,k,n,l,t;
float s,c,epsi;
char tl;

clrscr();
printf("Cho so phuong trinh n = ");
scanf("%d",&n);
printf("Cho cac phan tu cua ma tran a : \n");
for (i=1;i<=n;i++)
    for (j=1;j<=n;j++)
        {
            printf("a[%d][%d] = ",i,j);
            scanf("%f",&a[i][j]);
        }
printf("\n");
printf("Ma tran a ma ban da nhap");
printf("\n");
for (i=1;i<=n;i++)
    {
        for (j=1;j<=n;j++)
            printf("%15.5f",a[i][j]);
        printf("\n");
    }
printf("\n");
t=1;
flushall();
while (t)
    {
        printf("Co sua ma tran a khong(c/k)?");
        scanf("%c",&tl);
        if (toupper(tl)=='C')
            {
                printf("Cho chi so hang can sua : ");
                scanf("%d",&i);
                printf("Cho chi so cot can sua : ");
                scanf("%d",&j);
                printf("a[%d][%d] = ",i,j);
                scanf("%f",&a[i][j]);
            }
    }

```

```

        flushall();
    }
    if (toupper(tl)=='K')
        t=0;
}
printf("Ma tran a \n");
printf("\n");
for (i=1;i<=n;i++)
{
    for (j=1;j<=n;j++)
        printf("%10.5f",a[i][j]);
    printf("\n");
}
printf("\n");
printf("Cho cac phan tu cua ma tran f : \n");
for (i=1;i<=n;i++)
{
    printf("f[%d] = ",i);
    scanf("%f",&f[i]);
}
printf("\n");
printf("Ma tran f ma ban da nhap");
printf("\n");
for (i=1;i<=n;i++)
    printf("f[%d] = %10.5f\n",i,f[i]);
printf("\n");
t=1;
flushall();
while (t)
{
    printf("Co sua ma tran f khong(c/k)?");
    scanf("%c",&tl);
    if (toupper(tl)=='C')
    {
        printf("Cho chi so hang can sua : ");
        scanf("%d",&i);
        printf("f[%d] = ",i);
        scanf("%f",&f[i]);
    }
}

```

```

        flushall();
    }
    if (toupper(tl)=='K')
        t=0;
}
printf("\n");
printf("Ma tran f");
printf("\n");
for (i=1;i<=n;i++)
    printf("f[%d] = %10.5f\n",i,f[i]);
printf("\n");
for (i=1;i<=n;i++)
    x0[i]=0.0;
x0[1]=1.0;
printf("Cho so lan lap l = ");
scanf("%d",&l);
epsi=1e-5;
for (i=1;i<=n;i++)
{
    c=1.0/a[i][i];
    for (j=1;j<=n;j++)
        if (j!=i)
            a[i][j]*=c;
    f[i]*=c;
    a[i][i]=0.0;
}
k=1;
t=0;
do
{
    for (i=1;i<=n;i++)
    {
        x1[i]=f[i];
        for (j=1;j<=n;j++)
            x1[i]=x1[i]-a[i][j]*x0[j];
    }
    s=0.0;
    for (i=1;i<=n;i++)

```

```

    s=s+fabs(x1[i]-x0[i]);
if (s>=epsi)
    for (i=1;i<=n;i++)
        x0[i]=x1[i];
if (s<epsi)
    {
        t=1;
        printf("\n");
        printf("Phep lap hoi tu sau %d buoc tinh",k);
        printf("\n");
        printf("NGHIEM CUA HE");
        printf("\n");
        for (i=1;i<=n;i++)
            printf("x[%d] = %10.5f\n",i,x1[i]);
    }
k=k+1;
if (k>1)
    {
        t=1;
        printf("Phep lap khong hoi tu sau %d buoc tinh",k-1);
    }
}
while (t==0);
getch();
}

```

§6. PHƯƠNG PHÁP LẶP GAUSS - SEIDEL

Phương pháp lặp Gauss-Seidel được cải tiến từ phương pháp lặp đơn. Nội dung cơ bản của phương pháp là ở chỗ khi tính nghiệm xấp xỉ thứ $(k+1)$ của ẩn x_i ta sử dụng các xấp xỉ thứ $(k+1)$ đã tính của các ẩn x_1, \dots, x_{i-1} . Giả sử đã cho hệ: $\mathbf{AX} = \mathbf{B}$ và ta có nghiệm:

$$x_i = \beta_i + \sum_{j=1}^n \alpha_{ij} x_j \quad i = 1, \dots, n$$

Lấy xấp xỉ ban đầu tùy ý $x_1^{(0)}, x_2^{(0)}, \dots, x_n^{(0)}$ và tất nhiên ta cố gắng lấy chúng tương ứng với x_1, x_2, \dots, x_n (càng gần càng tốt). Tiếp theo ta giả sử rằng đã biết xấp xỉ thứ k $x_i^{(k)}$ của nghiệm. Theo Seidel ta sẽ tìm xấp xỉ thứ $(k+1)$ của nghiệm theo các công thức sau:

$$x_1^{(k+1)} = \beta_1 + \sum_{j=1}^n \alpha_{ij} x_j^{(k)}$$

$$x_2^{(k+1)} = \beta_1 + \alpha_{21} x_1^{(k+1)} + \sum_{j=2}^n \alpha_{ij} x_j^{(k)}$$

.....

$$x_i^{(k+1)} = \beta_i + \sum_{j=1}^{i-1} \alpha_{ij} x_j^{(k+1)} + \sum_{j=i}^n \alpha_{ij} x_j^{(k)}$$

.....

$$x_n^{(k+1)} = \beta_n + \sum_{j=1}^{n-1} \alpha_{ij} x_j^{(k+1)} + \alpha_{nn} x_n^{(k)}$$

Thông thường phương pháp Gauss - Seidel hội tụ nhanh hơn phương pháp lặp đơn nhưng tính toán phức tạp hơn. Để dễ hiểu phương pháp này chúng ta xét một ví dụ cụ thể:

Cho hệ phương trình :

$$\begin{cases} 10x_1 + x_2 + x_3 = 12 \\ 2x_1 + 10x_2 + x_3 = 13 \\ 2x_1 + 2x_2 + 10x_3 = 14 \end{cases}$$

nghiệm đúng của hệ là (1, 1, 1)

Ta đưa về dạng thuận tiện cho phép lặp :

$$\begin{cases} x_1 = 1.2 - 0.1x_2 - 0.1x_3 \\ x_2 = 1.3 - 0.2x_1 - 0.1x_3 \\ x_3 = 1.4 - 0.2x_1 - 0.2x_2 \end{cases}$$

Lấy $x_1^{(0)} = 1.2$; $x_2^{(0)} = 0$; $x_3^{(0)} = 0$;

Sử dụng phương pháp lặp Seidel ta có :

$$\begin{cases} x_1^1 = 1.2 - 0.1 \times 0 - 0.1 \times 0 = 1.2 \\ x_2^1 = 1.3 - 0.2 \times 1.2 - 0.1 \times 0 = 1.06 \\ x_3^1 = 1.4 - 0.2 \times 1.2 - 0.2 \times 1.06 = 0.948 \end{cases}$$

$$\begin{cases} x_1^2 = 1.2 - 0.1 \times 1.06 - 0.1 \times 0.948 = 0.9992 \\ x_2^2 = 1.3 - 0.2 \times 0.9992 - 0.1 \times 0.948 = 1.00536 \\ x_3^2 = 1.4 - 0.2 \times 0.9992 - 0.2 \times 1.00536 = 0.999098 \end{cases}$$

và cứ thế tiếp tục. Chương trình mô tả thuật toán Gauss - Seidel như sau:

Chương trình 4-8

```
#include <conio.h>
#include <stdio.h>
```

```

#include <math.h>
#include <stdlib.h>
#include <ctype.h>
#define max 6

void main()
{
    float b[max],x[max];
    float a[max][max];
    int i,j,k,n,dem,t1;
    float t,s,d,w,epsi;
    char tl;

    clrscr();
    printf("Cho so an so n = ");
    scanf("%d",&n);
    printf("Cho cac phan tu cua ma tran a : \n");
    for (i=1;i<=n;i++)
        for (j=1;j<=n;j++)
            {
                printf("a[%d][%d] = ",i,j);
                scanf("%f",&a[i][j]);
            }
    printf("\n");
    printf("Ma tran a ma ban da nhap\n");
    printf("\n");
    for (i=1;i<=n;i++)
        {
            for (j=1;j<=n;j++)
                printf("%10.5f",a[i][j]);
            printf("\n");
        }
    printf("\n");
    t1=1;
    fflush();
    while (t1)
        {
            printf("Co sua ma tran a khong(c/k)?");

```

```

scanf("%c",&t1);
if (toupper(t1)=='C')
{
    printf("Cho chi so hang can sua : ");
    scanf("%d",&i);
    printf("Cho chi so cot can sua : ");
    scanf("%d",&j);
    printf("a[%d][%d] = ",i,j);
    scanf("%f",&a[i][j]);
    fflush();
}
if (toupper(t1)=='K')
    t1=0;
}
printf("Ma tran a \n");
printf("\n");
for (i=1;i<=n;i++)
{
    for (j=1;j<=n;j++)
        printf("%15.5f",a[i][j]);
    printf("\n");
}
printf("\n");
printf("Cho cac phan tu cua ma tran b : \n");
for (i=1;i<=n;i++)
{
    printf("b[%d] = ",i);
    scanf("%f",&b[i]);
}
printf("\n");
printf("Ma tran b");
printf("\n");
for (i=1;i<=n;i++)
    printf("b[%d] = %10.5f \n",i,b[i]);
printf("\n");
t1=1;
fflush();
while (t1)

```

```

{
    printf("Co sua ma tran b khong(c/k)?");
    scanf("%c",&t1);
    if (toupper(t1)=='C')
    {
        printf("Cho chi so hang can sua : ");
        scanf("%d",&i);
        printf("b[%d] = ",i);
        scanf("%f",&b[i]);
        fflush();
    }
    if (toupper(t1)=='K')
        t1=0;
}
printf("\n");
printf("Ma tran b");
printf("\n");
for (i=1;i<=n;i++)
    printf("b[%d] = %10.5f\n",i,b[i]);
printf("\n");
printf("Cho so lan lap k : ");
scanf("%d",&k);
printf("\n");
w=1;
epsi=1e-8;

for (i=1;i<=n;i++)
    x[i]=0.0;
dem = 0;
do
{
    dem=dem+1;
    for (i=1;i<=n;i++)
    {
        s=0.0;
        for (j=1;j<=n;j++)
            s=s+a[i][j]*x[j];
        d=x[i];
    }
}

```

```

        x[i]=(1-w)*d+w*(-s+a[i][i]*d+b[i])/a[i][i];
        t=fabs(d-x[i]);
    }
}
while ((dem<=k)&&(t>epsi*fabs(x[n])));
if (t<epsi*fabs(x[n]))
{
    printf("Nghiem sau %d lan lap la :\n",dem);
    for (i=1;i<=n;i++)
        printf("x[%d] = %12.8f\n",i,x[i]);
}
else
{
    printf("Khong dat do chinh xac sau %d lan lap\n",k);
    printf("Nghiem cua lan lap cuoi cung la : \n");
    for (i=1;i<=n;i++)
        printf("x[%d] = %12.8f\n",i,x[i]);
}
getch();
}

```

§7. PHƯƠNG PHÁP CRAMER

Một trường hợp riêng của hệ phương trình, trong đó số phương trình bằng số ẩn, nghĩa là hệ có dạng :

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n = b_2 \\ \dots \\ a_{n1}x_1 + a_{n2}x_2 + \dots + a_{nn}x_n = b_n \end{cases}$$

trong đó $A = (a_{ij})$ là một ma trận không suy biến. Một hệ phương trình như vậy có tên là *hệ thống Cramer*

Định lí Cramer: Hệ thống Cramer có nghiệm duy nhất được cho bởi công thức

$$x_i = \frac{|A^{(i)}|}{|A|} \quad i = 1, \dots, n$$

trong đó $A^{(i)}$ là ma trận nhận được từ A bằng cách thay cột thứ i bởi cột B . Như vậy để giải hệ bằng phương pháp Cramer chúng ta lần lượt tính các định thức của ma trận và ma trận thay thế rồi tìm nghiệm theo công thức Cramer. Chương trình sau mô tả thuật toán này:

Chương trình 4-9

```
// Cramer;
#include <conio.h>
#include <stdio.h>
#include <ctype.h>
#define max 50

void main()
{
    float r[max][max],a[max][max];
    float b[max],x[max];
    float delta[max];
    int i,j,k,l,t,n,ok1,ok2,t1;
    float c,d;
    char tl;

    clrscr();
    printf("Cho so an cua phuong trinh n = ");
    scanf("%d",&n);
    printf("Cho cac phan tu cua ma tran a : \n");
    for (i=1;i<=n;i++)
        for (j=1;j<=n;j++)
            {
                printf("a[%d][%d] = ",i,j);
                scanf("%f",&a[i][j]);
            }
    printf("\n");
    printf("Ma tran a ma ban da nhap\n");
    printf("\n");
    for (i=1;i<=n;i++)
        {
            for (j=1;j<=n;j++)
                printf("%10.5f",a[i][j]);
            printf("\n");
        }
    printf("\n");
```

```

t1=1;
flushall();
while (t1)
{
    printf("Co sua ma tran a khong(c/k)?");
    scanf("%c",&t1);
    if (toupper(t1)=='C')
    {
        printf("Cho chi so hang can sua : ");
        scanf("%d",&i);
        printf("Cho chi so cot can sua : ");
        scanf("%d",&j);
        printf("a["i","j"] = ");
        scanf("%f",&a[i][j]);
        flushall();
    }
    if (toupper(t1)=='K')
        t1=0;
}
printf("Ma tran a \n");
printf("\n");
for (i=1;i<=n;i++)
{
    for (j=1;j<=n;j++)
        printf("%10.5f",a[i][j]);
    printf("\n");
}
printf("\n");
printf("Cho cac phan tu cua ma tran b : \n");
for (i=1;i<=n;i++)
{
    printf("b[%d] = ",i);
    scanf("%f",&b[i]);
}
printf("\n");
printf("Ma tran b ma ban da nhap\n");
printf("\n");
for (i=1;i<=n;i++)

```

```

printf("b[%d] = %10.5f\n",i,b[i]);
printf("\n");
t1=1;
flushall();
while (t1)
{
printf("Co sua ma tran b khong(c/k)?");
scanf("%c",&t1);
if (toupper(t1)=='C')
{
printf("Cho chi so hang can sua : ");
scanf("%d",&i);
printf("b[%d] = ",i);
scanf("%f",&b[i]);
flushall();
}
if (toupper(t1)=='K')
t1=0;
}
printf("\n");
printf("Ma tran b\n");
printf("\n");
for (i=1;i<=n;i++)
printf("%10.5f",b[i]);
printf("\n");

```

//thay b vao tung cot cua a de tinh cac dinh thuc

```

for (k=0;k<=n;k++)
{
for (i=1;i<=n;i++)
for (j=1;j<=n;j++)//thay cot b vao a
if (i==k)
r[j][i]=b[j];
else
r[j][i]=a[j][i];
}

```

//tinh dinh thuc

```

d=1.0;
i=1;

```

```

ok2=1;
while (ok2&&(i<=n))
{
    if (r[i][i]==0.0)
    {
        ok1=1;
        t=t+1;
        while (ok1&&(t<=n))
            if (r[t][i]!=0)
            {
                for (j=i;j<=n;j++)
                {
                    c=r[i][j];
                    r[i][j]=r[t][j];
                    r[k][j]=c;
                }
                d=-d;
                ok1=0;
            }
        else
            t=t+1;
        if (k>n)
        {
            printf("\n");
            printf("*** MA TRAN SUY BIEN ***");
            ok2=0;
            d=0.0;
        }
    }
    if (r[i][i]!=0)
    {
        c=r[i][i];
        for (j=i+1;j<=n;j++)
            r[i][j]=r[i][j]/c;
        for (t=i+1;t<=n;t++)
        {
            c=r[t][i];
            for (j=i+1;j<=n;j++)

```

```

                                r[t][j]=r[t][j]-r[i][j]*c;
                                }
                                }
                                i=i+1;
                                }
                                if (ok2)
                                for (i=1;i<=n;i++)
                                d=d*r[i][i];
                                delta[k]=d;
                                }
                                if (delta[0]==0.0)
                                printf("He da cho vo nghiem\n");
                                else
                                {
                                printf("\nNGHIEM CUA HE :\n");
                                printf("\n");
                                for (i=1;i<=n;i++)
                                {
                                x[i]=delta[i]/delta[0];
                                printf("x[%d] = %10.5f\n",i,x[i]);
                                }
                                }
                                getch();
                                }

```

§8. HỆ PHƯƠNG TRÌNH SỐ PHỨC

Giả sử ta có một hệ phương trình dạng số phức dạng $AX = B$ trong đó $A = C + jD$, $B = E + jF$ và $X = Y + jZ$. Ta viết lại phương trình dưới dạng :

$$(C + jD)(Y + jZ) = (E + jF)$$

Nhân biểu thức về trái và cân bằng phần thực với phần thực và phần ảo với phần ảo ta nhận được hệ mới :

$$\begin{cases} CY - DZ = E \\ DY + CZ = F \end{cases}$$

Như vậy chúng ta nhận được một hệ gồm $2n$ phương trình số thực. Giải hệ này và kết hợp các phần thực và phần ảo ta nhận được nghiệm của hệ phương trình ban đầu. Chương trình giải hệ phương trình như vậy cho ở dưới đây:

Chương trình 4-10

```
#include <conio.h>
#include <stdio.h>
#include <math.h>
#include <stdlib.h>
#include <ctype.h>
#define max 20

void main()
{
    int i,j,k,l,n,m;
    float s,t,a[max][max],b[max][max],x[max];

    clrscr();
    printf("Cho so an so cua phuong trinh n = ");
    scanf("%d",&n);
    printf("Cho phan thuc cua cac he so,ke ca ve phai \n");
    for (i=1;i<=n;i++)
        for (j=1;j<=n+1;j++)
            {
                printf("a[%d][%d] = ",i,j);
                scanf("%f",&a[i][j]);
            }
    printf("\n");
    printf("Cho phan ao cua cac he so,ke ca ve phai \n");
    for (i=1;i<=n;i++)
        for (j=1;j<=n+1;j++)
            {
                printf("b[%d][%d] = ",i,j);
                scanf("%f",&b[i][j]);
            }
    for (i=1;i<=n;i++)
        a[i][2*n+1]=a[i][n+1];
    for (i=n+1;i<=2*n;i++)
        a[i][2*n+1]=b[i-n][n+1];
    for (i=n+1;i<=2*n;i++)
        for (j=1;j<=n;j++)
```

```

        a[i][j]=b[i-n][j];
for (i=1;i<=n;i++)
    for (j=n+1;j<=2*n;j++)
        a[i][j]=-b[i][j-n];
for (i=n+1;i<=2*n;i++)
    for (j=n+1;j<=2*n;j++)
        a[i][j]=a[i-n][j-n];
m=2*n;
for (k=1;k<=m-1;k++)
    {
        s=0.0;
        for (i=k;i<=m;i++)
            {
                t=fabs(a[i][k]);
                if (s<=t)
                    {
                        s=t;
                        l=i;
                    }
            }
        for (j=k;j<=m+1;j++)
            {
                s=a[k][j];
                a[k][j]=a[l][j];
                a[l][j]=s;
            }
        if (fabs(a[k][k]/a[l][l])<=1e-08)
            {
                printf("Ma tran suy bien\n");
                getch();
                exit(1);
            }
        for (i=k+1;i<=m;i++)
            {
                s=a[i][k]/a[k][k];
                a[i][k]=0.0;
                for (j=k+1;j<=m+1;j++)
                    a[i][j]-=s*a[k][j];
            }
    }

```

```

    }
}
x[m]=a[m][m+1]/a[m][m];
for (i=m-1;i>=1;i--)
{
    s=a[i][m+1];
    for (j=i+1;j<=m;j++)
        s-=a[i][j]*x[j];
    x[i]=s/a[i][i];
}
printf("\n");
printf("Nghiem phuc cua he\n");
for (i=1;i<=n;i++)
    if (x[i+n]<0)
        printf("%10.5f-%10.5fj\n",x[i],fabs(x[i+n]));
    else
        printf("%10.5f+%10.5fj\n",x[i],x[i+n]);
getch();
}

```

Dùng chương trình này giải hệ phương trình :

$$\begin{cases} (3 + 7j)x + (-2 + 4j)y + (1 - 3j)z + (4 + 2j)r = 8 + 36j \\ (5 - 6j)x + (2 + 5j)z + (-3 + j)r = 4 + 10j \\ (4 + 5j)x + (1 + 2j)y + (-5 - j)z + 6r = 13 - 3j \\ (2 + 4j)x + (1 - j)y + (2 - 3j)r = -10 + 6j \end{cases}$$

Ta nhận được các nghiệm $x = 2 + 3j$; $y = 1 - 2j$; $z = -1 + 4j$ và $r = 1 - j$

Ngoài các phương pháp nêu trên ta thấy rằng từ hệ phương trình $AX = B$ ta có thể tìm nghiệm X của hệ bằng cách viết lại phương trình dưới dạng $X = B/A = A^{-1}B$ với A^{-1} là ma trận nghịch đảo của A . Do vậy trước hết ta cần tìm A^{-1} và sau đó tính tích $A^{-1}B$.