

Nama : Rachel Savitri

NIM : 21120122140111

Kelas : C

Penjabaran Kode Pada Metode

Dekomposisi LU Gauss

```
int main()
{
    cout.precision(4);
    cout.setf(ios::fixed);
    int m,n,i,j,k, pilih, maks, flag=0, count=0;
    cout<<"\t\t\t RACHEL SAVITRI " <<endl;
    cout<<endl;
    cout<<"Inputan berupa Sistem persamaan linier yang akan dijadikan
dalam bentuk matriks" <<endl<<endl;
    cout<<"-Silahkan tentukan ordo matriks-" <<endl<<endl;
    cout<<"Masukkan matriks ordo: ";
    cin>>n;
    cout<<endl;
    double a[n][n+1];
    double x[n];
    double epsilon,y;
    for (i=0; i<n; i++)
    {
        for (j=0; j<=n; j++)
        {
            cout<<"Masukkan nilai matriks["<<i+1<<"]["<<j+1<<"] = ";
            cin>>a[i][j];
        }
    }
    cout<<"\nMasukkan nilai awal variabel:\n";
    for (i=0; i<n; i++)
    {
        cout<<"Nilai awal x"<<i<<" = ";
        cin>>x[i];
    }

    for (i=0; i<n; i++)
        for (k=i+1; k<n; k++)
            if (abs(a[i][i])<abs(a[k][i]))
                for (j=0; j<=n; j++)
                {
                    double temp=a[i][j];
                    a[i][j]=a[k][j];
                    a[k][j]=temp;
                }
    }
```

Kode dimulai dengan meminta pengguna untuk menginputkan ordo matriks SPL dan elemen-elemennya. Kemudian, baris matriks koefisien diurutkan berdasarkan elemen diagonalnya

untuk menghindari pembagian dengan nilai nol pada proses eliminasi Gauss. Langkah selanjutnya adalah melakukan eliminasi Gauss untuk mengubah matriks koefisien menjadi matriks segitiga atas. Setelah itu, pengguna diminta untuk memilih metode penyelesaian SPL: apakah menggunakan metode error atau iterasi. Berdasarkan pilihan pengguna, SPL kemudian diselesaikan dan solusinya ditampilkan.

```
//-----Kode Testing-----//
cout<<endl<<endl;
cout<<"Pengerjaan:"<<endl;
cout<<"1. Dengan Error"<<endl;
cout<<"2. Dengan Iterasi"<<endl;
cout<<"Masukkan Pilihan : ";
cin>>pilih;

if(pilih==1)
{
    cout<<"\nMasukkan epsilon :";
    cin>>epsilon;
    cout<<endl;
    cout<<setw(3)<<"Iterasi"<<setw(10);
    for (i=0; i<n; i++)
        cout<<"x"<<i<<setw(18);
    cout<<"\n-----";

    do
    {
        cout<<"\n"<<count+1<<". "<<setw(16);
        for (i=0; i<n; i++)
        {
            y=x[i];
            x[i]=a[i][n];
            for (j=0; j<n; j++)
            {
                if (j!=i)
                    x[i]=x[i]-a[i][j]*x[j];
            }
            x[i]=x[i]/a[i][i];
            if (abs(x[i]-y)<=epsilon)
                flag++;
            cout<<x[i]<<setw(18);
        }
        cout<<"\n";
        count++;
    }
    while(flag<n);
    cout<<"\nSolusinya : \n";
    for (i=0; i<n; i++)
        cout<<"x"<<i<<" = "<<x[i]<<endl;
    return 0;
}

else if(pilih==2)
{
    cout<<"\nMasukkan banyak iterasi:";
    cin>>maks;
```

```

        cout<<endl;
        cout<<setw(3)<<"Iterasi"<<setw(10);
        for(i=0; i<n; i++)
            cout<<"x"<<i<<setw(18);
        cout<<"\n-----"
        -----";
        for (m=0; m<maks; m++)
        {
            cout<<"\n"<<count+1<<". "<<setw(16);
            for (i=0; i<n; i++)
            {
                y=x[i];
                x[i]=a[i][n];
                for (j=0; j<n; j++)
                {
                    if (j!=i)
                        x[i]=x[i]-a[i][j]*x[j];
                }
                x[i]=x[i]/a[i][i];
                if(abs(x[i]-y)<=epsilon)
                    flag++;
                cout<<x[i]<<setw(18);
            }
            cout<<"\n";
            count++;
        }
        cout<<"\nSolusinya : \n";
        for (i=0; i<n; i++)
            cout<<"x"<<i<<" = "<<x[i]<<endl;
        return 0;
    }
}

```

Bagian kode tersebut merupakan tahap pengujian dari dua metode penyelesaian sistem persamaan linier (SPL), yaitu metode menggunakan error dan metode iteratif. Di sini, pengguna diminta untuk memilih metode yang ingin digunakan. Jika metode dengan error dipilih, program akan meminta nilai epsilon sebagai kriteria akurasi solusi, kemudian melakukan iterasi hingga solusi yang diperoleh memenuhi kriteria error yang ditentukan. Setelah itu, solusi SPL akhir akan ditampilkan. Sementara itu, jika metode iteratif dipilih, program akan meminta jumlah iterasi yang diinginkan, kemudian melakukan iterasi sesuai dengan jumlah yang diminta. Setelah iterasi selesai, solusi SPL akhir akan ditampilkan. Dengan menggunakan bagian kode ini, pengguna dapat memilih metode penyelesaian SPL yang sesuai dengan kebutuhan mereka dan memperoleh solusi yang tepat.

Hasil Output Dengan Iterasi 3x

```
"D:\tekkom\smt 4\metode nu x + v
-Silahkan tentukan ordo matriks-

Masukkan matriks ordo: 3

Masukkan nilai matriks[1][1] = 1
Masukkan nilai matriks[1][2] = 2
Masukkan nilai matriks[1][3] = 3
Masukkan nilai matriks[1][4] = 5
Masukkan nilai matriks[2][1] = 7
Masukkan nilai matriks[2][2] = 5
Masukkan nilai matriks[2][3] = 10
Masukkan nilai matriks[2][4] = 12
Masukkan nilai matriks[3][1] = 1
Masukkan nilai matriks[3][2] = 3
Masukkan nilai matriks[3][3] = 5
Masukkan nilai matriks[3][4] = 7

Masukkan nilai awal variabel:
Nilai awal x0 = 1
Nilai awal x1 = 1
Nilai awal x2 = 1

Pengerjaan:
1. Dengan Error
2. Dengan Iterasi
Masukkan Pilihan : 2

Masukkan banyak iterasi:3

Iterasi      x0      x1      x2
-----
1.      -0.4286      0.8095      1.2698
2.      -0.6780      0.4429      1.5974
3.      -0.8841      -0.0343      1.9842

Solusinya :
x0 = -0.8841
x1 = -0.0343
x2 = 1.9842

Process returned 0 (0x0)  execution time : 32.441 s
Press any key to continue.
```

Hasil Output Dengan Error

```
"D:\tekkom\smt 4\metode nu x + v
RACHEL SAVITRI

Inputan berupa Sistem persamaan linier yang akan dijadikan dalam bentuk matriks
-Silahkan tentukan ordo matriks-

Masukkan matriks ordo: 3

Masukkan nilai matriks[1][1] = 4
Masukkan nilai matriks[1][2] = 2
Masukkan nilai matriks[1][3] = 1
Masukkan nilai matriks[1][4] = 8
Masukkan nilai matriks[2][1] = 2
Masukkan nilai matriks[2][2] = 3
Masukkan nilai matriks[2][3] = 3
Masukkan nilai matriks[2][4] = 6
Masukkan nilai matriks[3][1] = 1
Masukkan nilai matriks[3][2] = 3
Masukkan nilai matriks[3][3] = 5
Masukkan nilai matriks[3][4] = 10

Masukkan nilai awal variabel:
Nilai awal x0 = 1
Nilai awal x1 = 1
Nilai awal x2 = 1

Pengerjaan:
1. Dengan Error
2. Dengan Iterasi
Masukkan Pilihan : 1

Masukkan epsilon :0.00001

Iterasi      x0      x1      x2
-----
1.      1.2500      0.1667      1.6500
2.      1.5042      -0.6528      2.0908
3.      1.8037      -1.2933      2.4152
4.      2.0428      -1.7771      2.6577
5.      2.2241      -2.1405      2.8395
```

```
"D:\tekkom\smt 4\metode nu x + v
31. 2.7689 -3.2302 3.3843
32. 2.7690 -3.2303 3.3844
33. 2.7691 -3.2304 3.3844
34. 2.7691 -3.2305 3.3845
35. 2.7691 -3.2306 3.3845
36. 2.7692 -3.2306 3.3845
37. 2.7692 -3.2307 3.3846
38. 2.7692 -3.2307 3.3846
39. 2.7692 -3.2307 3.3846
40. 2.7692 -3.2307 3.3846
41. 2.7692 -3.2307 3.3846

Solusinya :
x0 = 2.7692
x1 = -3.2307
x2 = 3.3846

Process returned 0 (0x0) execution time : 36.099 s
Press any key to continue.
```

Pembuktian Hasil

Kalkulator Sistem Persamaan Linier

Pilih berapa variabel di dalam sistem persamaan

3 variabel

Masukkan koefisien-koefisien untuk sistem persamaan serta pilihan pembulatan

1	4	q	+	2	r	+	1	s	=	8
2	2	q	+	3	r	+	3	s	=	6
3	1	q	+	3	r	+	5	s	=	10

Pembulatan: angka di belakang koma

Solusi:

$q = 2,76923$
 $r = -3,23077$
 $s = 3,38462$