Tugas Besar I Matriks dan Ruang Vektor

Sistem Persamaan Linier, Determinan, dan Aplikasinya

Nama :

Muhammad Asyroful Nur Maulana Yusuf (119140026)

Muhammad A.M Hikar Syahrial (119140031)

Makruf Alkarkhi (119140075)

INSTITUT TEKNOLOGI SUMATERA

TAHUN AJARAN

2020/2021

Foto Kelompok

Muhammad Asyroful Nur Maulana Yusuf (119140026)

Muhammad A.M Hikar Syahrial (119140031)

Makruf Alkarkhi (119140075)

BAB I

A. Deskripsi Masalah

Buatlah program dalam Bahasa C untuk

1. Menghitung solusi SPL dengan metode eliminasi Gauss, metode Eliminasi Gauss-Jordan, metode matriks balikan, dan kaidah Cramer (kaidah Cramer khusus untuk SPL dengan n peubah dan n persamaan).

2. Menghitung matriks balikan

3. Menghitung determinan matriks dengan berbagai metode (reduksi baris dan

ekspansi kofaktor).

Spesifikasi program adalah sebagai berikut:

1. Program dapat menerima masukan (input) baik dari keyboard maupun membaca masukan dari file text. Untuk SPL, masukan dari keyboard adalah m, n, koefisien aij , dan bi. Masukan dari file berbentuk matriks augmented tanpa tanda kurung, setiap elemen matriks dipisah oleh spasi. Misalnya,

3 4.5 2.8 10 12

-3 7 8.3 11 -4

0.5 -10 -9 12 0

2. Untuk persoalan menghitung determinan dan matriks balikan, masukan dari keyboard adalah n dan koefisien aij. Masukan dari file berbentuk matriks, setiap elemen matriks dipisah oleh spasi. Misalnya,

3 4.5 2.8 10

-3 7 8.3 11

0.5 -10 -9 12

3. Untup persoalan SPL, luaran (output) program adalah solusi SPL. Jika solusinya tunggal, tuliskan nilainya. Jika solusinya tidak ada, tuliskan solusi tidak ada, jika solusinya banyak, maka tuliskan solusinya dalam bentuk parametrik (misalnya x4 = -2, x3 = 2s – t, x2 = s, dan x1 = t.)

4. Untuk persoalan determinan dan matriks balikan, maka luarannya sesuai dengan persoalan masing-masing

6. Luaran program harus dapat ditampilkan pada layar komputer dan dapat disimpan ke dalam file.

7. Bahasa program yang digunakan adalah C.

8. Program tidak harus berbasis GUI, cukup text-based saja, namun boleh menggunakan GUI (memakai kakas Eclipse misalnya).

9. Program dapat dibuat dengan pilihan menu. Urutan menu dan isinya dipersilakan dirancang masing-masing. Misalnya, menu:

MENU

1. Sistem Persamaaan Linier
2. Determinan
3. Matriks balikan
4. Keluar

Untuk pilihan menu nomor 1 ada sub-menu lagi yaitu pilihan metode:

1. Metode eliminasi Gauss
2. Metode eliminasi Gauss-Jordan
3. Metode matriks balikan
4. Kaidah Cramer

Begitu juga untuk pilihan menu nomor 2 dan 3.

BAB II

SPL dengan Metode Eliminasi Gauss

Metode "Eliminasi Gauss" merupakan suatu cara penyelesaian SPL dengan menggunakan bentuk matriks melalui teknik penyederhanaan matriks menjadi matriks yang lebih sederhana (diperkenalkan oleh Carl Friedrich Gauss), yaitu dengan melakukan operasi baris sehingga matriks tersebut menjadi matriks yang Eselon-baris. Teknis operasionalnya: dengan mengubah persamaan linier tersebut ke dalam matriks imbuhan (matriks yang diperluas atau teraugmentasi) dan mengoperasikannya. Setelah terbentuk matriks eselon-baris, maka lakukanlah substitusi balik untuk mendapatkan nilai dari variabelvariabel tersebut.

Eliminasi Gauss vs Eliminasi Gauss-Jordan

Metode Eliminasi Gauss bertujuan untuk mengubah matriks A (matriks Jacobi atau matriks koefisien) menjadi matriks segitiga atas, yaitu berbentuk:

1 x x x

0 1 x x

0 0 1 x

Metode Eliminasi Gauss-Jordan bertujuan untuk mengubah matriks A menjadi matriks diagonal (matriks identitas), yaitu semua elemen pada diagonal matriks bernilai 1, sedangkan semua elemen lainnya bernilai nol, sehingga bentuk matriksnya adalah

1 0 0 x

0 1 0 x

0 0 1 x

Metode Eliminasi Gauss-Jordan "lebih berat" dalam realisasinya, karena memerlukan tahapan "operasi komputasi" yang lebih banyak dibandingkan Eliminasi Gauss. Oleh karena itu, Eliminasi Gauss-Jordan tidak banyak digunakan dalam Komputasi Numerik dalam Ilmu Teknik.

Determinan

Determinan adalah suatu fungsi tertentu yang menghubungkan suatu bilangan real dengan suatu matriks persegi (matriks bujur-sangkar). Determinan matriks hanya dimiliki oleh matriks persegi. Determinan matriks juga digunakan untuk mencari matriks INVERS (matriks balikan) atau dapat juga digunakan untuk penyelesaian SPL dengan aturan Cramer. Bagaimanakah mencari determinan untuk matriks berordo 2, matriks berordo 3, dan bahkan matriks yang berordo lebih tinggi?

Determinan Matriks Persegi Jika diketahui matriks berordo 2 berikut :

$$A = \begin{matrix} a & b \\ c & d \end{matrix}$$

Maka determinan dari matriks di atas dapat dihitung menggunakan rumus berikut :

Det (A) = det $\begin{matrix} a & b \\ c & d \end{matrix}$ = (a×d - b×c)

Determinan Matriks Berordo 2 Jika diketahui matriks berordo 3 berikut:

$$\begin{matrix} a11 & a12 & a13 \\ a21 & a22 & a23 \\ a31 & a32 & a33 \end{matrix}$$

Maka determinan dari matriks di atas dapat dihitung dengan :

(a). Ekspansi Kofaktor

– Minor dan Kofaktor: Ekspansi Kofaktor Pada Baris, dan

– Minor dan Kofaktor: Ekspansi Kofaktor Pada Kolom

(b). Metode Sarrus

(c). Determinan Matriks Segitiga Atas/Bawah (Ordo sembarang)

Jika diketahui matriks berordo 3 berikut:

$$\begin{matrix} a11 & a12 & a13 \\ a21 & a22 & a23 \\ a31 & a32 & a33 \end{matrix}$$

Maka langkah-langkah penentuan determinan dari matriks di atas dengan **Minor** dan **Kofaktor** adalah sebagai berikut :

(a). Pertama kali, buat **minor** dari $a11$

M11 = $\begin{matrix} a22 & a23 \\ a32 & a33 \end{matrix}$ => det (M11) = |M11| = $(a22 \times a33 - a23 \times a32)$

(b). Kemudian, **kofaktor** dari $a11$

C11 = $(-1)^{1+1}$. det(M11) = $-1)^{1+1}$. $(a22 \times a33 - a23 \times a32)$

Langkah-langkah penentuan determinan dari matriks di atas sebagai berikut :

(c). Kemudian**, minor** dan **kofaktor** dari $a12$

M12 = $\begin{matrix} a21 & a23 \\ a31 & a33 \end{matrix}$ => det (M12) = |M12| = $(a21 \times a33 - a23 \times a31)$

C12 = $(-1)^{1+2}$. det(M12) = $(-1)^{3}$. $(a21 \times a33 - a23 \times a31)$

(d). Kemudian, **minor** dan **kofaktor** dari $a13$

M13 = $\begin{matrix} a21 & a22 \\ a31 & a32 \end{matrix}$ => det (M13) = |M13| = $(a21 \times a32 - a22 \times a31)$

C13 = $(-1)^{1+3}$. det(M13) = $(-1)^{4}$. $(a21 \times a32 - a22 \times a31)$

(e). Secara keseluruhan, **Determinan** adalah

det(A) = $a11 \times C11 + a12 \times C12 + a13 \times C13$

Invers

Invers matriks adalah sebuah kebalikan (invers) dari kedua matriks di mana apabila matriks tersebut dikalikan menghasilkan matriks persegi (AB = BA = |). Simbol dari invers matriks adalah pangkat -1 di atas hurufnya. Contoh matriks B adalah invers matriks A ditulis B = A^{-}1 dan matriks A adalah invers dari matriks B ditulis A = B-1. Matriks A dan B merupakan dua matriks yang saling invers (berkebalikan).

Penyelesaian Invers Matriks bisa dilakukan dengan dua cara :

1. Rumus Adjoint

$$A^{-1} = \frac{1}{det(A)} \cdot Adj(A)$$

2. Rumus Kofaktor

$$Adj(A) = \begin{pmatrix} C_{11} & C_{21} & \cdots & C_{n1} \\ C_{12} & C_{22} & \cdots & C_{n2} \\ \vdots & \vdots & \ddots & \vdots \\ C_{1n} & C_{12} & \cdots & C_{nm} \end{pmatrix}$$

Matriks Minor M_{ij} dari Matriks A

$$M_{11} = \begin{vmatrix} a_{22} & a_{23} \\ a_{32} & a_{33} \end{vmatrix} \quad M_{21} = \begin{vmatrix} a_{12} & a_{13} \\ a_{32} & a_{33} \end{vmatrix} \quad M_{31} = \begin{vmatrix} a_{12} & a_{13} \\ a_{22} & a_{23} \end{vmatrix}$$

$$M_{12} = \begin{vmatrix} a_{21} & a_{23} \\ a_{31} & a_{33} \end{vmatrix} \quad M_{22} = \begin{vmatrix} a_{11} & a_{13} \\ a_{31} & a_{33} \end{vmatrix} \quad M_{32} = \begin{vmatrix} a_{11} & a_{13} \\ a_{21} & a_{23} \end{vmatrix}$$

$$M_{13} = \begin{vmatrix} a_{21} & a_{22} \\ a_{31} & a_{32} \end{vmatrix} \quad M_{23} = \begin{vmatrix} a_{11} & a_{12} \\ a_{31} & a_{32} \end{vmatrix} \quad M_{33} = \begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix}$$

Kofaktor

$$C_{ij} = (-1)^{i+j}|\mathbf{M}_{ij}|$$

$$C_{11} = (-1)^{1+1}|M_{11}| = |M_{11}|$$
$$C_{12} = (-1)^{1+2}|M_{12}| = -|M_{12}|$$
$$C_{13} = (-1)^{1+3}|M_{13}| = |M_{13}|$$
$$C_{21} = (-1)^{2+1}|M_{21}| = -|M_{21}|$$
$$C_{22} = (-1)^{2+2}|M_{22}| = |M_{22}|$$
$$C_{23} = (-1)^{2+3}|M_{23}| = -|M_{23}|$$
$$C_{31} = (-1)^{3+1}|M_{31}| = |M_{31}|$$
$$C_{32} = (-1)^{3+2}|M_{32}| = -|M_{32}|$$
$$C_{33} = (-1)^{3+3}|M_{32}| = |M_{33}|$$

SPL dengan Metode Cramer

Kaidah Cramer adalah salah satu cara yang dapat digunakan untuk menyelesaikan sistem persamaan linear. Metode ini menggunakan determinan suatu matriks dan matriks lain yang diperoleh dengan mengganti salah satu kolom dengan vektor yang terdiri dari angka di sebelah kanan persamaannya.

Contoh

Berikut adalah sistem persamaan linear:

$$\begin{cases} a_1x + b_1y & = c_1 \\ a_2x + b_2y & = c_2 \end{cases}$$

Maka matriks persamaanya adalah:

$$\begin{bmatrix} a_1 & b_1 \\ a_2 & b_2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} c_1 \\ c_2 \end{bmatrix}.$$

Jika $a_1b_2 - b_1a_2$ bukan nol, maka x dan y dapat dicari dengan menggunakan determinan matriks berikut:

$$x = \frac{\begin{vmatrix} c_1 & b_1 \\ c_2 & b_2 \end{vmatrix}}{\begin{vmatrix} a_1 & b_1 \\ a_2 & b_2 \end{vmatrix}} = \frac{c_1b_2 - b_1c_2}{a_1b_2 - b_1a_2}, \quad y = \frac{\begin{vmatrix} a_1 & c_1 \\ a_2 & c_2 \end{vmatrix}}{\begin{vmatrix} a_1 & b_1 \\ a_2 & b_2 \end{vmatrix}} = \frac{a_1c_2 - c_1a_2}{a_1b_2 - b_1a_2}.$$

Untuk matriks 3×3, caranya sama:

$$\begin{cases} a_1x + b_1y + c_1z &= d_1 \\ a_2x + b_2y + c_2z &= d_2 \\ a_3x + b_3y + c_3z &= d_3 \end{cases}$$

Berikut bentuk persamaannya jika diubah menjadi matriks:

$$\begin{bmatrix} a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \\ a_3 & b_3 & c_3 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} d_1 \\ d_2 \\ d_3 \end{bmatrix}.$$

Nilai x, y dan z kemudian bisa dicari dengan menggunakan cara:

$$x = \frac{\begin{vmatrix} d_1 & b_1 & c_1 \\ d_2 & b_2 & c_2 \\ d_3 & b_3 & c_3 \end{vmatrix}}{\begin{vmatrix} a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \\ a_3 & b_3 & c_3 \end{vmatrix}}, \quad y = \frac{\begin{vmatrix} a_1 & d_1 & c_1 \\ a_2 & d_2 & c_2 \\ a_3 & d_3 & c_3 \end{vmatrix}}{\begin{vmatrix} a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \\ a_3 & b_3 & c_3 \end{vmatrix}}, \text{ and } z = \frac{\begin{vmatrix} a_1 & b_1 & d_1 \\ a_2 & b_2 & d_2 \\ a_3 & b_3 & d_3 \end{vmatrix}}{\begin{vmatrix} a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \\ a_3 & b_3 & c_3 \end{vmatrix}}.$$

BAB III

A. Struktur Data

```
#include <iostream>

#include <math.h>

#include <bits/stdc++.h>

#include<iomanip>

#include <cstdlib>

#define M 10

using namespace std;


//cramer

void input(int array[][3], int array1[][1]);

int determinan(int array[][3]);

int hitung(int array[][3], int a, int b, int c);

int salin(int array[][3], int array1[][1], int a);

void eksekusi_salin(int array[][3], int array1[][3]);


//rumus invers

void cetak_matriks(float a[][M], int n)

{

    for (int i = 0; i < n; i++)

    {

        for (int j = 0; j <= n; j++)

        {

            cout << a[i][j] << " ";

        }

        cout << endl;

    }

}
```

```
int operasi(float a[][M], int n)
{
    int i, j, k = 0, c, flag = 0, m = 0;
    float pro = 0;

    for (i = 0; i < n; i++)
    {
        if (a[i][i] == 0)
        {
            c = 1;
            while ((i + c) < n && a[i + c][i] == 0)
                c++;
            if ((i + c) == n)
            {
                flag = 1;
                break;
            }
            for (j = i, k = 0; k <= n; k++)
                swap(a[j][k], a[j+c][k]);
        }
        for (j = 0; j < n; j++)
        {
            if (i != j)
            {
                float pro = a[j][i] / a[i][i];

                for (k = 0; k <= n; k++)
                {
                    a[j][k] = a[j][k] - (a[i][k]) * pro;
                }
            }
```

```
        }
    }
    return flag;
}

void cetak_hasil(float a[][M], int n, int flag)
{
    cout << "hasilnya adalah : ";

    if (flag == 2)
    {
      cout << "banyak solusi" << endl;
    }
    else if (flag == 3)
    {
        cout << "tidak ada solusi" << endl;
    }

    else
    {
        for (int i = 0; i < n; i++)
        {
            cout << a[i][n] / a[i][i] << " ";
        }

    }
}

int cek(float a[][M], int n, int flag)
{
    int i, j;
```

```
    float hasil;

    flag = 3;

    for (i = 0; i < n; i++)
    {
        hasil = 0;
        for (j = 0; j < n; j++)
        {
            hasil = hasil + a[i][j];
        }
        if (hasil == a[i][j])
        {
            flag = 2;
        }

    }
    return flag;
}

//rumus determinan
int determinan( int matriks[10][10], int n) {
   int det = 0;
   int submatriks[10][10];
   if (n == 2)
   {
       return ((matriks[0][0] * matriks[1][1]) - (matriks[1][0] * matriks[0][1]));
   }
   else
   {
     for (int x = 0; x < n; x++)
```

```
      {
          int subi = 0;

          for (int i = 1; i < n; i++)
          {
                int subj = 0;

            for (int j = 0; j < n; j++)
            {
               if (j == x)
               {
                   continue;
               }

               submatriks[subi][subj] = matriks[i][j];
               subj++;
            }
            subi++;
          }
          det = det + (pow(-1, x) * matriks[0][x] * determinan( submatriks, n - 1 ));
      }
   }
   return det;
}

int main ()
{
             int pilih,pilih2;

             cout<<">>TUGAS BESAR MATRIKS DAN RUANG VEKTOR"<<endl;
```

```
	cout<<"Nama Anggota : "<<endl;
	cout<<"1. Muhammad Asyroful Nur Maulana Yusuf (119140026) "<<endl;
	cout<<"2. Muhammad A.M Hikar Syahrial (119140031)"<<endl;
	cout<<"3. Makruf Alkarhi (119140075)"<<endl<<endl;

//menu
	menu :
	cout<<"Menu: "<<endl;
	cout<<"1. Sistem Persamaan Linier"<<endl;
	cout<<"2. Determinan"<<endl;
	cout<<"3. Matriks Balikan"<<endl;
	cout<<"4. Keluar"<<endl;
	cout<<"Pilih? ";cin>>pilih;

	if (pilih==1){
		cout<<"Menu: "<<endl;
		cout<<"1. Metode Eliminasi Gauss"<<endl;
		cout<<"2. Metode Elimminasi Gauss-Jordan"<<endl;
		cout<<"3. Metode Matriks Balikan (Invers)"<<endl;
		cout<<"4. Kaidah Cramer"<<endl;
		cout<<"Pilih?";cin>>pilih2;

		if (pilih2==1){
			goto gauss;
		}
		else if (pilih2==2){
			goto gaussj;
		}
		else if (pilih2==3){
			goto invers;
		}
```

```
                else if (pilih2==4){
                        goto cramer;
                }
                else {
                        goto menu;
                }
        }
        else if (pilih==2){
                goto determinan;
        }
        else if (pilih==3){
                goto invers;
        }
        else {
                return 0;
        }

//alamat gauss
gauss :
{
    int n,i,j,k;
    cout.precision(4);
    cout.setf(ios::fixed);
    cout<<"\nInput banyak persamaan:\n"; cin>>n;

    float a[n][n+1],x[n];
    cout<<"\nInput nilai augmentented matrik perbaris :\n";
    for (i=0;i<n;i++)
    {
        for (j=0;j<=n;j++)
        {
```

```
        cin>>a[i][j];
    }
}
for (i=0;i<n;i++)
{
    for (k=i+1;k<n;k++)
    {
        if ((a[i][i])<(a[k][i]))
        {
            for (j=0;j<=n;j++)
            {
                float temp=a[i][j];
                a[i][j]=a[k][j];
                a[k][j]=temp;
            }
        }
    }
}
cout<<"\n Kondisi matriks :\n";

for (i=0;i<n;i++)
{
    for (j=0;j<=n;j++)
    {
        cout<<a[i][j]<<setw(16);
    }

    cout<<"\n";
}
for (i=0;i<n-1;i++)
{
```

```
   for (k=i+1;k<n;k++)
   {
       float t=a[k][i]/a[i][i];
       for (j=0;j<=n;j++)
           a[k][j]=a[k][j]-t*a[i][j];
   }
     }

     cout<<"\n\nMatriks Gauss eliminasi:\n";
     for (i=0;i<n;i++)
     {
for (j=0;j<=n;j++)
{
   cout<<a[i][j]<<setw(16);
}
cout<<"\n";
      }
      for (i=n-1;i>=0;i--)
      {
          x[i]=a[i][n];

          for (j=i+1;j<n;j++)
          {
   if (j!=i)
   {
       x[i]=x[i]-a[i][j]*x[j];
   }
  }

  x[i]=x[i]/a[i][i];
}
```

```
            cout<<"\n Hasil Matriks:\n";
            for (i=0;i<n;i++)
    {
        cout<<x[i]<<endl;
    }

    return 0;
}

           //alamat gauss jordan
           gaussj :
           {
              float a[M][M];;
      int n = 3, flag = 0, baris, kolom;

      cout << "input banyak persamaan : "; cin >> baris;
      cout << "input banyak kolom pada augmented matriks : "; cin >> kolom;

      for(int i = 0; i < baris; i++)
      {
         for(int j = 0; j < kolom; j++)
         {
            cin >> a[i][j];
         }
      }

      flag = operasi(a, n);
      if (flag == 1)
      {
         flag = cek(a, n, flag);
```

```
}

cout << "hasil augumented matriks  : " << endl;
cetak_matriks(a, n);
cout << endl;
cetak_hasil(a, n, flag);

return 0;
    }

    //alamat invers
    invers:
    {
int n,i,j,k,l;
float a[20][20];

cout <<endl<<"Masukkan ordo Matriks A (n x n)"<<endl;
cout <<"n : "; cin >>n;
cout <<endl;

for (i=1;i<=n;i++)
{
   for (j=1;j<=n;j++)
   {
      cout <<"A("<<i<<","<<j<<") : ";
      cin >>a[i][j];
   }
}

cout <<endl;
```

```
for (i=1;i<=n;i++)
{
    for (j=1;j<=n;j++)
    {
        cout <<"    "<<a[i][j];
    }

    cout <<endl;
}

for (j=n+1;j<=n+n;j++)
{
    i=j-n;
    a[i][j]=1;
}

for (j=n+1;j<=n+n;j++)
{
    for (i=1;i<=n;i++)
    {
        if (i!=(j-n)) a[i][j]=0;
    }
}

for (i=1;i<=n;i++)
{
    for (j=1;j<=n+n;j++)
    {
        if (i!=j)
        {
            a[i][j]=a[i][j]/a[i][i];
```

```
        }
    }

    for (j=1;j<=n+n;j++)
    {
        if (i==j)
        {

        }
    }
    for (l=1;l<=n;l++)
    {
        if (i!=l)
        {
            for (j=i+1;j<=n+n;j++)
            {
                a[l][j]=a[l][j]-(a[i][j]*a[l][i]);
            }
        }
    }

    for (k=1;k<=n;k++)
    {
        if (i!=k)
        {
            a[k][i]=0;
        }
    }
}

cout <<"Maka invers dari matriks A adalah : "<<endl;
```

```
cout.precision(4);
cout.setf(ios::fixed);

for (i=1;i<=n;i++)
{
   for (j=n+1;j<=n+n;j++)
   {
      cout <<"    "<<a[i][j];
   }
   cout <<endl;
}
   system("PAUSE");
   return EXIT_SUCCESS;
    }


    // alamat determinan
    determinan :
    {
int n, i, j;
int matriks[10][10];
cout << "Input ukuran Matriks:\n"; cin >> n;
cout << "Isi Elemen Matriks:\n";

for (i = 0; i < n; i++)
{
   for (j = 0; j < n; j++)
   {
      cin >> matriks[i][j];
   }
}
```

```
    cout<<"Nilai Matriks:"<<endl;

    for (i = 0; i < n; i++)
    {
        for (j = 0; j < n; j++)
        {
            cout << matriks[i][j] <<" ";
        }

        cout<<endl;
    }

    cout<<"Hasil Determinan Matriks "<< determinan(matriks, n);

    return 0;
     }

     //rumus cramer
     cramer :
{
    cout<<"Masuk matriks 3x3 perbaris\n";
    cout<<"Berikan spasi setiap imputan \n";
    cout<<"\nContoh:\n\n";
    cout<<"  3  4  5    6 -> 6 Angka bagian kanan\n";
    cout<<"  2  7  5    3 \n";
    cout<<"  1  3  4    2 \n";

    bool sahi=true;
    while(sahi)
    {
        cout<<"Masukan...\n";
```

```
        int matriks[3][3];
        int matriks1[3][1];
        int reserve[3][3];
        int det00, detr[3], sp1=0,cont=0, teen=1;
        char in;
        input(matriks, matriks1);
        eksekusi_salin(reserve, matriks);
        det00=determinan(matriks);
        while(sp1<3)
        {
            detr[cont]=salin(matriks, matriks1, sp1);
            eksekusi_salin(matriks,reserve);
            cont++;
            sp1++;
        }
        cont=0;
        while(cont<3)
        {
            cout<<"x"<<teen<<" = "<<  detr[cont]<<" /"<<det00<<endl;
            cont++;
            teen++;
        }
        cout<<"Ketik n ketika mau keluar\n";
        cin>>in;
        if(in=='n' || in=='N')
        {
            return 1;
        }
    }
    return 0;
}
```

```
}

void input(int array[][3], int array1[][1])
{
    int baris=0, kolom=0, x=0;

    while(baris<3)
    {
        kolom=0;
        while(kolom<3)
        {
            cin>>array[baris][kolom];
            kolom++;
        }
        cin>>array1[x][0];
        x++;
        baris++;
    }
}

int determinan(int array[][3])
{
    int baris=1, kolom=1, z=0, temp=0, cont=1, x=0;

    while(x<3)
    {
        temp=temp+cont*(array[0][x]*hitung(array,baris, kolom, z));
        kolom=kolom*0;
        z=z+cont;
        cont=cont*-1;
        x++;
```

```
    }

    cout<<"\nDeterminan Matriks : "<<temp<<"\n\n";

    return temp;
}

int hitung(int array[][3], int a, int b, int c)
{
    int temp1;

    temp1=(array[a][b]*array[a+1][b+1+c])-(array[a+1][b]*array[a][b+1+c]);

    return temp1;
}

int salin(int array[][3], int array1[][1], int a)
{
    int kolom=0, temp;

    while(kolom<3)
    {
        array[kolom][a]=array1[kolom][0];
        kolom++;
    }

    int i=0, j=0;

    while(i<3)
    {
        j=0;
```

```
        while(j<3)
        {
            cout<<array[i][j]<<"  ";
            j++;
        }
        cout<<endl;
        i++;
    }

    temp=determinan(array);

    return temp;
}

void eksekusi_salin(int array[][3], int array1[][3])
{
    int baris=0, kolom=0;

    while(baris<3)
    {
        kolom=0;
        while(kolom<3)
        {
            array[baris][kolom]=array1[baris][kolom];
            kolom++;
        }
        baris++;
    }
}
```

B. Definisi (Atribut dan Method)

Pada program yang sudah kami buat, kami menggunakan beberapa metode dalam c++ antara lain :

1. Sub Program

   Subprogram merupakan program bagian dengan blok terpisah dan didalam program utama, dan akan dipanggil pada program utama jika subprogram itu diperlukan untuk dijalankan. atau bisa juga diartikan blok program yang merupakan sekumpulan instruksi yang bertujuan untuk menyelesaikan suatu tugas khusus. Sebuah sub progam dibuat untuk membantu mengerjakan tugas yang kompleks secara efektif dan efisien.

2. Percabangan

   Percabangan adalah cara yang digunakan untuk mengambil keputusan apabila di dalam program dihadapkan pada kondisi tertentu. Jumlah kondisinya bisa satu, dua atau lebih.Percabangan mengevaluasi kondisi atau ekspresi yang hasilnya benar atau salah. Kondisi atau ekspresi tersebut disebut ekspresi boolean. Hasil dari pengecekan kondisi adalah True atau False. Bila benar (True), maka pernyataan yang ada di dalam blok kondisi tersebut akan dieksekusi. Bila salah (False), maka blok pernyataan lain yang dieksekusi.

3. Looping

   Looping Berasal Dari Kata Loop Artinya Lingkaran, memiliki ciri melingkar atau kembali atau mengulang. Looping artinya perulangan. Adapun looping berarti suatu instruksi untuk membentuk aksi kerja secara berulang pada blok yang sama, dimana didalam blok yang akan di ulang ada terdapat statement atau pernyataan.

   Didalam looping, ada 3 bagian yang mutlak harus ada agar terjadi perulangan yang diinginkan yang berlaku pada 3 perulangan yang sering digunakan yaitu For, Do While, While yaitu ;

   a. Inisialisasi : adalah untuk menyatakan keadaan awal dari variable control pada perulangan .
   b. Kondisi (Ekspresi Perbandingan) : adalah merupakan ekspresi yang berguna untuk mengakhiri perulangan.
   c. Iterasi : adalah bagian untuk mengatur/perubahan nilai dari variable control, iteriasi bisa berupa increment (nilai dari inisialisai bertambah/menaik) atau Decrement (nilai dari inisialisasi berkurang/menurun).

4. Rekursif

   Fungsi rekursif merupakan fungsi yang memanggil dirinya sendiri. Terdapat dua komponen penting dalam fungsi rekursif, yaitu kondisi kapan berhentinya fungsi dan pengurangan atau pembagian data ketika fungsi memanggil dirinya sendiri.

5. Array Multidimensi

Array multidimensi merupakan sebuah variabel yang menyimpan sekumpulan data yang memiliki tipe sama dan elemen yang akan diakses melalui banyak indeks atau subskrip. Array seperti ini biasa digunakan untuk matik, array 2 dimensi juga termasuk kedalam array multidimensi. Array dua dimensi biasanya digunakan untuk merepresentasikan nilai dari sebuah tabel. mengidentifikasi tiap elemen array harus dispesifikasikan nilai baris dan kolom. Array multidimensi sebenarnya adalah array dari array.Deklarasi array multidimensi dilakukan dengan adanya lebih dari satu pasangan kurung siku di dalam deklarasi array.

C. Garis Besar

Pada program tersebut juga dapat kita tarik garis besar bahwa program tersebut memang dibuat untuk menyelesaikan permasalahan terkait materi yang bersifat matriks seperti metode eliminasi Gauss, metode eliminasi Gauss-Jordan, determinan, matriks balikan, matriks kofaktor, matriks adjoin, kaidah Cramer. Sehingga program tersebut sangat mengutamakan konsep pemrograman yaitu, array multidimensi. Dimana konsep tersebut secara umum array multidimensi merujuk pada array yang mempunyai ukuran lebih dari dua subskrip. maka bentuk pendeklarasian array multidimensi sama saja dengan pendeklarasi array satu dimensi maupun dua dimensi.

Array Multidimensi merupakan array yang serupa dengan array satu dimensi maupun array dua dimensi, namun array multidimensi dapat memiliki memori yang lebih besar. Biasanya array multidimensi digunakan untuk menyebut array dengan dimensi lebih dari dua atau array yang mempunyai lebih dari dua subskrip, seperti untuk menyebut array tiga dimensi, empat dimensi, lima dimensi dan seterusnya.

BAB IV

A. Output Program

Berikut adalah hasil eksekusi dari program yang sudah kami buat, antara lain sebagai berikut :

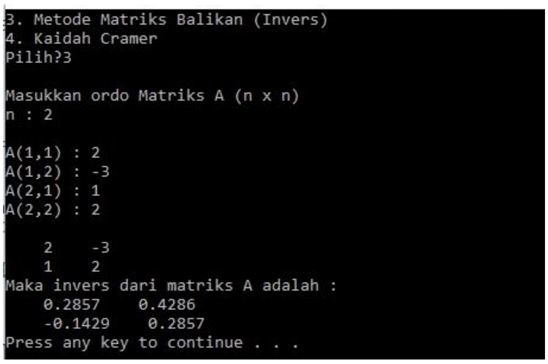
1. Sistem Persamaan Linier
   a. Metode Eliminasi Gauss

```
Menu:
1. Sistem Persamaan Linier
2. Determinan
3. Matriks Balikan
4. Keluar
Pilih? 1
Menu:
1. Metode Eliminasi Gauss
2. Metode Elimminasi Gauss-Jordan
3. Metode Matriks Balikan (Invers)
4. Kaidah Cramer
Pilih?1

Input banyak persamaan:
4

Input nilai augmentented matrik perbaris :
1 2 -1 -1 1
2 5 -7 -5 -2
2 -1 1 3 4
5 2 -4 2 6

 Kondisi matriks :
5.0000          2.0000          -4.0000          2.0000          6.0000
2.0000          5.0000          -7.0000         -5.0000         -2.0000
2.0000         -1.0000           1.0000          3.0000          4.0000
1.0000          2.0000          -1.0000         -1.0000          1.0000


Matriks Gauss eliminasi:
5.0000          2.0000          -4.0000          2.0000          6.0000
-0.0000          4.2000          -5.4000         -5.8000         -4.4000
-0.0000         -0.0000           0.2857         -0.2857         -0.2857
0.0000          0.0000           0.0000          2.6667          3.3333

 Hasil Matriks:
0.5000
1.0000
0.2500
1.2500

Process returned 0 (0x0)   execution time : 109.153 s
```

   b. Metode Eliminasi Gauss Jordan

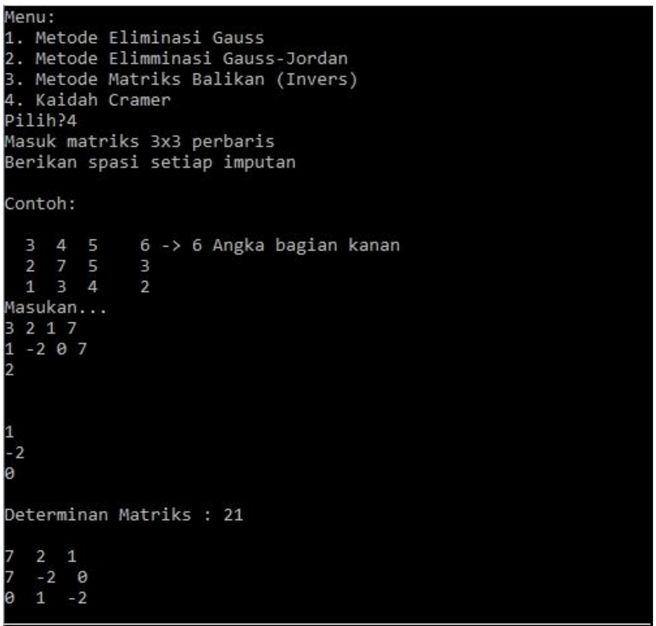
```
Nama Anggota :
1. Muhammad Asyroful Nur Maulana Yusuf (119140026)
2. Muhammad A.M Hikar Syahrial (119140031)
3. Makruf Alkarhi (119140075)

Menu:
1. Sistem Persamaan Linier
2. Determinan
3. Matriks Balikan
4. Keluar
Pilih? 1
Menu:
1. Metode Eliminasi Gauss
2. Metode Elimminasi Gauss-Jordan
3. Metode Matriks Balikan (Invers)
4. Kaidah Cramer
Pilih?2
input banyak persamaan : 3
input banyak kolom pada augmented matriks : 7
0 1 0 0 1 0 2
0 0 0 1 1 0 -1
0 1 0 0 0 1 1
hasil augumented matriks  :
0 1 0 0
0 0 0 1
0 1 0 0

hasilnya adalah : tidak ada solusi

Process returned 0 (0x0)   execution time : 90.092 s
```

c. Metode Matriks Balikan

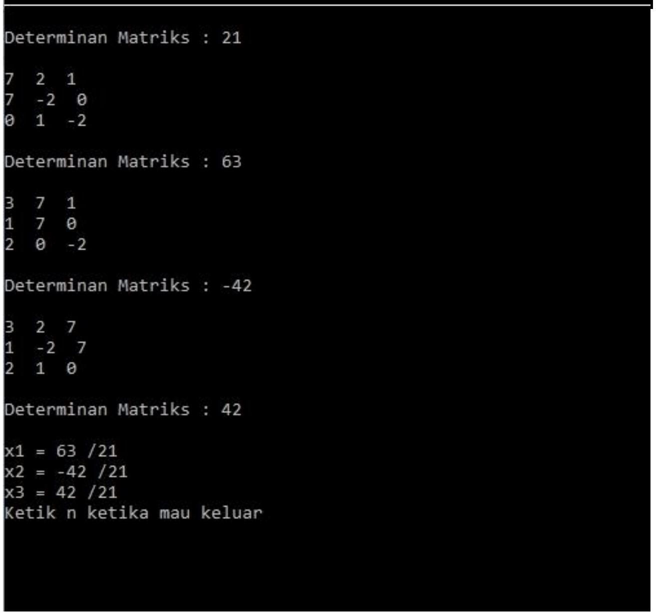


```
3. Metode Matriks Balikan (Invers)
4. Kaidah Cramer
Pilih?3

Masukkan ordo Matriks A (n x n)
n : 2

A(1,1) : 2
A(1,2) : -3
A(2,1) : 1
A(2,2) : 2

    2    -3
    1    2
Maka invers dari matriks A adalah :
    0.2857    0.4286
    -0.1429    0.2857
Press any key to continue . . .
```

d. Kaidah Cramer

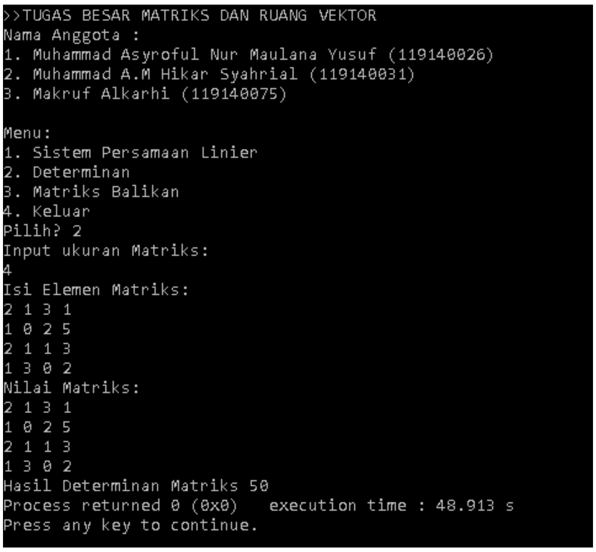


```
Menu:
1. Metode Eliminasi Gauss
2. Metode Elimminasi Gauss-Jordan
3. Metode Matriks Balikan (Invers)
4. Kaidah Cramer
Pilih?4
Masuk matriks 3x3 perbaris
Berikan spasi setiap imputan

Contoh:

  3  4  5    6 -> 6 Angka bagian kanan
  2  7  5    3
  1  3  4    2
Masukan...
3 2 1 7
1 -2 0 7
2



1
-2
0

Determinan Matriks : 21

7  2  1
7  -2  0
0  1  -2
```

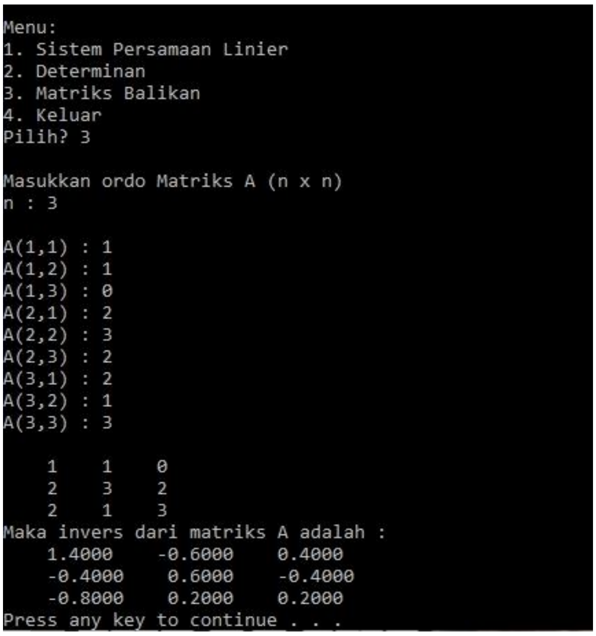


```
Determinan Matriks : 21

7  2  1
7  -2  0
0  1  -2

Determinan Matriks : 63

3  7  1
1  7  0
2  0  -2

Determinan Matriks : -42

3  2  7
1  -2  7
2  1  0

Determinan Matriks : 42

x1 = 63 /21
x2 = -42 /21
x3 = 42 /21
Ketik n ketika mau keluar
```

2. Determinan



3. Matriks Balikan



B. Analisis

Pada program kali ini sesuai dengan judul yaitu membuat aplikasi matriks, tentu bagian utama dari pembuatan program adalah dengan konsep array multidimensi. Berikut adalah analisis dari setiap menu :

1. Sistem Persamaan Linier

Pada bagian sistem persamaan linier bagian eliminasi gauss dapat dilihat bahwa program berjalan sesuai dengan yang diharapkan atau program berjalan dengan baik. di bagian eliminasi gauss juga kita menggunakan konsep penukaran dengan bantuan variabel temp. Kemudian pada bagian eliminasi gauss jordan juga berjalan sesuai dengan harapan. di bagian ini kita menggunakan fungsi sub program, yaitu cetak matriks dan juga cetak hasil supaya jika terjadi kesalahan lebih mudah untuk mengubahnya. Kemudian pada bagian matriks balikan (invers) program berjalan sesuai dengan harapan tetapi program hanya mencari nilai matriks balikan tetapi tidak memberikan nilai variabelnya. di bagian ini kita menggunakan cout.precision(4); untuk hasil yang desimal tak hingga seperti ⅓ = 0.3333… kita menggunakan itu agar desimal yang tersimpan di belakang koma hanya 4 saja. Kemudian pada bagian kaidah cramer program berjalan sesuai dengan harapan. dibagian ini kita juga menggunakan sub program yang bernama input, eksekusi_salin, determinan. supaya jika terjadi kesalahan lebih mudah untuk mengubahnya.

2. Determinan

   Pada bagian ini program berjalan sesuai dengan harapan dan juga menghasilkan output yang sesuai, di bagian ini kita menggunakan konsep rekursif yang dimana det = det + (pow(-1, x) * matriks[0][x] * determinan(submatriks, n - 1)); akan terus memanggil dirinya sendiri hingga tepat pada basisnya yaitu dimana if (n == 2) atau ketika nilai n menjadi 0 program akan me-return ((matriks[0][0] * matriks[1][1]) - (matriks[1][0] * matriks[0][1]));

3. Matriks Balikan

   Pada bagian ini program juga berjalan sesuai dengan harapan dan juga menghasilkan output yang sesuai, di bagian matriks balikan (invers) program berjalan sesuai dengan harapan tetapi program hanya mencari nilai matriks balikan tetapi tidak memberikan nilai variabelnya. di bagian ini kita menggunakan cout.precision(4); untuk hasil yang desimal tak hingga seperti ⅓ = 0.3333… kita menggunakan itu agar desimal yang tersimpan di belakang koma hanya 4 saja.

BAB V

A. Kesimpulan

Berdasarkan dari program yang sudah kami buat mengenai Matriks dan Ruang Vektor Sistem Persamaan Linier, Determinan, dan Aplikasinya, maka dapat disimpulkan bahwa :

1. Metode eliminasi Gauss, metode eliminasi Gauss-Jordan, determinan, matriks balikan, matriks kofaktor, matriks adjoin, dan kaidah Cramer dapat dihitung dengan pemrograman bahasa C++.
2. Semua program berjalan sesuai dengan metode yang diberikan tetapi pada SPL di bagian matriks balikan (invers) program hanya mencari nilai matriks balikan dan tidak memberikan nilai dari setiap variabelnya.
3. Melalui program ini bisa dikatakan bahwa cara ini lebih efisien dan tingkat keakuratan lebih tinggi dibandingkan dengan menghitung manual.

B. Saran

Adapun saran yang kami berikan sebagai berikut :

1. Sebaiknya memahami materi tentang metode eliminasi Gauss, metode eliminasi Gauss-Jordan, determinan, matriks balikan, matriks kofaktor, matriks adjoin, dan kaidah Cramer sebelum membuat programnya.
2. Lebih bersabar dan butuh ketelitian dalam membuat program.

Daftar Pustaka

Anton, H. (2010). Elementary Linear Algebra 10th Edition. John Wiley & Sons Ltd.

Martono, T. (2017). Aljabar Matriks untuk Metode Kuantitatif. PT Penerbit IPB Press.

Sasongko, S. B. (2010). Metode Numerik dengan Scilab. ANDI OFFSET.

Yohanes, W. (2020, Oktober Sabtu). *Invers Matriks Kelas 12*. Diambil kembali dari Quipper.com: https://www.quipper.com/id/blog/mapel/matematika/invers-matriks-kelas-12/#:~:text=Invers%20matriks%20adalah%20sebuah%20kebalikan,pangkat%20%2D1%20di%20atas%20hurufnya