

Zulkarnain
Desty Septianawati
Havizul



MODUL PEMBELAJARAN ALJABAR LINIER DENGAN APLIKASI MATLAB

MODUL PEMBELAJARAN ALJABAR LINEAR DENGAN APLIKASI MATLAB

Copyright © 2022

Penulis

Zulkarnain, Desty Septianawati, Havizul

Desain Sampul

Muzammil Akbar

Penyunting Naskah

Siti Shofiyatus Sa'diyah

Gambar Ilustrasi : Freepik



Penerbit

Insight Mediatama

Watesnegoro No. 4 (61385) Mojokerto

Whatsapp 081234880343

www.insightmediatama.store

MODUL PEMBELAJARAN ALJABAR LINEAR
DENGAN APLIKASI MATLAB

Penulis: Zulkarnain, Desty Septianawati, Havizul

Mojokerto – C.V. Insight Mediatama, 2022
v + 70 hlm : 21 x 29,7 cm

Cetakan Ke – 1, Agustus 2022

ISBN : 978-623-5451-34-3

© **All Rights Reserved** Ketentuan Pidana Pasal 112-119 Undang-undang Nomor 28 Tahun 2014 Tentang Hak Cipta. Dilarang keras menerjemahkan, memfotokopi, atau memperbanyak sebagian atau seluruh isi buku ini tanpa izin tertulis dari penulis dan penerbit

KATA PENGANTAR

Assallam mualaikum wr.wb

Alhamdulillahirrobbilalamin, penulis panjatkan puji syukur kehadiran Allah SWT, atas rahmat serta hidayah-Nya dapat menyelesaikan penyusunan modul yang berjudul “Modul Pembelajaran Aljabar Linear dengan Aplikasi Matlab” ini dengan baik. Shalawat dan salam semoga terlimpahkan kepada Nabi Muhammad SAW. Atas terselesaikannya penyusunan modul ini penulis menyampaikan ucapan terima kasih yang tidak terhingga, khususnya kepada kedua orang tua yang memberikan semangat, nasihat dan dorongan dalam pembuatan modul ini, teman sejawat dan para editor yang telah berkontribusi dalam penyusunan serta semua pihak yang telah membantu dan berperan serta dalam penyelesaian modul ini.

Modul ini disusun agar dapat memperluas ilmu pembaca tentang materi aljabar linear dengan aplikasi matlab yang penulis sajikan. Semoga buku ini dapat memberikan manfaat dan menjadi sumbangan pemikiran kepada pembaca khususnya para mahasiswa. Penulis menyadari dalam penyusunan modul ini masih banyak kekurangan dan jauh dari sempurna, baik pada teknis penulisan maupun materinya, mengingat akan kemampuan yang dimiliki oleh penulis. Pandangan dan saran juga diperlukan oleh penulis guna memperbaiki penulisan pada edisi selanjutnya agar modul ini dapat disempurnakan dengan baik.

Pontianak, Juli 2022

Zulkarnain
Desty Septianawati
Havizul

DAFTAR ISI

KATA PENGANTAR.....	iii
DAFTAR ISI.....	iv
DAFTAR GAMBAR.....	v
OPERASI MatriKS	1
1. Penjumlahan Matriks	1
2. Pengurangan Matriks	10
3. Perkalian Matriks	16
4. Transpose Matriks	22
5. Determinan Matriks	26
a. Metode Biasa	26
b. Determinan Metode Chio	28
c. Determinan Metode Crout	33
d. Metode Doolittle	39
6. Invers Suatu Matriks	46
SISTEM PERSAMAAN LINEAR	55
1. Metode Eliminasi Gauss	55
2. Metode Eliminasi Gauss Jordan	60
3. Aturan Cramer	64
DAFTAR PUSTAKA.....	70

DAFTAR GAMBAR

Gambar 1. <i>Flowchart</i> Program Operasi Penjumlahan Matriks 3x3	1
Gambar 2. <i>Flowchart</i> Program Operasi Penjumlahan Matriks 3x3 v2 (Interaksi User)	3
Gambar 3. <i>Flowchart</i> Program Operasi Penjumlahan Matriks 3x3 v3 (Array & Looping For)	5
Gambar 4 <i>Flowchart</i> Program Operasi Penjumlahan Matriks mxn	7
Gambar 5. <i>Flowchart</i> Program Operasi Penjumlahan Matriks mxn Di Matlab	9
Gambar 6. <i>Flowchart</i> Program Operasi Pengurangan Matriks 3x3.....	11
Gambar 7. <i>Flowchart</i> Program Operasi Pengurangan Matriks MxN	13
Gambar 8. <i>Flowchart</i> Program Operasi Pengurangan Matriks MxN	15
Gambar 9. <i>Flowchart</i> Program Operasi Perkalian Matriks MxN Versi 1	17
Gambar 10. <i>Flowchart</i> Program Operasi Perkalian Matriks MxN Versi 2	20
Gambar 11. <i>Flowchart Program</i> Operasi Transpose Matriks MxN Versi 1	23
Gambar 12. <i>Flowchart</i> Program Operasi Transpose Matriks Ordo MxN Versi 2.....	25
Gambar 13. <i>Flowchart</i> Program Matlab Operasi Perhitungan Determinan Matriks Ordo MxN Metode Chio	31

OPERASI MATRIKS

1. Penjumlahan matriks

Jika A dan B adalah sembarang matriks ber-ordo sama maka jumlah matriks A dan B (ditulis $A + B$) adalah matriks yang diperoleh dengan menjumlahkan setiap elemen matriks A dengan elemen matriks yang seletak. Matriks – matriks yang ordonya berbeda tidak dapat dijumlahkan.

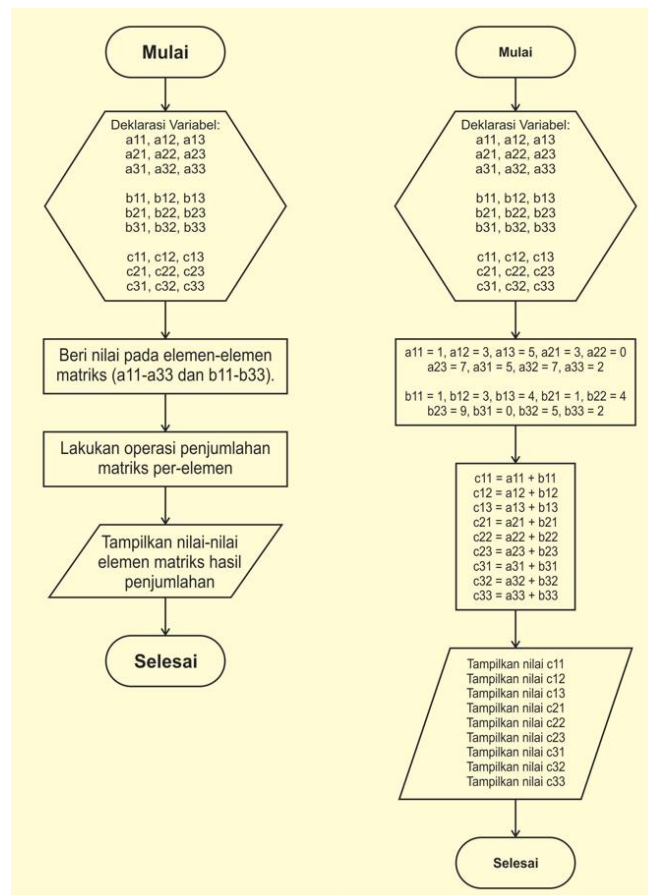
Contoh:

$$A = \begin{pmatrix} 1 & 3 & 5 \\ 3 & 0 & 7 \\ 5 & 7 & 2 \end{pmatrix} \quad B = \begin{pmatrix} 1 & 3 & 4 \\ 1 & 4 & 9 \\ 0 & 5 & 2 \end{pmatrix}$$

$$C = A + B = \begin{pmatrix} 1+1 & 3+3 & 5+4 \\ 3+1 & 0+4 & 7+9 \\ 5+0 & 7+5 & 2+2 \end{pmatrix}$$

$$= \begin{pmatrix} 2 & 6 & 9 \\ 4 & 4 & 16 \\ 5 & 12 & 4 \end{pmatrix}$$

Flowchart Program Operasi Penjumlahan Matriks Ordo 3x3 Versi 1



Gambar 1. Flowchart Program Operasi Penjumlahan Matriks 3x3

Kode Program Matlab Untuk Operasi Penjumlahan Matriks Ordo 3x3 Versi 1:

```
clc

a11 = 1;
a12 = 3;
a13 = 5;
a21 = 3;
a22 = 0;
a23 = 7;
a31 = 5;
a32 = 7;
a33 = 2;

b11 = 1;
b12 = 3;
b13 = 4;
b21 = 1;
b22 = 4;
b23 = 9;
b31 = 0;
b32 = 5;
b33 = 2;

c11 = a11 + b11;
c12 = a12 + b12;
c13 = a13 + b13;
c21 = a21 + b21;
c22 = a22 + b22;
c23 = a23 + b23;
c31 = a31 + b31;
c32 = a32 + b32;
c33 = a33 + b33;

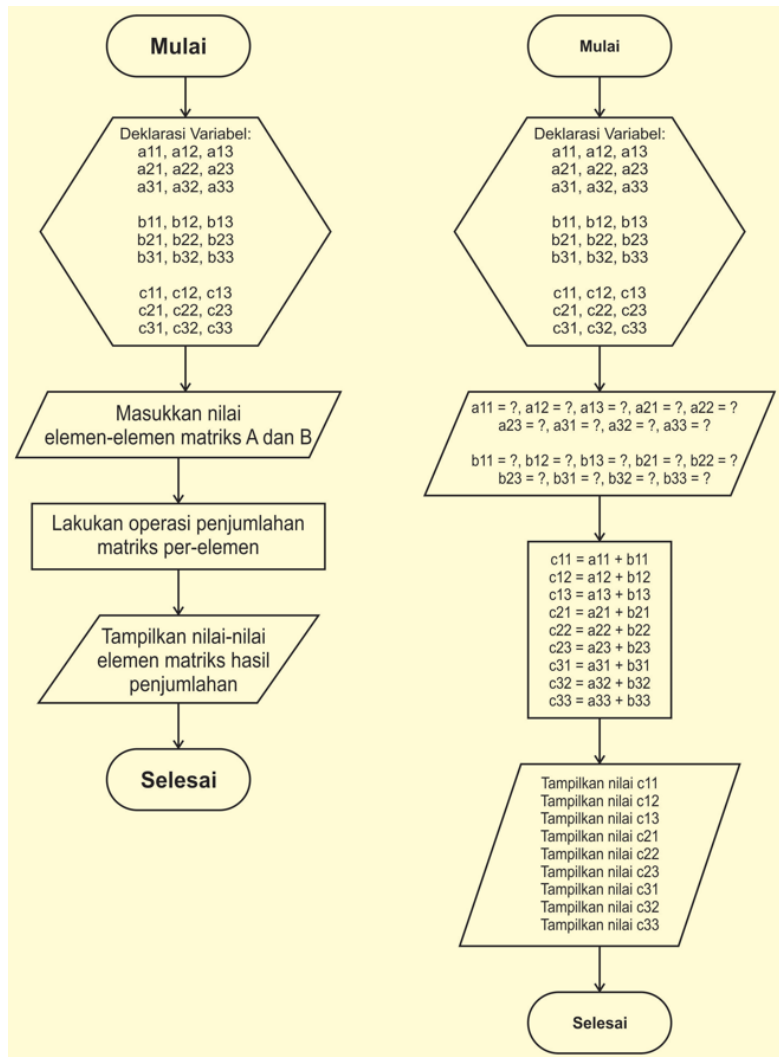
fprintf('c11 = %d', c11)
disp(' ')
fprintf('c12 = %d', c12)
disp(' ')
fprintf('c13 = %d', c13)
disp(' ')
fprintf('c21 = %d', c21)
disp(' ')
fprintf('c22 = %d', c22)
disp(' ')
fprintf('c23 = %d', c23)
disp(' ')
fprintf('c31 = %d', c31)
disp(' ')
fprintf('c32 = %d', c32)
disp(' ')
fprintf('c33 = %d', c33)
disp(' ')
```

Tampilan Output Program Operasi Penjumlahan Matriks Ordo 3x3 Versi 1

```
c11 = 2
c12 = 6
c13 = 9
c21 = 4
```


$$\begin{aligned} c_{22} &= 4 \\ c_{23} &= 16 \\ c_{31} &= 5 \\ c_{32} &= 12 \\ c_{33} &= 4 \end{aligned}$$

Flowchart Program Operasi Penjumlahan Matriks Ordo 3x3 Versi 2 (Interaksi User)



Gambar 2. Flowchart Program Operasi Penjumlahan Matriks 3x3 v2 (Interaksi User)

Kode Program Matlab Operasi Penjumlahan Matriks Ordo 3x3 Versi 2

```
clc

%input nilai matriks A
a11 = input('Masukkan nilai a11: ');
a12 = input('Masukkan nilai a12: ');
a13 = input('Masukkan nilai a13: ');
a21 = input('Masukkan nilai a21: ');
a22 = input('Masukkan nilai a22: ');
a23 = input('Masukkan nilai a23: ');
a31 = input('Masukkan nilai a31: ');
```

```

a32 = input('Masukkan nilai a32: ');
a33 = input('Masukkan nilai a33: ');
disp(' ')

%input nilai matriks B
b11 = input('Masukkan nilai b11: ');
b12 = input('Masukkan nilai b12: ');
b13 = input('Masukkan nilai b13: ');
b21 = input('Masukkan nilai b21: ');
b22 = input('Masukkan nilai b22: ');
b23 = input('Masukkan nilai b23: ');
b31 = input('Masukkan nilai b31: ');
b32 = input('Masukkan nilai b32: ');
b33 = input('Masukkan nilai b33: ');

%operasi penjumlahan, A + B
ab11 = a11 + b11;
ab12 = a12 + b12;
ab13 = a13 + b13;
ab21 = a21 + b21;
ab22 = a22 + b22;
ab23 = a23 + b23;
ab31 = a31 + b31;
ab32 = a32 + b32;
ab33 = a33 + b33;

%Tampilkan nilai hasil pengurangan Matriks A dengan Matriks B
disp(' ')
disp('Hasil dari operasi A+B: ')
fprintf('ab11 = %d', ab11)
disp(' ')
fprintf('ab12 = %d', ab12)
disp(' ')
fprintf('ab13 = %d', ab13)
disp(' ')
fprintf('ab21 = %d', ab21)
disp(' ')
fprintf('ab22 = %d', ab22)
disp(' ')
fprintf('ab23 = %d', ab23)
disp(' ')
fprintf('ab31 = %d', ab31)
disp(' ')
fprintf('ab32 = %d', ab32)
disp(' ')
fprintf('ab33 = %d', ab33)
disp(' ')

```

Tampilan *Output* Program Operasi Penjumlahan Matriks Ordo 3x3 Versi 2

```

Masukkan nilai a11: 1
Masukkan nilai a12: 3
Masukkan nilai a13: 5
Masukkan nilai a21: 3
Masukkan nilai a22: 0
Masukkan nilai a23: 7
Masukkan nilai a31: 5
Masukkan nilai a32: 7
Masukkan nilai a33: 2

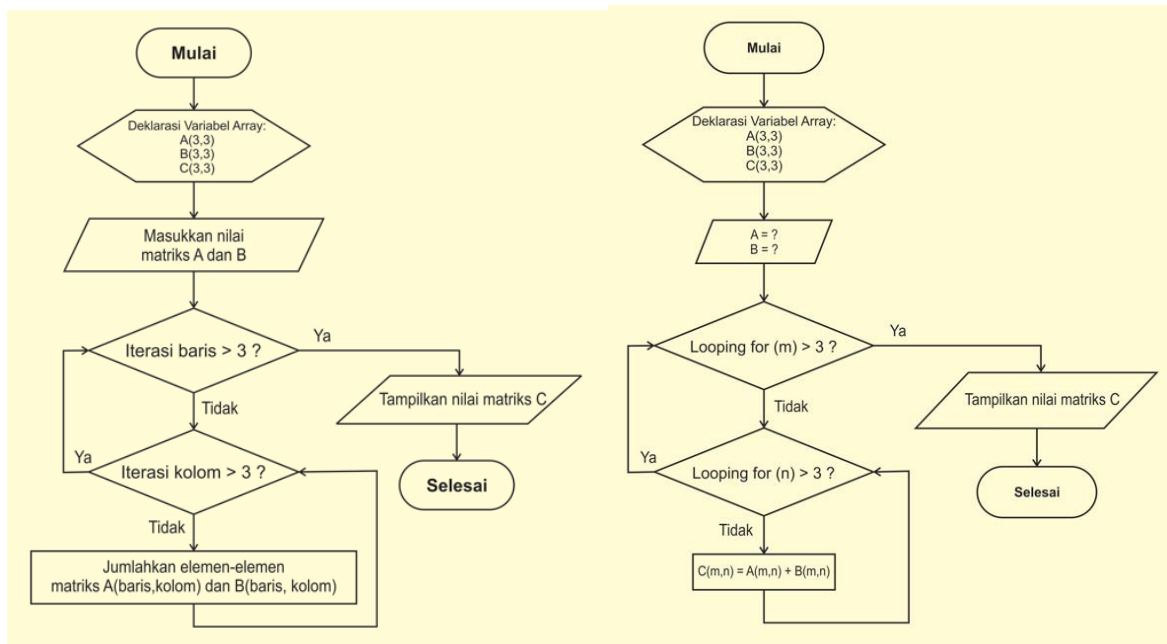
```

Masukkan nilai b11: 1
 Masukkan nilai b12: 3
 Masukkan nilai b13: 4
 Masukkan nilai b21: 1
 Masukkan nilai b22: 4
 Masukkan nilai b23: 9
 Masukkan nilai b31: 0
 Masukkan nilai b32: 5
 Masukkan nilai b33: 2

Hasil dari operasi A+B:

ab11 = 2
 ab12 = 6
 ab13 = 9
 ab21 = 4
 ab22 = 4
 ab23 = 16
 ab31 = 5
 ab32 = 12
 ab33 = 4

Flowchart Program Operasi Penjumlahan Matriks Ordo 3x3 Versi 3 (Array & Looping For)



Gambar 3. Flowchart Program Operasi Penjumlahan Matriks 3x3 v3 (Array & Looping For)

Kode Program Matlab Operasi Penjumlahan Matriks Ordo 3x3 Versi 3

```
clc

disp('Operasi Penjumlahan Terhadap Matriks Ordo 3x3')
disp('A + B = C')
disp(' ')
disp('Notasi dalam program: ')
disp('A(1,1), A(1,2), A(1,3), ..., A(3,3)')
disp('..., B(3,1), B(3,2), B(3,3)')
disp('C(1,1), C(1,2), C(1,3), ..., C(3,3)')
```

```

disp(' ')
disp('Dibuat oleh: Havizul')
disp('=====')
disp(' ')

%input nilai matriks A
A = input('Nilai matriks A = ')

%input nilai matriks B
B = input('Nilai matriks B = ')

%operasi penjumlahan, A + B
for m=1:3      %iterasi baris 1 hingga 3
    for n=1:3  %iterasi kolom 1 hingga 3
        C(m,n) = A(m,n) + B(m,n);
    end
end

%Tampilkan nilai hasil penjumlahan Matriks A dengan Matriks B
disp(' ')
disp('Hasil dari A + B = C')
disp('C =')
disp(C)
fprintf('Nilai Matriks C = %d', C)
disp(' ')

```

Tampilan Output Program Operasi Penjumlahan Matriks Ordo 3x3 Versi 3

Operasi Penjumlahan Terhadap Matriks Ordo 3x3
 $A + B = C$

Notasi dalam program:

$A(1,1), A(1,2), A(1,3), \dots, A(3,3)$

$\dots, B(3,1), B(3,2), B(3,3)$

$C(1,1), C(1,2), C(1,3), \dots, C(3,3)$

Dibuat oleh: Havizul

=====

Nilai matriks A = [1 3 5; 3 0 7; 5 7 2]

A =

1	3	5
3	0	7
5	7	2

Nilai matriks B = [1 3 4; 1 4 9; 0 5 2]

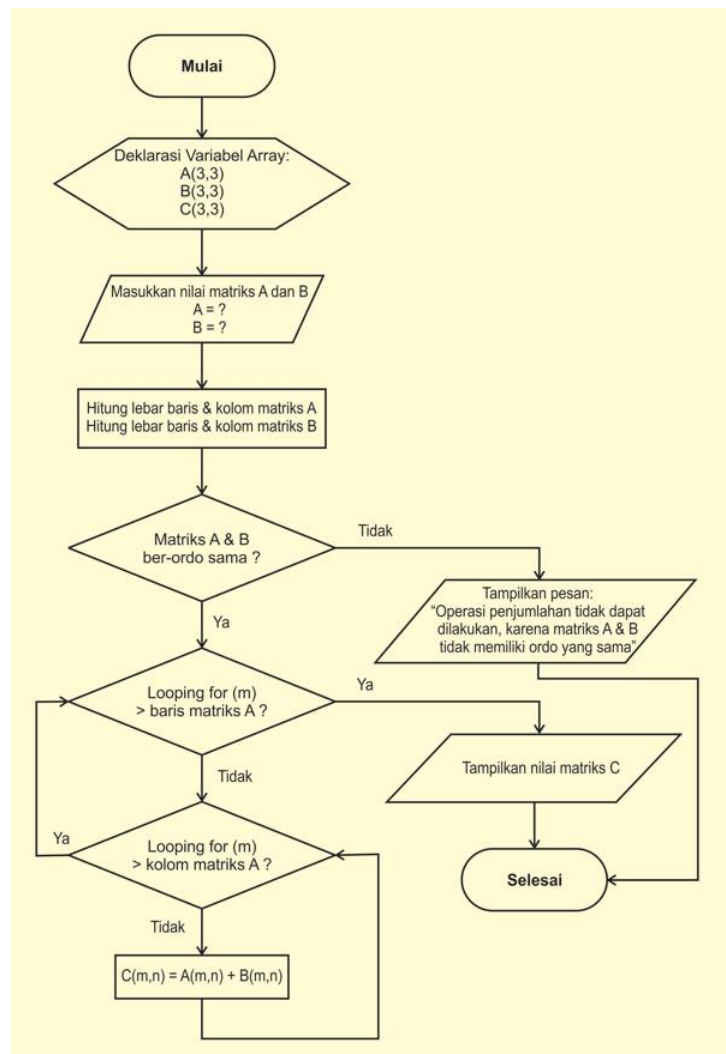
B =

1	3	4
1	4	9
0	5	2

Hasil dari $A + B = C$

C =

2	6	9
4	4	16
5	12	4

Flowchart Program Operasi Penjumlahan Matriks Ordo mxn Versi 4 (Ordo Bebas)

Gambar 4. Flowchart Program Operasi Penjumlahan Matriks mxn

Kode Program Matlab Untuk Operasi Penjumlahan Matriks Ordo MxN Versi 4

```

clc

disp('Operasi Penjumlahan Terhadap Matriks Ordo mxn')
disp('Matriks Square -> m = n')
disp('A + B = C')
disp(' ')
disp('Notasi dalam program: ')
disp('A(1,1), A(1,2), A(1,3), ..., A(3,3)')
disp('..., B(3,1), B(3,2), B(3,3)')
disp('C(1,1), C(1,2), C(1,3), ..., C(3,3)')
disp(' ')
disp('Dibuat oleh: Havizul')
disp('=====')
disp(' ')

%input nilai matriks A
A = input('Nilai matriks A = ')

```

```


%input nilai matriks B
B = input('Nilai matriks B = ')

%Ordo matriks A dan B harus sama
[barisA, kolomA] = size(A);
[barisB, kolomB] = size(B);
if (barisA ~= barisB) || (kolomA ~= kolomB) %-> cek apakah matriks memiliki
ordo yang sama
    disp(' ')
    disp('Matriks A dan B harus memiliki ordo yang sama')
    (' ')
else
    %operasi penjumlahan, A + B
    for i=1:3          %iterasi baris 1 hingga 3
        for j=1:3      %iterasi kolom 1 hingga 3
            C(i,j) = A(i,j) + B(i,j);
        end
    end
end

%Tampilkan nilai hasil penjumlahan Matriks A dengan Matriks B
disp(' ')
disp('Hasil dari A + B = C')
disp('C =')
disp(C)
fprintf('Nilai Matriks C = %d', C)
end

disp(' ')

```

Tampilan *Output* Program Penjumlahan V4 (Jika Matriks A Dan B Memiliki Ordo Yang Setara)

Operasi Penjumlahan Terhadap Matriks Ordo $m \times n$
Matriks Square $\rightarrow m = n$
 $A + B = C$

Notasi dalam program:
 $A(1,1), A(1,2), A(1,3), \dots, A(3,3)$
 $\dots, B(3,1), B(3,2), B(3,3)$
 $C(1,1), C(1,2), C(1,3), \dots, C(3,3)$

Dibuat oleh: Havizul

=====

Nilai matriks A = [1 3 5; 3 0 7; 5 7 2]

A =

1	3	5
3	0	7
5	7	2

Nilai matriks B = [1 3 4; 1 4 9; 0 5 2]

B =

1	3	4
1	4	9
0	5	2

Hasil dari $A + B = C$
 $C =$

2	6	9
4	4	16
5	12	4

Tampilan *Output* Program Penjumlahan V4 (Jika Matriks A Dan B Memiliki Ordo Yang Tidak Setara)

Notasi dalam program:
 $A(1,1), A(1,2), A(1,3), \dots, A(3,3)$
 $\dots, B(3,1), B(3,2), B(3,3)$
 $C(1,1), C(1,2), C(1,3), \dots, C(3,3)$

Dibuat oleh: Havizul

=====

Nilai matriks A = [1 3 5; 3 0 7; 5 7 2]

A =

1	3	5
3	0	7
5	7	2

Nilai matriks B = [1 3; 3 7; 5 7]

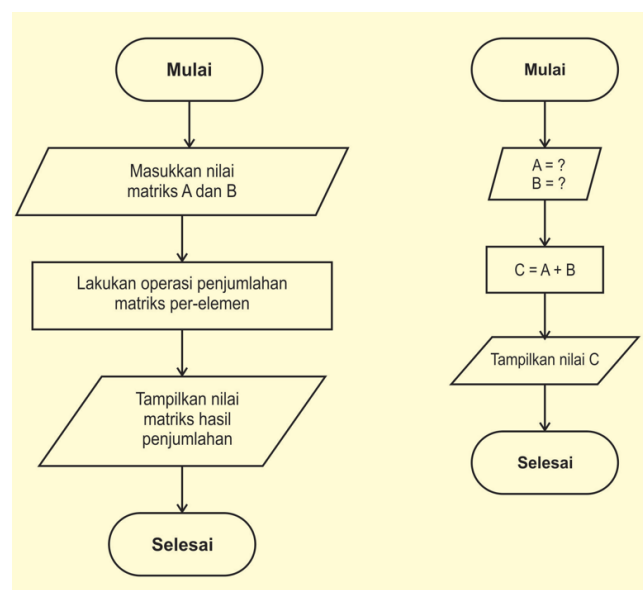
B =

1	3
3	7
5	7

Penjumlahan Matriks tidak dapat dilakukan !
 Matriks A dan B harus memiliki ordo yang sama.

ans =

Flowchart Program Operasi Penjumlahan Matriks Ordo $m \times n$ Versi 5 (Khusus Matlab)



Gambar 5. Flowchart Program Operasi Penjumlahan Matriks $m \times n$ Di Matlab

Kode Program Operasi Penjumlahan Matriks MxN (Khusus Matlab)

```

clc

%Input nilai matriks A dan matriks B
A = [1 3 5; 3 0 7; 5 7 2]
B = [1 3 4; 1 4 9; 0 5 2]

%Operasi penjumlahan matriks A dan matriks B
C = A + B

```

Tampilan Output Program Operasi Penjumlahan Matriks MxN

```

A =
     1     3     5
     3     0     7
     5     7     2

B =
     1     3     4
     1     4     9
     0     5     2

C =
     2     6     9
     4     4    16
     5    12     4

```

2. Pengurangan Matriks

Matriks lawan dari matriks A (ditulis $-A$) adalah matriks yang elemen – elemennya adalah lawan (negative) dari elemen – elemen matriks A yang seletak. Jika A dan B adalah sembarang matriks berordo sama maka pengurangan matriks A dengan matriks B (ditulis $A - B$) adalah matriks yang diperoleh dengan menjumlahkan matriks A dengan lawan matriks B. Jika $A - B = A + (-B)$

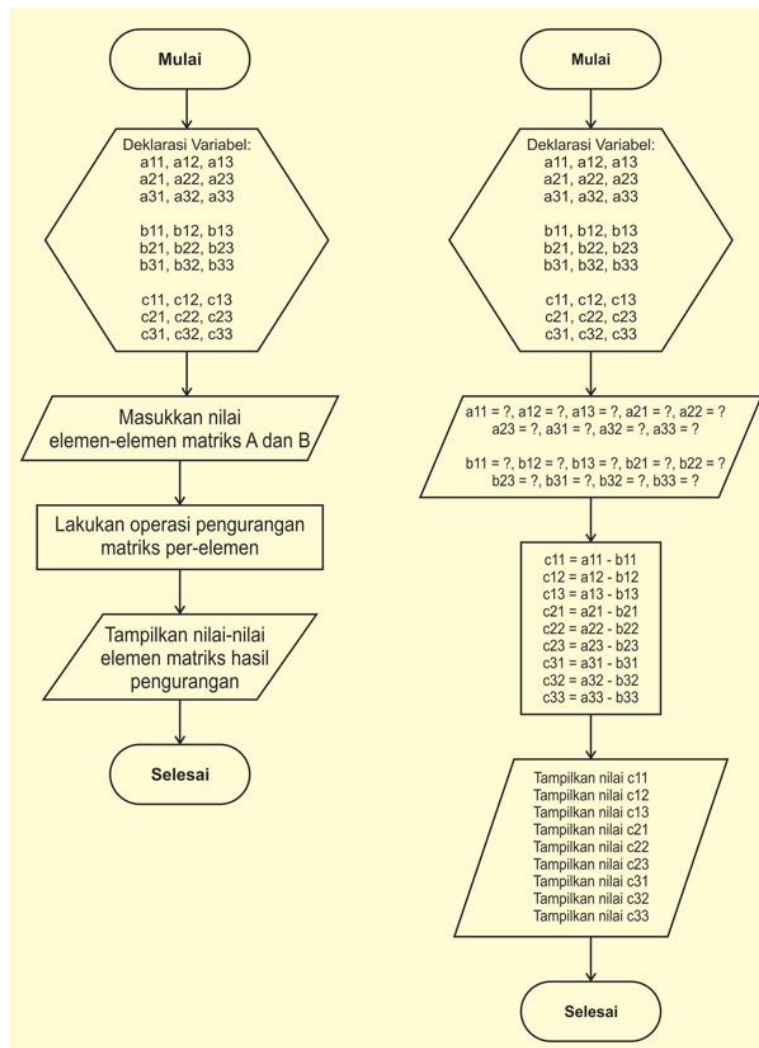
Contoh:

$$A = \begin{pmatrix} 1 & 3 & 5 \\ 3 & 0 & 7 \\ 5 & 7 & 2 \end{pmatrix} \quad B = \begin{pmatrix} 1 & 3 & 4 \\ 1 & 4 & 9 \\ 0 & 5 & 2 \end{pmatrix}$$

$$A - B = A + (-B)$$

$$\begin{aligned}
&= \begin{pmatrix} 1 & 3 & 5 \\ 3 & 0 & 7 \\ 5 & 7 & 2 \end{pmatrix} + \begin{pmatrix} -1 & -3 & -4 \\ -1 & -4 & -9 \\ 0 & -5 & -2 \end{pmatrix} \\
&= \begin{pmatrix} 1 + (-1) & 3 + (-3) & 5 + (-4) \\ 3 + (-1) & 0 + (-4) & 7 + (-9) \\ 5 + 0 & 7 + (-5) & 2 + (-2) \end{pmatrix} \\
&= \begin{pmatrix} 0 & 0 & 1 \\ 2 & -4 & -2 \\ 5 & 2 & 0 \end{pmatrix}
\end{aligned}$$

Flowchart Program Operasi Pengurangan Matriks Ordo 3x3 Versi 1



Gambar 6. Flowchart Program Operasi Pengurangan Matriks 3x3

Kode Program Operasi Pengurangan Matriks Ordo 3x3 Versi 1

```
clc

disp('Operasi Pengurangan Terhadap Matriks Ordo 3x3')
disp(' ')
disp('Dibuat oleh: Havizul')
disp('=====')
disp(' ')

%input nilai matriks A
a11 = input('Masukkan nilai a11: ');
a12 = input('Masukkan nilai a12: ');
a13 = input('Masukkan nilai a13: ');
a21 = input('Masukkan nilai a21: ');
a22 = input('Masukkan nilai a22: ');
a23 = input('Masukkan nilai a23: ');
a31 = input('Masukkan nilai a31: ');
a32 = input('Masukkan nilai a32: ');
a33 = input('Masukkan nilai a33: ');
```

```

%input nilai matriks B
b11 = input('Masukkan nilai b11: ');
b12 = input('Masukkan nilai b12: ');
b13 = input('Masukkan nilai b13: ');
b21 = input('Masukkan nilai b21: ');
b22 = input('Masukkan nilai b22: ');
b23 = input('Masukkan nilai b23: ');
b31 = input('Masukkan nilai b31: ');
b32 = input('Masukkan nilai b32: ');
b33 = input('Masukkan nilai b33: ');

%operasi pengurangan, A - B = A + (-B)
ab11 = a11 + (-b11);
ab12 = a12 + (-b12);
ab13 = a13 + (-b13);
ab21 = a21 + (-b21);
ab22 = a22 + (-b22);
ab23 = a23 + (-b23);
ab31 = a31 + (-b31);
ab32 = a32 + (-b32);
ab33 = a33 + (-b33);

%Tampilkan nilai hasil pengurangan Matriks A dengan Matriks B
disp(' ')
disp('Hasil dari operasi A-B: ')
disp(' ')
fprintf('ab11 = %f', ab11)
disp(' ')
fprintf('ab12 = %f', ab12)
disp(' ')
fprintf('ab13 = %f', ab13)
disp(' ')
fprintf('ab21 = %f', ab21)
disp(' ')
fprintf('ab22 = %f', ab22)
disp(' ')
fprintf('ab23 = %f', ab23)
disp(' ')
fprintf('ab31 = %f', ab31)
disp(' ')
fprintf('ab32 = %f', ab32)
disp(' ')
fprintf('ab33 = %f', ab33)
disp(' ')

```

Tampilan *Output* Program Operasi Pengurangan Matriks Ordo 3x3 Versi 1

Operasi Pengurangan Terhadap Matriks Ordo 3x3

Dibuat oleh: Havizul

=====

```

Masukkan nilai a11: 1
Masukkan nilai a12: 3
Masukkan nilai a13: 5
Masukkan nilai a21: 3
Masukkan nilai a22: 0
Masukkan nilai a23: 7
Masukkan nilai a31: 5
Masukkan nilai a32: 7

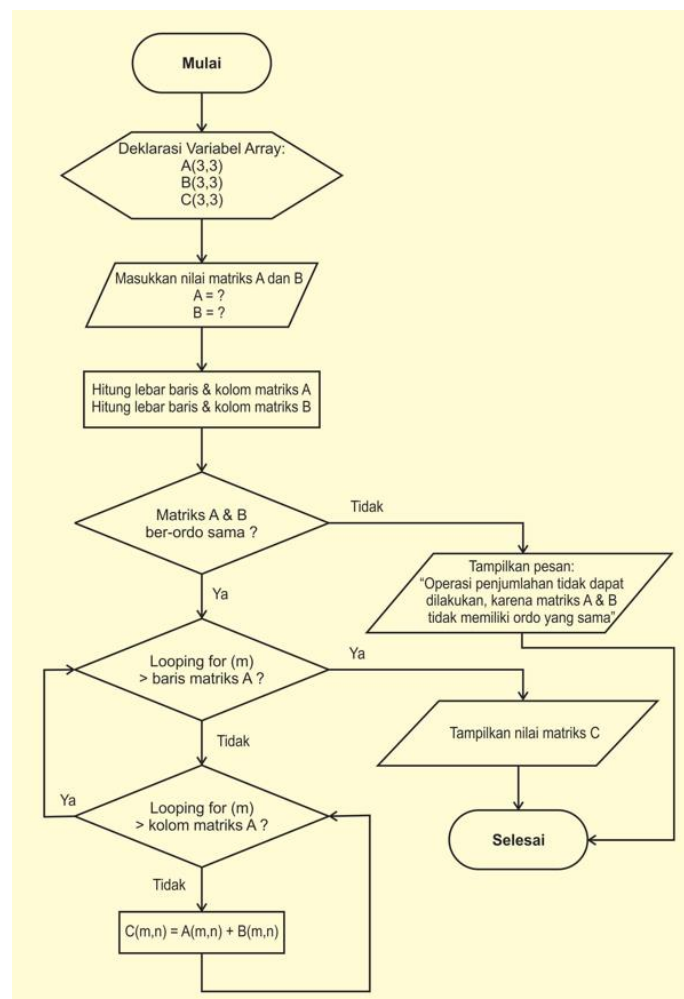
```

Masukkan nilai a33: 2
 Masukkan nilai b11: 1
 Masukkan nilai b12: 3
 Masukkan nilai b13: 4
 Masukkan nilai b21: 1
 Masukkan nilai b22: 4
 Masukkan nilai b23: 9
 Masukkan nilai b31: 0
 Masukkan nilai b32: 5
 Masukkan nilai b33: 2

Hasil dari operasi A-B:

ab11 = 0.000000
 ab12 = 0.000000
 ab13 = 1.000000
 ab21 = 2.000000
 ab22 = -4.000000
 ab23 = -2.000000
 ab31 = 5.000000
 ab32 = 2.000000
 ab33 = 0.000000

Flowchart Program Operasi Pengurangan Matriks Ordo MxN Versi 2



Gambar 7. Flowchart Program Operasi Pengurangan Matriks MxN

Kode Program Operasi Pengurangan Matriks Ordo MxN Versi 2

```

clc

disp('Operasi Pengurangan Terhadap Matriks Ordo mxn')
disp('Matriks Square -> m = n')
disp('A - B = C')
disp(' ')
disp('Notasi dalam program: ')
disp('A(1,1), A(1,2), A(1,3), ..., A(m,n)')
disp('..., B(3,1), B(3,2), ..., B(m,n)')
disp('C(1,1), C(1,2), C(1,3), ..., C(m,n)')
disp(' ')
disp('Dibuat oleh: Havizul')
disp('=====')
disp(' ')

%input nilai matriks A
A = input('Nilai matriks A = ')

%input nilai matriks B
B = input('Nilai matriks B = ')

%Ordo matriks A dan B harus sama
[barisA, kolomA] = size(A);
[barisB, kolomB] = size(B);
if (barisA ~= barisB) || (kolomA ~= kolomB) %-> cek apakah matriks memiliki
ordo yang sama
    disp(' ')
    disp('Pengurangan Matriks tidak dapat dilakukan !')
    disp('Matriks A dan B harus memiliki ordo yang sama.')
    (' ')
else
    %operasi pengurangan, A - B
    for m=1:barisA      %iterasi baris 1 hingga barisA
        for n=1:kolomA  %iterasi kolom 1 hingga kolomA
            C(m,n) = A(m,n) + (-B(m,n));
        end
    end
end

%Tampilkan nilai hasil pengurangan Matriks A dengan Matriks B
disp(' ')
disp('Hasil dari A - B = C')
disp('C =')
disp(C)

end

disp(' ')

```

Tampilan Output Program Operasi Pengurangan Matriks Ordo MxN Versi 2

```

Operasi Pengurangan Terhadap Matriks Ordo mxn
Matriks Square -> m = n
A - B = C

```

```

Notasi dalam program:
A(1,1), A(1,2), A(1,3), ..., A(m,n)
..., B(3,1), B(3,2), ..., B(m,n)
C(1,1), C(1,2), C(1,3), ..., C(m,n)

```

Dibuat oleh: Havizul

Nilai matriks A = [1 4 3 7; 9 3 2 1; 2 4 2 1]

A =

1	4	3	7
9	3	2	1
2	4	2	1

Nilai matriks B = [3 4 2 7; 1 4 2 1; 4 4 7 1]

B =

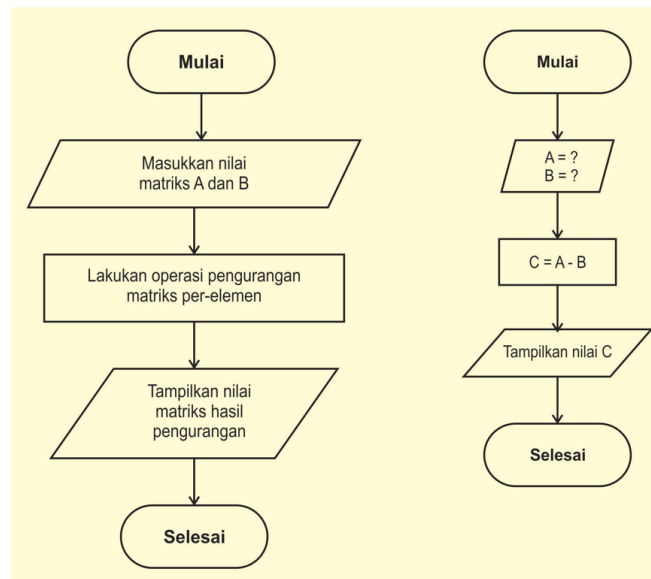
3	4	2	7
1	4	2	1
4	4	7	1

Hasil dari A - B = C

C =

-2	0	1	0
8	-1	0	0
-2	0	-5	0

Flowchart Program Operasi Pengurangan Matriks Ordo MxN Versi 3 (Khusus Matlab)



Gambar 8. Flowchart Program Operasi Pengurangan Matriks MxN

Kode Program Operasi Pengurangan Matriks Ordo MxN Versi 3 (Khusus Matlab)

```

clc
%Input nilai matriks A dan matriks B
A = [1 3 5; 3 0 7; 5 7 2]
B = [1 3 4; 1 4 9; 0 5 2]

%Operasi pengurangan matriks A dan matriks B
C = A - B
    
```

Tampilan *Output* Program Operasi Pengurangan Matriks Ordo MxN Versi 3

$$A = \begin{pmatrix} 1 & 3 & 5 \\ 3 & 0 & 7 \\ 5 & 7 & 2 \end{pmatrix}$$

$$B = \begin{pmatrix} 1 & 3 & 4 \\ 1 & 4 & 9 \\ 0 & 5 & 2 \end{pmatrix}$$

$$C = \begin{pmatrix} 0 & 0 & 1 \\ 2 & -4 & -2 \\ 5 & 2 & 0 \end{pmatrix}$$

3. Perkalian matriks

Jika A adalah matriks ordo $m \times r$ dan B adalah matriks ordo $r \times n$ maka hasil kali AB adalah matriks C ordo $m \times n$ yang elemen – elemennya ditentukan sebagai berikut. Untuk menentukan elemen C_{ij} , pilih baris ke – i matriks A dan kolom ke - j matriks B. Kalikan elemen -elemen yang bersesuaian dari baris dan kolom tersebut bersama – sama kemudian tambahkan hasilnya kalinya.

Contoh:

Matriks ordo 2 x 2

$$A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}; \quad B = \begin{pmatrix} p & q \\ r & s \end{pmatrix}$$

$$AB = \begin{pmatrix} ap + br & aq + bs \\ cp + dr & cq + ds \end{pmatrix}$$

Matriks Ordo 3 x 3

$$A = \begin{pmatrix} a & b & c \\ d & e & f \\ g & h & i \end{pmatrix}; \quad B = \begin{pmatrix} p & q & r \\ s & t & u \\ v & w & x \end{pmatrix}$$

$$AB = \begin{pmatrix} ap + bs + cv & aq + bt + cw & ar + bu + cx \\ dp + es + fv & dq + et + fw & dr + eu + fx \\ gp + hs + iv & gq + ht + iw & gr + hu + ix \end{pmatrix}$$

Contoh:

$$A = \begin{pmatrix} 0 & 2 \\ -1 & 1 \end{pmatrix} \text{ dan } B = \begin{pmatrix} 1 & -1 \\ 3 & 4 \end{pmatrix}$$

$$\begin{aligned} AB &= \begin{pmatrix} (0 \times 1) + (2 \times 3) & (0 \times (-1) + (2 \times 4)) \\ (-1 \times 1) + (1 \times 3) & (-1 \times (-1) + (1 \times 4)) \end{pmatrix} \\ &= \begin{pmatrix} 0 + 6 & 0 + 8 \\ -1 + 3 & 1 + 4 \end{pmatrix} \end{aligned}$$

$$AB = \begin{pmatrix} 6 & 8 \\ 2 & 5 \end{pmatrix}$$

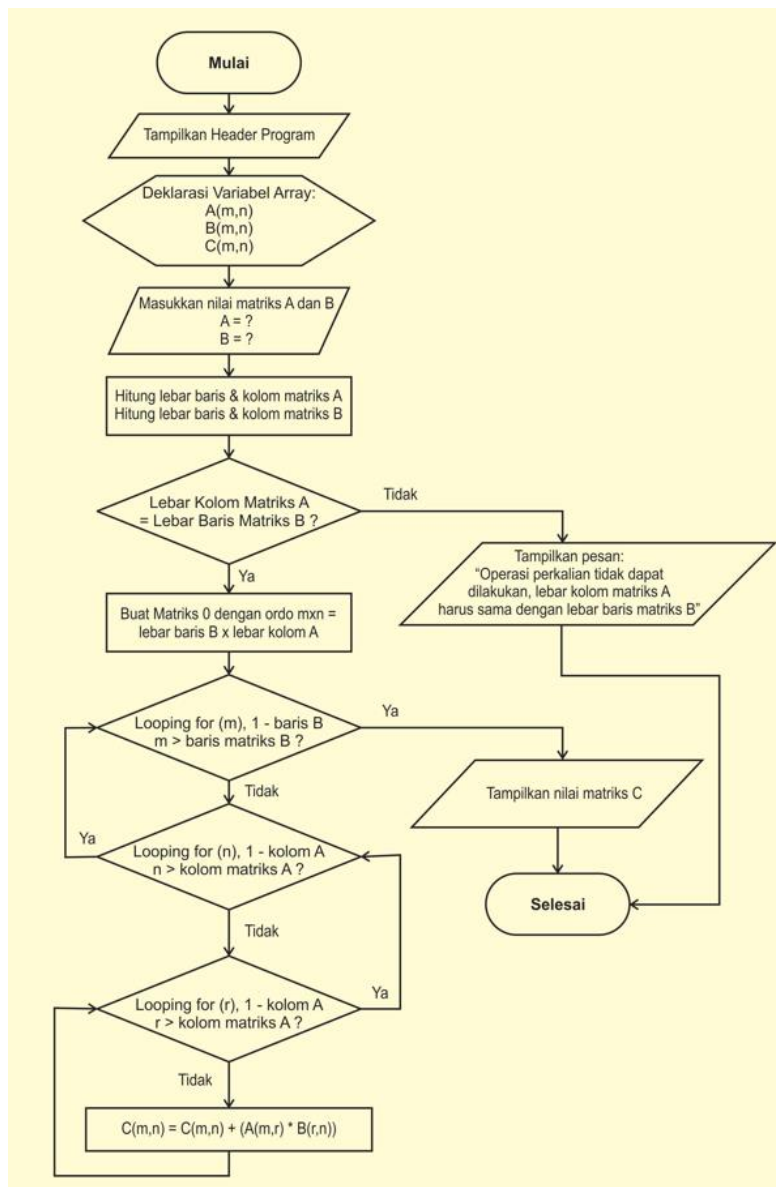
$$A = \begin{pmatrix} 1 & 3 & 5 \\ 3 & 0 & 7 \\ 5 & 7 & 2 \end{pmatrix} \cdot B = \begin{pmatrix} 1 & 3 & 4 \\ 1 & 4 & 9 \\ 0 & 5 & 2 \end{pmatrix}$$

$$AB = \begin{pmatrix} (1x1) + (3x1) + (5x0) & (1x3) + (3x4) + (5x5) & (1x4) + (3x9) + (5x2) \\ (3x1) + (0x1) + (7x0) & (3x3) + (0x4) + (7x5) & (3x4) + (0x9) + (7x2) \\ (5x1) + (7x1) + (2x0) & (5x3) + (7x4) + (2x5) & (5x4) + (7x9) + (2x2) \end{pmatrix}$$

$$AB = \begin{pmatrix} 1 + 3 + 0 & 3 + 12 + 25 & 4 + 27 + 10 \\ 3 + 0 + 0 & 9 + 0 + 35 & 12 + 0 + 14 \\ 5 + 7 + 0 & 15 + 28 + 10 & 20 + 63 + 4 \end{pmatrix}$$

$$AB = \begin{pmatrix} 4 & 40 & 41 \\ 3 & 44 & 26 \\ 12 & 43 & 87 \end{pmatrix}$$

Flowchart Program Operasi Perkalian Matriks Ordo MxN Versi 1



Gambar 9. Flowchart Program Operasi Perkalian Matriks MxN Versi 1

Kode Program Matlab Operasi Perkalian Matriks Ordo mxn Versi 1

```

clc

disp('Operasi Perkalian Matriks A(m,r) dan B(r,n) = C(m,n)')
disp('Jumlah kolom matriks A = Jumlah baris matriks B')
disp('A x B = C')
disp(' ')
disp('Dibuat oleh: Havizul')
disp('=====')
disp(' ')

%input nilai matriks A
A = input('Nilai matriks A = ')

%input nilai matriks B
B = input('Nilai matriks B = ')

%Ordo matriks A dan B harus sama
[barisA, kolomA] = size(A);
[barisB, kolomB] = size(B);
if kolomA ~= barisB
    disp(' ')
    disp('Perkalian Matriks tidak dapat dilakukan !')
    disp('Jumlah kolom matriks A harus sama dengan jumlah baris matriks B.')
    (' ')
else
    C = zeros(barisB, kolomA) %buat matriks 0 ordo mxn

    %operasi perkalian matriks, A x B
    for m=1:barisB %iterasi berdasarkan kolom matriks pertama
        for n=1:kolomA %iterasi berdasarkan baris matriks kedua
            for r=1:kolomA %iterasi pengali
                fprintf('C(m,n) = %d,%d', m,n)
                %disp(' ')
                fprintf('m,k = %d,%d', m,k)
                C(m,n) = C(m,n) + (A(m,r)*B(r,n));
            end
        end
    end

    %Tampilkan nilai hasil pengurangan Matriks A dengan Matriks B
    disp(' ')
    disp('Hasil dari A*B = C')
    disp('C =')
    disp(C)
    fprintf('Nilai Matriks C = %d', C)
end

disp(' ')

```

Tampilan Output Program Operasi Perkalian Matriks Ordo mxn

```

Operasi Perkalian Matriks A(m,r) dan B(r,n) = C(m,n)
Jumlah kolom matriks A = Jumlah baris matriks B
A x B = C

```

Dibuat oleh: Havizul

=====

Nilai matriks A = [0 2; -1 1]

$$A = \begin{pmatrix} 0 & 2 \\ -1 & 1 \end{pmatrix}$$

Nilai matriks B = [1 -1; 3 4]

$$B = \begin{pmatrix} 1 & -1 \\ 3 & 4 \end{pmatrix}$$

$$C = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$$

Hasil dari $A \cdot B = C$

$$C = \begin{pmatrix} 6 & 8 \\ 2 & 5 \end{pmatrix}$$

Operasi Perkalian Matriks $A(m,r)$ dan $B(r,n) = C(m,n)$

Jumlah kolom matriks A = Jumlah baris matriks B

$$A \times B = C$$

Dibuat oleh: Havizul

=====

Nilai matriks A = [1 3 5; 3 0 7; 5 7 2]

$$A = \begin{pmatrix} 1 & 3 & 5 \\ 3 & 0 & 7 \\ 5 & 7 & 2 \end{pmatrix}$$

Nilai matriks B = [1 3 4; 1 4 9; 0 5 2]

$$B = \begin{pmatrix} 1 & 3 & 4 \\ 1 & 4 & 9 \\ 0 & 5 & 2 \end{pmatrix}$$

$$C = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

Hasil dari $A \cdot B = C$

$$C = \begin{pmatrix} 4 & 40 & 41 \\ 3 & 44 & 26 \\ 12 & 53 & 87 \end{pmatrix}$$

Operasi Perkalian Matriks $A(m,r)$ dan $B(r,n) = C(m,n)$

Jumlah kolom matriks A = Jumlah baris matriks B

$$A \times B = C$$

Dibuat oleh: Havizul

Nilai matriks A = [1 3 5; 3 0 7; 5 7 2]

A =

1	3	5
3	0	7
5	7	2

Nilai matriks B = [1 3 4; 2 1 3]

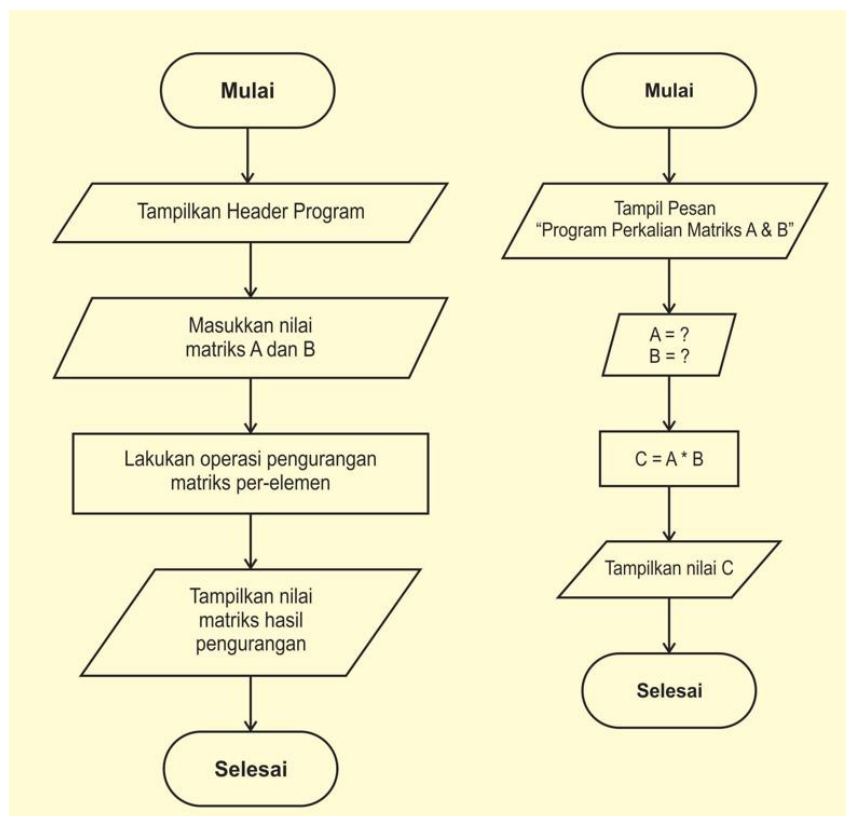
B =

1	3	4
2	1	3

Perkalian Matriks tidak dapat dilakukan !
Jumlah kolom matriks A harus sama dengan jumlah baris matriks B.

ans =

Flowchart Program Operasi Perkalian Matriks Ordo MxN Versi 2



Gambar 10. Flowchart Program Operasi Perkalian Matriks MxN Versi 2

Kode Program Matlab Operasi Perkalian Matriks Ordo mxn Versi 2 (Fungsi Perkalian Matlab)

```
clc
```

```
disp('Operasi Perkalian Terhadap Matriks Ordo mxn')
disp('Matriks A(m,n) dan matriks B(m,n)')
disp('A(m,n) * B(m,n) = C(m,n)')
```

```

disp(' ')
disp('Dibuat oleh: Havizul')
disp('=====')
disp(' ')

%input nilai matriks A
A = input('Nilai matriks A = ')

%input nilai matriks B
B = input('Nilai matriks B = ')

%Kalikan matriks A dan B
C = A*B;

%Tampilkan nilai hasil perkalian Matriks A dengan Matriks B
disp(' ')
disp('Hasil dari A * B = C')
disp('C =')
disp(C)

disp(' ')

```

Tampilan Output Program Operasi Perkalian Matriks Ordo mxn

Operasi Perkalian Terhadap Matriks Ordo mxn
Matriks A(m,n) dan matriks B(m,n)
 $A(m,n) * B(m,n) = C(m,n)$

Dibuat oleh: Havizul
=====

Nilai matriks A = [1 2 4; 3 1 7]

A =

1	2	4
3	1	7

Nilai matriks B = [2 4; 1 3; 7 1]

B =

2	4
1	3
7	1

Hasil dari A * B = C
C =

32	14
56	22

Operasi Perkalian Terhadap Matriks Ordo mxn
Matriks A(m,n) dan matriks B(m,n)
 $A(m,n) * B(m,n) = C(m,n)$

Dibuat oleh: Havizul
=====

Nilai matriks A = [1 2 4; 3 1 7; 4 8 2]

$$A = \begin{pmatrix} 1 & 2 & 4 \\ 3 & 1 & 7 \\ 4 & 8 & 2 \end{pmatrix}$$

Nilai matriks B = [1 3 2; 9 2 7; 3 9 2]

$$B = \begin{pmatrix} 1 & 3 & 2 \\ 9 & 2 & 7 \\ 3 & 9 & 2 \end{pmatrix}$$

Hasil dari A * B = C

$$C = \begin{pmatrix} 31 & 43 & 24 \\ 33 & 74 & 27 \\ 82 & 46 & 68 \end{pmatrix}$$

4. Transpose Matriks

Transpose Matriks adalah matriks baru yang diperoleh dengan cara menukar elemen-elemen baris menjadi elemen kolom atau sebaliknya.

$$(A + B)^T = A^T + B^T$$

Contoh:

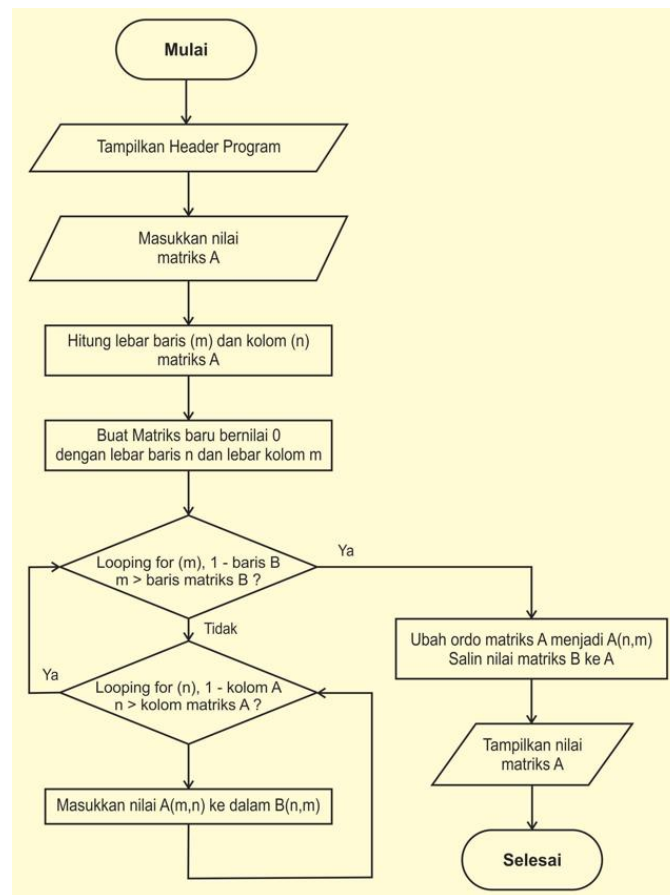
$$A = \begin{pmatrix} 1 & 3 & 5 \\ 3 & 0 & 7 \\ 5 & 7 & 2 \end{pmatrix} \quad B = \begin{pmatrix} 1 & 3 & 4 \\ 1 & 4 & 9 \\ 0 & 5 & 2 \end{pmatrix}$$

$$\begin{pmatrix} 1 & 3 & 5 \\ 3 & 0 & 7 \\ 5 & 7 & 2 \end{pmatrix} + \begin{pmatrix} 1 & 3 & 4 \\ 1 & 4 & 9 \\ 0 & 5 & 2 \end{pmatrix}^T = \begin{pmatrix} 1 & 3 & 5 \\ 3 & 0 & 7 \\ 5 & 7 & 2 \end{pmatrix}^T + \begin{pmatrix} 1 & 3 & 4 \\ 1 & 4 & 9 \\ 0 & 5 & 2 \end{pmatrix}^T$$

$$\begin{pmatrix} 1+1 & 3+3 & 5+4 \\ 3+1 & 0+4 & 7+9 \\ 5+0 & 7+5 & 2+2 \end{pmatrix}^T = \begin{pmatrix} 1 & 3 & 5 \\ 3 & 0 & 7 \\ 5 & 7 & 2 \end{pmatrix} + \begin{pmatrix} 1 & 1 & 0 \\ 3 & 4 & 5 \\ 4 & 9 & 2 \end{pmatrix}$$

$$\begin{pmatrix} 2 & 6 & 9 \\ 4 & 4 & 16 \\ 5 & 12 & 4 \end{pmatrix}^T = \begin{pmatrix} 2 & 4 & 5 \\ 6 & 4 & 12 \\ 9 & 16 & 4 \end{pmatrix}$$

$$\begin{pmatrix} 2 & 4 & 5 \\ 6 & 4 & 12 \\ 9 & 16 & 4 \end{pmatrix} = \begin{pmatrix} 2 & 4 & 5 \\ 6 & 4 & 12 \\ 9 & 16 & 4 \end{pmatrix}$$

Flowchart Program Operasi Transpose Matriks Ordo MxN Versi 1

Gambar 11. Flowchart Program Operasi Transpose Matriks MxN Versi 1

Kode Program Matlab Operasi Transpose Matriks Ordo mxn Versi 1

```

clc

disp('Operasi Transpose Matriks Ordo mxn')
disp('Matriks A(m,n)')
disp('Transpose A(m,n) = B(n,m), A(n,m) = B(m,n)')
disp(' ')
disp('Dibuat oleh: Havizul')
disp('=====')
disp(' ')

%input nilai matriks A
A = input('Nilai matriks A = ')

%Hitung lebar matriks
[barisA, kolomA] = size(A);

%lakukan pertukaran elemen-elemen matriks
B = zeros(kolomA, barisA);

for m = 1:1:barisA
    for n = 1:1:kolomA
        B(n,m) = A(m,n);
    end
end
A = B;
  
```

```

%Tampilkan nilai hasil transpose matriks A
disp(' ')
disp('Hasil dari Transpose A :')
disp('A =')
disp(A)

disp(' ')

```

Tampilan *Output* Program Matlab Operasi Transpose Matriks Ordo mxn Versi 1

Operasi Transpose Matriks Ordo mxn
 Matriks A(m,n)
 Transpose A(m,n) = A(n,m)

Dibuat oleh: Havizul

=====

Nilai matriks A = [1 3 4 2; 2 3 1 7; 9 7 8 1; 7 9 7 4]

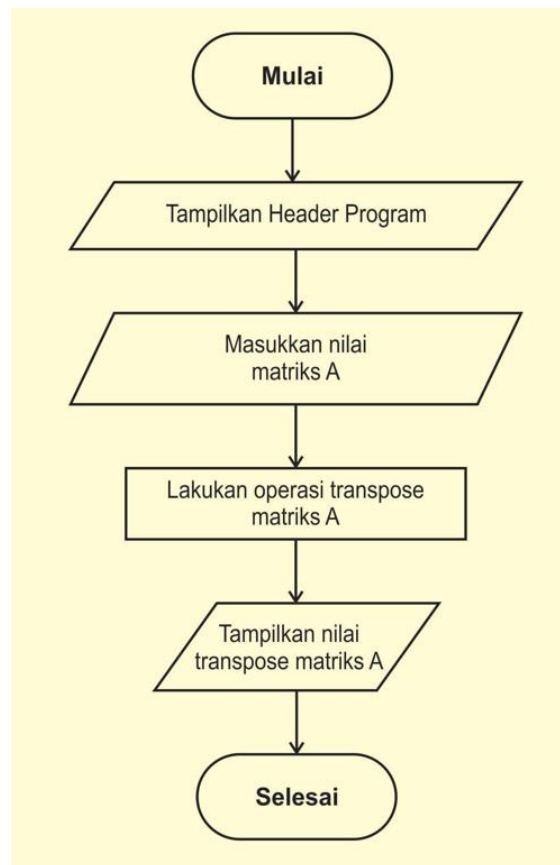
A =

1	3	4	2
2	3	1	7
9	7	8	1
7	9	7	4

Hasil dari Transpose A :

A =

1	2	9	7
3	3	7	9
4	1	8	7
2	7	1	4

Flowchart Program Operasi Transpose Matriks Ordo MxN Versi 2

Gambar 12. Flowchart Program Operasi Transpose Matriks Ordo MxN Versi 2

Kode Program Matlab Operasi Transpose Matriks Ordo mxn Versi 2

```

clc

disp('Operasi Transpose Matriks Ordo mxn')
disp('Matriks A(m,n)')
disp('Transpose A(m,n) = A(n,m)')
disp(' ')
disp('Dibuat oleh: Havizul')
disp('=====')
disp(' ')

%input nilai matriks A
A = input('Nilai matriks A = ')

%Operasi Transpose Matriks
A = A';

%Tampilkan nilai hasil transpose matriks A
disp(' ')
disp('Hasil dari Transpose A :')
disp('A =')
disp(A)
  
```

```
disp(' ');
```

Tampilan *Output Program Matlab Operasi Transpose Matriks Ordo mxn Versi 2*

```
Operasi Transpose Matriks Ordo mxn
```

```
Matriks A(m,n)
```

```
Transpose A(m,n) = A(n,m)
```

```
Dibuat oleh: Havizul
```

```
=====
```

```
Nilai matriks A = [1 2 3; 4 3 7]
```

```
A =
```

```
    1    2    3
    4    3    7
```

```
Hasil dari Transpose A :
```

```
A =
```

```
    1    4
    2    3
    3    7
```

5. Determinan Matriks

a. Metode Biasa

Misalkan A adalah suatu matrik bujursangkar. Fungsi determinan dinotasikan dengan \det dan kita mendefenisikan $\det(A)$ sebagai jumlah dari semua hasil kali elementer bertanda dari A. Angka $\det(A)$ disebut determinan dari A.

Tentukan:

$$1) \det \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$$

$$2) \det \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

Penyelesaian:

1) Hasil kali elementer bertanda dari matrik diatas adalah $a_{11}a_{22}$ dan $-a_{12}a_{21}$. Jadi determinannya $a_{11}a_{22} - a_{12}a_{21}$.

2) Hasil kali elementer bertanda dari matrik diatas adalah $a_{11}a_{22}a_{33}$, $-a_{11}a_{23}a_{32}$, $-a_{12}a_{21}a_{33}$, $a_{12}a_{23}a_{31}$, $a_{13}a_{21}a_{32}$ dan $-a_{13}a_{22}a_{31}$. Jadi determinannya adalah $a_{11}a_{22}a_{33} + a_{12}a_{23}a_{31} + a_{13}a_{21}a_{32} - a_{13}a_{22}a_{31} - a_{11}a_{23}a_{32} - a_{12}a_{21}a_{33}$.

$$(a_{11}a_{22}a_{33} + a_{12}a_{23}a_{31} + a_{13}a_{21}a_{32}) - (a_{13}a_{22}a_{31} + a_{11}a_{23}a_{32} + a_{12}a_{21}a_{33})$$

Hitunglah determinan dari:

$$\text{a) } A = \begin{bmatrix} 2 & -1 \\ 3 & 5 \end{bmatrix}$$

$$\text{b) } E = \begin{bmatrix} 2 & 3 & 4 \\ 5 & 6 & 2 \\ 1 & 0 & 0 \end{bmatrix}$$

Penyelesaian:

$$\text{a) } A = \begin{bmatrix} 2 & -1 \\ 3 & 5 \end{bmatrix}$$

$$\det A = 2.5 - (-1.3)$$

$$\det A = 10 + 3 = 13$$

$$\text{b) } E = \left(\begin{array}{ccc|cc} 2 & 3 & 4 & 2 & 3 \\ 5 & 6 & 2 & 5 & 6 \\ 1 & 0 & 0 & 1 & 0 \end{array} \right)$$

$$\text{Det } E = [(2.6.0) + (3.2.1) + 4.5.0] - [(1.6.4) + (0.2.2) + (0.5.3)]$$

$$\text{Det } E = (0 + 6 + 0) - (24 + 0 + 0)$$

$$\text{Det } E = 6 - 24 = -18$$

Kode Program Matlab Operasi Perhitungan Determinan Matriks

```
clc

A = [2 -1; 3 5]
E = [2 3 4; 5 6 2; 1 0 0]

determinanA = det(A)
determinanE = det(E)
```

Tampilan *output* Program Matlab Operasi Perhitungan Determinan Matriks

```
A =

     2     -1
     3      5

E =

     2      3      4
     5      6      2
     1      0      0

determinanA =
    13
```

determinanE =
-18.0000000000000007

b. Determinan Metode Chio

Matriks Ordo 3 x 3

$$A = \begin{bmatrix} 3 & 1 & 2 \\ 2 & 1 & 3 \\ 1 & 3 & 1 \end{bmatrix}$$

$$\det A = \frac{1}{a_{11} \text{ordo}^{-2}} \left[\begin{array}{c|c|c} \begin{bmatrix} 3 & 1 \\ 2 & 1 \\ 3 & 1 \\ 1 & 3 \end{bmatrix} & \begin{bmatrix} 3 \\ 2 \\ 3 \\ 1 \end{bmatrix} & \begin{bmatrix} 2 \\ 3 \\ 2 \\ 1 \end{bmatrix} \end{array} \right]$$

$$\det A = \frac{1}{(3)^{3-2}} \left[\begin{array}{c|c|c} \begin{bmatrix} 3 & 1 \\ 2 & 1 \\ 3 & 1 \\ 1 & 3 \end{bmatrix} & \begin{bmatrix} 3 \\ 2 \\ 3 \\ 1 \end{bmatrix} & \begin{bmatrix} 2 \\ 3 \\ 2 \\ 1 \end{bmatrix} \end{array} \right]$$

$$\det A = \frac{1}{3} \begin{bmatrix} (3-2) & (9-4) \\ (9-1) & (3-2) \end{bmatrix}$$

$$\det A = \frac{1}{3} \begin{bmatrix} 1 & 5 \\ 8 & 1 \end{bmatrix}$$

$$\det A = \frac{1}{3} (1 - 40)$$

$$\det A = \frac{1}{3} (-39)$$

$$\det A = -13$$

Kode Program Matlab Operasi Perhitungan Determinan Matriks Ordo 3x3 Metode Chio

```
clc

a11 = input('Masukkan nilai a11 = ');
a12 = input('Masukkan nilai a12 = ');
a13 = input('Masukkan nilai a13 = ');
a21 = input('Masukkan nilai a21 = ');
a22 = input('Masukkan nilai a22 = ');
a23 = input('Masukkan nilai a23 = ');
a31 = input('Masukkan nilai a31 = ');
a32 = input('Masukkan nilai a32 = ');
a33 = input('Masukkan nilai a33 = ');
ord3 = 3;

varCh1 = 1 / (a11^(ord3-2));
c11 = a11*a22 - a12*a21;
c12 = a11*a23 - a13*a21;
c21 = a11*a32 - a12*a31;
c22 = a11*a33 - a13*a31;
```

```

d11 = c11 * c22;
d12 = c12 * c21;

detAChio = varCh1 * (d11 - d12)

```

Tampilan *Output* Program Matlab Operasi Perhitungan Determinan Matriks Ordo 3x3 Metode Chio

```

clc

A = [2 -1; 3 5]
E = [2 3 4; 5 6 2; 1 0 0]

determinanA = det(A)
determinanE = det(E)

```

Matriks Ordo 4 x 4

$$A = \begin{bmatrix} 2 & 1 & 6 & 7 \\ 3 & 2 & 4 & 5 \\ 4 & 4 & 2 & 3 \\ 5 & 6 & 1 & 4 \end{bmatrix}$$

$$\det A = \frac{1}{2^{4-2}} \begin{vmatrix} \begin{bmatrix} 2 & 1 \\ 3 & 2 \end{bmatrix} & \begin{bmatrix} 2 & 6 \\ 3 & 4 \end{bmatrix} & \begin{bmatrix} 2 & 7 \\ 3 & 5 \end{bmatrix} \\ \begin{bmatrix} 2 & 1 \\ 4 & 4 \end{bmatrix} & \begin{bmatrix} 2 & 6 \\ 4 & 2 \end{bmatrix} & \begin{bmatrix} 2 & 7 \\ 4 & 3 \end{bmatrix} \\ \begin{bmatrix} 2 & 1 \\ 5 & 6 \end{bmatrix} & \begin{bmatrix} 2 & 6 \\ 5 & 1 \end{bmatrix} & \begin{bmatrix} 2 & 7 \\ 5 & 4 \end{bmatrix} \end{vmatrix}$$

$$\det A = \frac{1}{2^2} \begin{bmatrix} (4-3) & (8-18) & (10-21) \\ (8-4) & (4-24) & (6-28) \\ (12-5) & (2-30) & (8-35) \end{bmatrix}$$

$$\det A = \frac{1}{4} \begin{bmatrix} 1 & -10 & -11 \\ 4 & -20 & -22 \\ 7 & -28 & -27 \end{bmatrix}$$

$$\det A = \frac{1}{4} \frac{1}{1^{3-2}} \begin{vmatrix} \begin{bmatrix} 1 & -10 \\ 4 & -20 \end{bmatrix} & \begin{bmatrix} 1 & -11 \\ 4 & -22 \end{bmatrix} \\ \begin{bmatrix} 1 & -10 \\ 7 & -28 \end{bmatrix} & \begin{bmatrix} 1 & -11 \\ 7 & -27 \end{bmatrix} \end{vmatrix}$$

$$\det A = \frac{1}{4} \frac{1}{1^1} \begin{bmatrix} (-20 - (-40)) & (-22 - (-44)) \\ (-28 - (-70)) & (-27 - (-77)) \end{bmatrix}$$

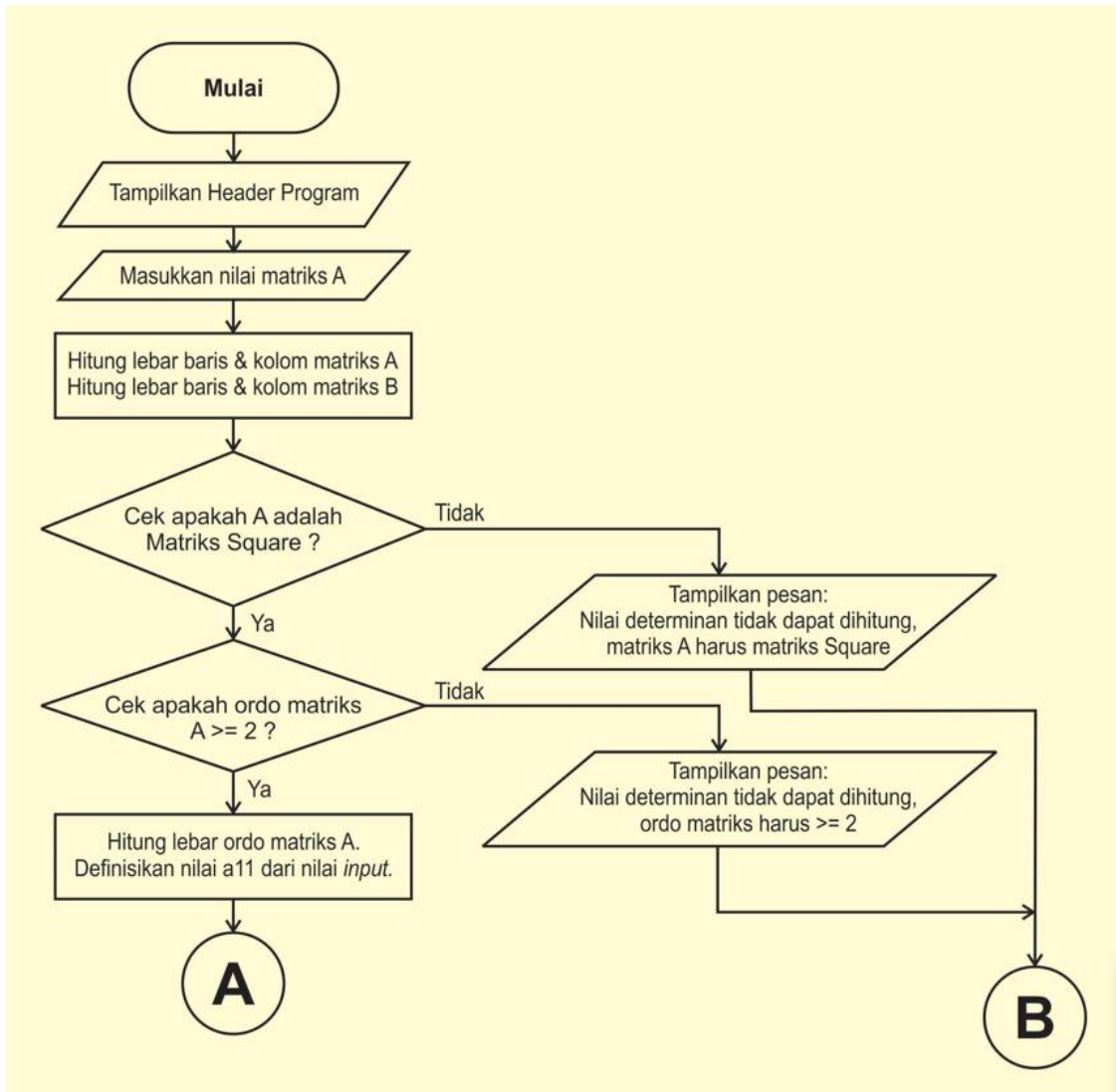
$$\det A = \frac{1}{4} \begin{bmatrix} 20 & 22 \\ 42 & 50 \end{bmatrix}$$

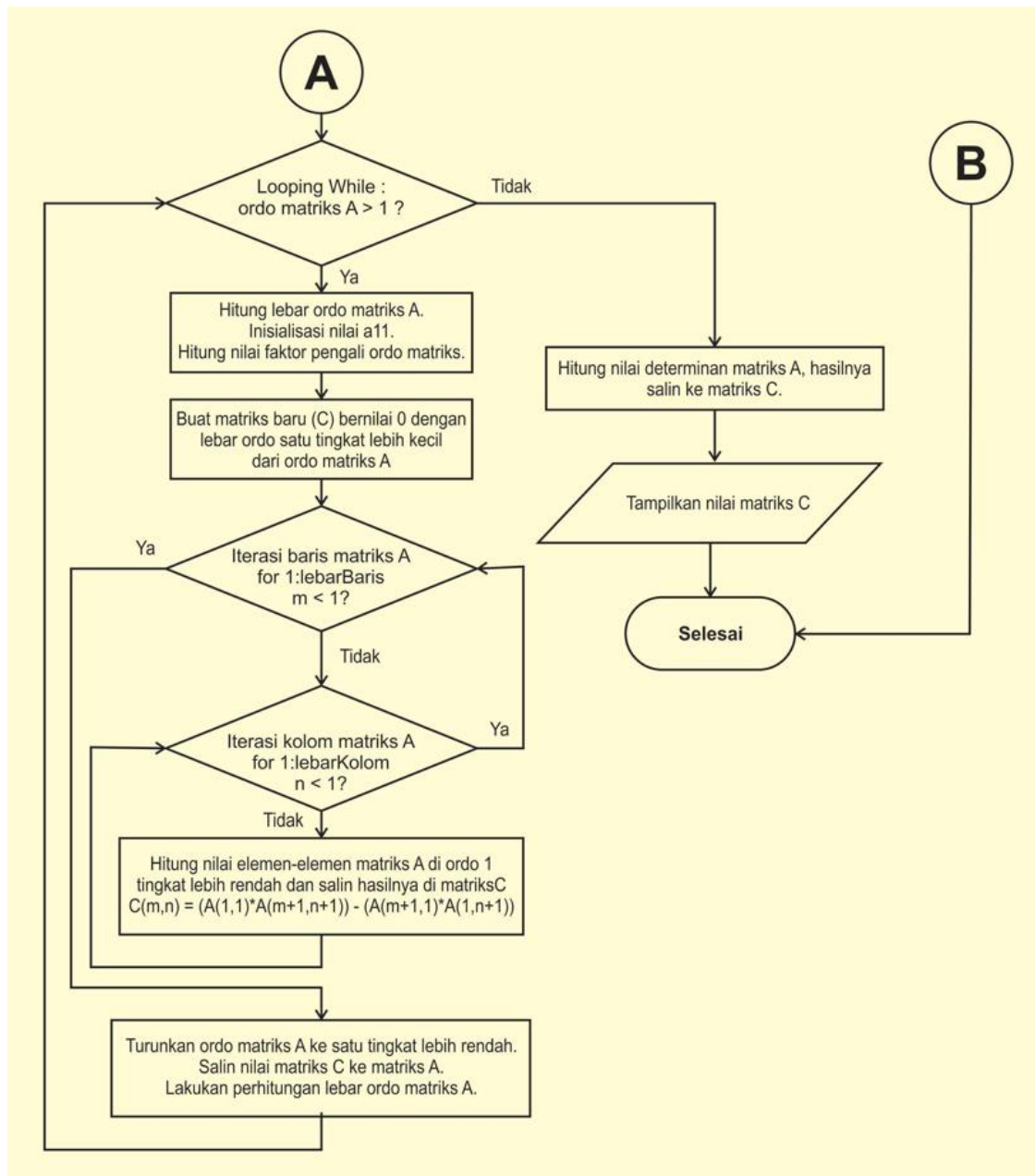
$$\det A = \frac{1}{4} (1000 - 924)$$

$$\det A = \frac{1}{4} 76$$

$$\det A = 19$$

Flowchart Program Matlab Operasi Perhitungan Determinan Matriks Ordo MxN Metode Chio





Gambar 13. Flowchart Program Matlab Operasi Perhitungan Determinan Matriks Ordo MxN Metode Chio

Kode Program Matlab Operasi Perhitungan Determinan Matriks Ordo MxN Metode Chio

```

clc

disp('Perhitungan Determinan Matriks Ordo mxn')
disp('A = Matriks Square')
disp(' ')
disp('Dibuat oleh: Havizul')
disp('=====')
disp(' ')

%input nilai matriks A
A = input('Nilai matriks A = ')
detA = 0;
  
```

```

%Matriks A harus Matriks Square
[barisA, kolomA] = size(A);
if barisA ~= kolomA
    disp(' ')
    disp('Determinan matriks A tidak dapat dihitung !')
    disp('Matriks A harus matriks square.')
    (' ')
else
    if barisA < 2 || kolomA < 2
        disp(' ')
        disp('Matriks A harus memiliki ordo 2 atau lebih besar.')
        (' ')
    else
        %perhitungan determinan matriks
        f = 1;
        while length(A) > 1
            ordo = length(A);
            all = A(1,1);
            f = f * (1/all^(ordo-2));
            C = zeros(barisA-1, kolomA-1);

            for m=1:barisA-1      %iterasi berdasarkan baris matriks
                for n=1:kolomA-1 %iterasi berdasarkan kolom matriks
                    C(m,n) = (A(1,1)*A(m+1,n+1)) - (A(m+1,1)*A(1,n+1));
                end
            end

            [barisC, kolomC] = size(C);
            A = zeros(barisC, kolomC);
            A = C;
            [barisA, kolomA] = size(A);
        end

        detA = f*A;
    end

    %Tampilkan nilai hasil pengurangan Matriks A dengan Matriks B
    disp(' ')
    fprintf('Nilai Determinan A = %d', detA)
    disp(' ')
    disp('_____')

end

disp(' ')

```

Tampilan *output* program perhitungan determinan matriks ordo mxn metode chio:

Perhitungan Determinan Matriks Ordo mxn
A = Matriks Square

Dibuat oleh: Havizul

=====

Nilai matriks A = [9 8 4 3; 7 8 3 2; 4 2 3 8; 3 2 8 9]

A =

9	8	4	3
7	8	3	2
4	2	3	8
3	2	8	9

Nilai Determinan A = -534

c. Determinan Metode Crout

$$L_{ik} = a_{ij} - \sum_{k=1}^{j-1} l_{ik} u_{ik}, j \leq i, i = 2, 3, \dots, n$$

$$U_{ij} = \frac{a_{ij} - \sum_{k=1}^{i-1} l_{ik} u_{ik}}{l_{ii}}, i \leq j, j = 2, 3, \dots, n$$

Matriks ordo 3 x 3

$$A = \begin{bmatrix} 3 & 1 & 2 \\ 2 & 1 & 3 \\ 1 & 3 & 1 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

$$\begin{bmatrix} l_{11} & 0 & 0 \\ l_{21} & l_{22} & 0 \\ l_{31} & l_{32} & l_{33} \end{bmatrix} \begin{bmatrix} 1 & u_{12} & u_{13} \\ 0 & 1 & u_{23} \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 3 & 1 & 2 \\ 2 & 1 & 3 \\ 1 & 3 & 1 \end{bmatrix}$$

Rumus Perhitungan:

Iterasi 1

$$l_{11} = a_{11} \quad l_{11} = 3;$$

$$l_{21} = a_{21} \quad l_{21} = 2;$$

$$l_{31} = a_{31} \quad l_{31} = 1$$

Iterasi 2

$$u_{12} = \frac{a_{12}}{a_{11}} \quad u_{12} = \frac{1}{3}$$

$$u_{13} = \frac{a_{13}}{a_{11}} \quad u_{13} = \frac{2}{3}$$

Iterasi 3

$$l_{22} = a_{22} - l_{21}u_{12} \quad l_{22} = 1 - 2 \cdot \frac{1}{3} = 1 - \frac{2}{3} = \frac{1}{3}$$

$$l_{32} = a_{32} - l_{31}u_{12} \quad l_{32} = 3 - 1 \cdot \frac{1}{3} = \frac{8}{3}$$

Iterasi 4

$$u_{23} = \frac{a_{23} - l_{21}u_{13}}{l_{22}} = \frac{3 - 2 \cdot \frac{2}{3}}{\frac{1}{3}} = \frac{\frac{9}{3} - \frac{4}{3}}{\frac{1}{3}} = \frac{5}{3} \times \frac{3}{1} = 5$$

Iterasi 5

$$l_{33} = a_{33} - l_{31}u_{13} - l_{32}u_{23}$$

$$l_{33} = 1 - 1 \cdot \frac{2}{3} - \frac{8}{3} \cdot 5 = 1 - \frac{2}{3} - \frac{40}{3} = \frac{3 - 2 - 40}{3} = \frac{-39}{3} = -13$$

$$\begin{bmatrix} 3 & 0 & 0 \\ 2 & \frac{1}{3} & 0 \\ 1 & \frac{8}{3} & -13 \end{bmatrix} \begin{bmatrix} 1 & \frac{1}{3} & \frac{2}{3} \\ 0 & 1 & 5 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 3 & 1 & 2 \\ 2 & 1 & 3 \\ 1 & 3 & 1 \end{bmatrix}$$

$$\text{Det L} = 3 \times \frac{1}{3} \times (-13) = -13$$

$$\text{Det U} = 1 \times 1 \times 1 = 1$$

$$\text{Jadi, Det A} = \text{Det L} \times \text{Det U} = -13 \times 1 = -13$$

Kode program Matlab perhitungan determinan matriks ordo 3x3 metode crout:

```
clc

disp('Perhitungan Determinan Matriks Ordo 3x3 Metode Crout.')
disp('=====')

a11 = input('Masukkan nilai a11: ');
a12 = input('Masukkan nilai a12: ');
a13 = input('Masukkan nilai a13: ');
a21 = input('Masukkan nilai a21: ');
a22 = input('Masukkan nilai a22: ');
a23 = input('Masukkan nilai a23: ');
a31 = input('Masukkan nilai a31: ');
a32 = input('Masukkan nilai a32: ');
a33 = input('Masukkan nilai a33: ');

if a11 == 0
    disp(' ')
    disp('Nilai determinan = 0.')
    disp(' ')

    disp('Dalam metode Crout, nilai a11 tidak boleh bernilai 0')
    disp(' ');
else
    l11 = a11;
    l21 = a21;
    l31 = a31;

    u12 = a12/a11;
    u13 = a13/a11;

    l22 = a22 - (l21*u12);
    l32 = a32 - (l31*u12);

    u23 = (a23 - (l21*u13)) / l22;
    l33 = a33 - (l31*u13) - (l32*u23);

    detL = l11*l22*l33;
    detU = 1*1*1;
```



```

detA = detL * detU;

disp(' ')
disp('Determinan Matriks A = ');
disp(detA);
disp(' ')
fprintf('o11 = %f', o11)
end

```

Tampilan *output* program perhitungan determinan matriks ordo 3x3 metode crout:

Perhitungan Determinan Matriks Ordo 3x3 Metode Crout.

```

=====
Masukkan nilai a11: 3
Masukkan nilai a12: 1
Masukkan nilai a13: 2
Masukkan nilai a21: 2
Masukkan nilai a22: 1
Masukkan nilai a23: 3
Masukkan nilai a31: 1
Masukkan nilai a32: 3
Masukkan nilai a33: 1

```

```

Determinan Matriks A =
-13.0000

```

Matriks Ordo 4 x 4

$$A = \begin{bmatrix} 2 & 1 & 6 & 7 \\ 3 & 2 & 4 & 5 \\ 4 & 4 & 2 & 3 \\ 5 & 6 & 1 & 4 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix},$$

Kasus n = 4

$$\begin{bmatrix} l_{11} & 0 & 0 & 0 \\ l_{21} & l_{22} & 0 & 0 \\ l_{31} & l_{32} & l_{33} & 0 \\ l_{41} & l_{42} & l_{43} & l_{44} \end{bmatrix} \begin{bmatrix} 1 & u_{12} & u_{13} & u_{14} \\ 0 & 1 & u_{23} & u_{24} \\ 0 & 0 & 1 & u_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix},$$

Rumus perhitungan:

Iterasi 1

$$l_{11} = a_{11} = 2; l_{21} = a_{21} = 3; l_{31} = a_{31} = 4; l_{41} = a_{41} = 5$$

Iterasi 2

$$u_{12} = \frac{a_{12}}{a_{11}} = \frac{1}{2}$$

$$u_{13} = \frac{a_{13}}{a_{11}} = \frac{6}{2} = 3$$

$$u_{14} = \frac{a_{14}}{a_{11}} = \frac{7}{2}$$

Iterasi 3

$$l_{22} = a_{22} - l_{21}u_{12} = 2 - 3 \cdot \frac{1}{2} = \frac{1}{2}$$

$$l_{32} = a_{32} - l_{31}u_{12} = 4 - 4 \cdot \frac{1}{2} = 2$$

$$l_{42} = a_{42} - l_{41}u_{12} = 6 - 5 \cdot \frac{1}{2} = \frac{7}{2}$$

Iterasi 4

$$u_{23} = \frac{a_{23} - l_{21}u_{13}}{l_{22}} = \frac{4 - 3 \cdot 3}{\frac{1}{2}} = \frac{4 - 9}{\frac{1}{2}} = -\frac{5}{1} \times \frac{2}{1} = -10$$

$$u_{24} = \frac{a_{24} - l_{21}u_{14}}{l_{22}} = \frac{5 - 3 \cdot \frac{7}{2}}{\frac{1}{2}} = \frac{5 - \frac{21}{2}}{\frac{1}{2}} = \frac{\frac{10}{2} - \frac{21}{2}}{\frac{1}{2}} = \frac{\frac{-11}{2}}{\frac{1}{2}} = -\frac{11}{2} \times \frac{2}{1} = -11$$

Iterasi 5

$$l_{33} = a_{33} - l_{31}u_{13} - l_{32}u_{23}$$

$$l_{33} = 2 - 4 \cdot 3 - 2(-10)$$

$$l_{33} = 2 - 12 + 20 = 10$$

$$l_{43} = a_{43} - l_{41}u_{13} - l_{42}u_{23}$$

$$l_{43} = 1 - 5 \cdot 3 - \frac{7}{2}(-10)$$

$$l_{43} = 1 - 15 + 35 = 21$$

Iterasi 6

$$u_{34} = \frac{a_{34} - l_{31}u_{14} - l_{32}u_{24}}{l_{33}}$$

$$u_{34} = \frac{3 - 4 \cdot \frac{7}{2} - 2 - 2(-11)}{10}$$

$$u_{34} = \frac{3 - 14 + 22}{10}$$

$$u_{34} = \frac{11}{10}$$

Iterasi 7

$$l_{44} = a_{44} - l_{41}u_{14} - l_{42}u_{24} - l_{43}u_{34}$$

$$l_{44} = 4 - 5 \cdot \frac{7}{2} - \frac{7}{2}(-11) - 21 \cdot \frac{11}{10}$$

$$l_{44} = 4 - \frac{35}{2} + \frac{77}{2} - \frac{231}{10}$$

$$l_{44} = 4 + 21 - \frac{231}{10}$$

$$l_{44} = 25 - \frac{231}{10}$$

$$l_{44} = \frac{250}{10} - \frac{231}{10}$$

$$l_{44} = \frac{19}{10}$$

$$\begin{bmatrix} 2 & 0 & 0 & 0 \\ 3 & \frac{1}{2} & 0 & 0 \\ 4 & 2 & 10 & 0 \\ 5 & \frac{7}{2} & 21 & \frac{19}{10} \end{bmatrix} \begin{bmatrix} 1 & \frac{1}{2} & 3 & \frac{7}{2} \\ 0 & 1 & -10 & -11 \\ 0 & 0 & 1 & \frac{11}{10} \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 2 & 1 & 6 & 7 \\ 3 & 2 & 4 & 5 \\ 4 & 4 & 2 & 3 \\ 5 & 6 & 1 & 4 \end{bmatrix}$$

$$\text{Det } l = 2 \times \frac{1}{2} \times 10 \times \frac{19}{10} = 19$$

$$\text{Det } U = 1 \times 1 \times 1 = 1$$

Jadi,

$$\text{Det } A = \text{Det } L \times \text{Det } U = 19 \times 1 = 19$$

Kode program Matlab perhitungan determinan matriks ordo 4x4 metode crout:

```

clc

disp('Perhitungan Determinan Matriks Ordo 4x4 Metode Crout.')
disp('=====')
disp(' ')
a11 = input('Masukkan nilai a11: ');
a12 = input('Masukkan nilai a12: ');
a13 = input('Masukkan nilai a13: ');
a14 = input('Masukkan nilai a14: ');

a21 = input('Masukkan nilai a21: ');
a22 = input('Masukkan nilai a22: ');
a23 = input('Masukkan nilai a23: ');
a24 = input('Masukkan nilai a24: ');

a31 = input('Masukkan nilai a31: ');
a32 = input('Masukkan nilai a32: ');
a33 = input('Masukkan nilai a33: ');
a34 = input('Masukkan nilai a34: ');

a41 = input('Masukkan nilai a41: ');
a42 = input('Masukkan nilai a42: ');
a43 = input('Masukkan nilai a43: ');
    
```

```

a44 = input('Masukkan nilai a44: ');

if a11 == 0
    disp(' ')
    disp('Nilai determinan = 0.')
    disp(' ')

    disp('Dalam metode Crout, nilai a11 tidak boleh bernilai 0')
    disp(' ');
else
    %Iterasi 1
    l11 = a11;
    l21 = a21;
    l31 = a31;
    l41 = a41;

    %Iterasi 2
    u12 = a12/a11;
    u13 = a13/a11;
    u14 = a14/a11;

    %Iterasi 3
    l22 = a22 - l21*u12;
    l32 = a32 - l31*u12;
    l42 = a42 - l41*u12;

    %Iterasi 4
    u23 = (a23 - (l21*u13)) / l22;
    u24 = (a24 - (l21*u14)) / l22;

    %Iterasi 5
    l33 = a33 - l31*u13 - l32*u23;
    l43 = a43 - l41*u13 - l42*u23;

    %Iterasi 6
    u34 = (a34 - l31*u14 - l32*u24) / l33;

    %Iterasi 7
    l44 = a44 - l41*u14 - l42*u24 - l43*u34;

    %Determinan
    detL = l11*l22*l33*l44;
    detU = 1*1*1*1;

    detA = detL * detU;

    disp(' ')
    disp('Determinan Matriks A = ');
    disp(detA);
    disp(' ')
    fprintf('o11 = %f', o11)
end

```

Kode program Matlab perhitungan determinan matriks ordo 4x4 metode crout:

Perhitungan Determinan Matriks Ordo 4x4 MetodeDoolittle.

=====

```
Masukkan nilai a11: 2
Masukkan nilai a12: 1
Masukkan nilai a13: 6
Masukkan nilai a14: 7
Masukkan nilai a21: 3
Masukkan nilai a22: 2
Masukkan nilai a23: 4
Masukkan nilai a24: 5
Masukkan nilai a31: 4
Masukkan nilai a32: 4
Masukkan nilai a33: 2
Masukkan nilai a34: 3
Masukkan nilai a41: 5
Masukkan nilai a42: 6
Masukkan nilai a43: 1
Masukkan nilai a44: 4
```

Determinan Matriks A =
19.0000

d. Metode Doolittle

$$U_{ik} = a_{ij} - \sum_{k=1}^{j-1} l_{ik} u_{ik} \quad i \leq j, j = 2, 3, \dots, n$$

$$L_{ij} = \frac{a_{ij} - \sum_{k=1}^{j-1} l_{ik} u_{ik}}{u_{ii}} \quad j \leq i, i = 2, 3, \dots, n$$

Matriks Ordo 3 x 3

$$A = \begin{bmatrix} 3 & 1 & 2 \\ 2 & 1 & 3 \\ 1 & 3 & 1 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0 & 0 \\ l_{21} & 1 & 0 \\ l_{31} & l_{32} & 1 \end{bmatrix} \begin{bmatrix} u_{11} & u_{12} & u_{13} \\ 0 & u_{22} & u_{23} \\ 0 & 0 & u_{33} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

Rumus perhitungan:

Iterasi 1

$$u_{11} = a_{11} = 3$$

$$u_{12} = a_{12} = 1$$

$$u_{13} = a_{13} = 2$$

Iterasi 2

$$l_{21} = \frac{a_{21}}{a_{11}} = \frac{2}{3}$$

$$l_{31} = \frac{a_{31}}{a_{11}} = \frac{1}{3}$$

Iterasi 3

$$u_{22} = a_{22} - l_{21}u_{12}$$

$$u_{22} = 1 - \frac{2}{3} \cdot 1 = \frac{1}{3}$$

$$u_{23} = a_{23} - l_{21}u_{13}$$

$$u_{23} = 3 - \frac{2}{3} \cdot 2$$

$$u_{23} = \frac{9}{3} - \frac{4}{3} = \frac{5}{3}$$

Iterasi 4

$$l_{32} = \frac{a_{32} - l_{31}u_{12}}{u_{22}} = \frac{3 - \frac{1}{3} \cdot 1}{\frac{1}{3}} = \frac{8}{3} \times \frac{3}{1} = 8$$

Iterasi 5

$$u_{33} = a_{33} - l_{31}u_{13} - l_{32}u_{23}$$

$$u_{33} = 1 - \frac{1}{3} \cdot 2 - 8 \cdot \frac{5}{3} = 1 - \frac{2}{3} - \frac{40}{3} = \frac{3 - 2 - 40}{3} = \frac{-39}{3} = -13$$

$$\begin{bmatrix} 1 & 0 & 0 \\ \frac{2}{3} & 1 & 0 \\ \frac{1}{3} & 8 & 1 \end{bmatrix} \begin{bmatrix} 3 & 1 & 2 \\ 0 & \frac{1}{3} & \frac{5}{3} \\ 0 & 0 & -13 \end{bmatrix} = \begin{bmatrix} 3 & 1 & 2 \\ 2 & 1 & 3 \\ 1 & 3 & 1 \end{bmatrix}$$

$$\text{Det L} = 1 \times 1 \times 1 = 1$$

$$\text{Det U} = 3 \times \frac{1}{3} \times -13 = -13$$

$$\text{Jadi, Det A} = \text{Det L} \times \text{Det U} = 1 \times -13 = -13$$

Kode program Matlab perhitungan determinan matriks ordo 3x3 metode doolittle:

```
clc
```

```
disp('Perhitungan Determinan Matriks Ordo 3x3 Metode Doolittle.')
disp('=====')
disp(' ')
a11 = input('Masukkan nilai a11: ');
a12 = input('Masukkan nilai a12: ');
a13 = input('Masukkan nilai a13: ');
```

```

a21 = input('Masukkan nilai a21: ');
a22 = input('Masukkan nilai a22: ');
a23 = input('Masukkan nilai a23: ');

a31 = input('Masukkan nilai a31: ');
a32 = input('Masukkan nilai a32: ');
a33 = input('Masukkan nilai a33: ');

if a11 == 0
    disp(' ')
    disp('Nilai determinan = 0.')
    disp(' ')

    %disp('Dalam metode Crout, nilai a11 tidak boleh bernilai 0')
    disp(' ');
else
    %Iterasi 1
    u11 = a11;
    u12 = a12;
    u13 = a13;

    %Iterasi 2
    l21 = a21/a11;
    l31 = a31/a11;

    %Iterasi 3
    u22 = a22 - l21*u12;
    u23 = a23 - l21*u13;

    %Iterasi 4
    l32 = (a32 - (l31*u12)) / u22;

    %Iterasi 5
    u33 = a33 - l31*u13 - l32*u23;

    %Determinan
    detL = 1*1*1*1;
    detU = u11*u22*u33;

    detA = detL * detU;

    disp(' ')
    disp('Determinan Matriks A = ');
    disp(detA);
    disp(' ')
    %fprintf('o11 = %f', o11)
end

```

Tampilan *output* program Matlab perhitungan determinan matriks ordo 3x3 metode doolittle:

```

Perhitungan Determinan Matriks Ordo 3x3 Metode Doolittle.
=====

```

```

Masukkan nilai a11: 3
Masukkan nilai a12: 1
Masukkan nilai a13: 2
Masukkan nilai a21: 2

```

Masukkan nilai a_{22} : 1
 Masukkan nilai a_{23} : 3
 Masukkan nilai a_{31} : 1
 Masukkan nilai a_{32} : 3
 Masukkan nilai a_{33} : 1

Determinan Matriks A =
 -13.0000

Matriks Ordo 4 x 4

$$A = \begin{bmatrix} 2 & 1 & 6 & 7 \\ 3 & 2 & 4 & 5 \\ 4 & 4 & 2 & 3 \\ 5 & 6 & 1 & 4 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix},$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ l_{21} & 1 & 0 & 0 \\ l_{31} & l_{32} & 1 & 0 \\ l_{41} & l_{42} & l_{43} & 1 \end{bmatrix} \begin{bmatrix} u_{11} & u_{12} & u_{13} & u_{14} \\ 0 & u_{22} & u_{23} & u_{24} \\ 0 & 0 & u_{33} & u_{34} \\ 0 & 0 & 0 & u_{44} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix}$$

Rumus perhitungan:

Iterasi 1

$$u_{11} = a_{11} = 2 ;$$

$$u_{12} = a_{12} = 1$$

$$u_{13} = a_{13} = 6$$

$$u_{14} = a_{14} = 7$$

Iterasi 2

$$l_{21} = \frac{a_{21}}{a_{11}} = \frac{3}{2}$$

$$l_{31} = \frac{a_{31}}{a_{11}} = \frac{4}{2} = 2$$

$$l_{41} = \frac{a_{41}}{a_{11}} = \frac{5}{2}$$

Iterasi 3

$$u_{22} = a_{22} - l_{21}u_{12}$$

$$u_{22} = 2 - \frac{3}{2} \cdot 1 = \frac{1}{2}$$

$$u_{23} = a_{23} - l_{21}u_{13}$$

$$u_{23} = 4 - \frac{3}{2} \cdot 6 = 4 - 9 = -5$$

$$u_{24} = a_{24} - l_{21}u_{14}$$

$$u_{24} = 5 - \frac{3}{2} \cdot 7 = \frac{10 - 21}{2} = -\frac{11}{2}$$

Iterasi 4

$$l_{32} = \frac{a_{32} - l_{31}u_{12}}{u_{22}}$$

$$l_{32} = \frac{4 - 2 \cdot 1}{\frac{1}{2}} = 4$$

$$l_{42} = \frac{a_{42} - l_{41}u_{12}}{u_{22}}$$

$$l_{42} = \frac{6 - \frac{5}{2} \cdot 1}{\frac{1}{2}} = \frac{\frac{7}{2}}{\frac{1}{2}} = 7$$

Iterasi 5

$$u_{33} = a_{33} - l_{31}u_{13} - l_{32}u_{23}$$

$$u_{33} = 2 - 2 \cdot 6 - 4 \cdot (-5)$$

$$u_{33} = 2 - 12 + 20$$

$$u_{33} = 10$$

$$u_{34} = a_{34} - l_{31}u_{14} - l_{32}u_{24}$$

$$u_{34} = 3 - 2 \cdot 7 - 4 \cdot \left(-\frac{11}{2}\right)$$

$$u_{34} = 3 - 14 + 22 = 11$$

Iterasi 6

$$l_{43} = \frac{a_{43} - l_{41}u_{13} - l_{42}u_{23}}{u_{33}}$$

$$l_{43} = \frac{1 - \frac{5}{2} \cdot 6 - 7(-5)}{10}$$

$$l_{43} = \frac{1 - \frac{5}{2} \cdot 6 - 7(-5)}{10}$$

$$l_{43} = \frac{1 - 15 + 35}{10}$$

$$l_{43} = \frac{21}{10}$$

Iterasi 7

$$u_{44} = a_{44} - l_{41}u_{14} - l_{42}u_{24} - l_{43}u_{34}$$

$$u_{44} = 4 - \frac{5}{2} \cdot 7 - 7 \cdot \left(-\frac{11}{2}\right) - \frac{21}{10} \cdot 11$$

$$u_{44} = 4 - \frac{35}{2} + \frac{77}{2} - \frac{21}{10} \cdot 11$$

$$u_{44} = 4 + 21 - \frac{21}{10} \cdot 11$$

$$u_{44} = 25 - \frac{231}{10}$$

$$u_{44} = \frac{250}{10} - \frac{231}{10}$$

$$u_{44} = \frac{19}{10}$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ l_{21} & 1 & 0 & 0 \\ l_{31} & l_{32} & 1 & 0 \\ l_{41} & l_{42} & l_{43} & 1 \end{bmatrix} \begin{bmatrix} 2 & u_{12} & u_{13} & u_{14} \\ 0 & \frac{1}{2} & u_{23} & u_{24} \\ 0 & 0 & 10 & u_{34} \\ 0 & 0 & 0 & \frac{19}{10} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix}$$

$$\text{Det L} = 1 \times 1 \times 1 = 1$$

$$\text{Det U} = 2 \times \frac{1}{2} \times 10 \times \frac{19}{10} = 19$$

$$\text{Det A} = \text{Det L} \times \text{Det U} = 1 \times 19 = 19$$

Kode Program Matlab:

```
clc

disp('Perhitungan Determinan Matriks Ordo 4x4 MetodeDoolittle.')
disp('=====')
disp(' ')
a11 = input('Masukkan nilai a11: ');
a12 = input('Masukkan nilai a12: ');
a13 = input('Masukkan nilai a13: ');
a14 = input('Masukkan nilai a14: ');

a21 = input('Masukkan nilai a21: ');
a22 = input('Masukkan nilai a22: ');
a23 = input('Masukkan nilai a23: ');
a24 = input('Masukkan nilai a24: ');

a31 = input('Masukkan nilai a31: ');
a32 = input('Masukkan nilai a32: ');
a33 = input('Masukkan nilai a33: ');
a34 = input('Masukkan nilai a34: ');

a41 = input('Masukkan nilai a41: ');
a42 = input('Masukkan nilai a42: ');
a43 = input('Masukkan nilai a43: ');
a44 = input('Masukkan nilai a44: ');
```

```

if a11 == 0
    disp(' ')
    disp('Nilai determinan = 0.')
    disp(' ')

    %disp('Dalam metode Crout, nilai a11 tidak boleh bernilai 0')
    disp(' ');
else
    %Iterasi 1
    u11 = a11;
    u12 = a12;
    u13 = a13;
    u14 = a14;

    %Iterasi 2
    l21 = a21/a11;
    l31 = a31/a11;
    l41 = a41/a11;

    %Iterasi 3
    u22 = a22 - l21*u12;
    u23 = a23 - l21*u13;
    u24 = a24 - l21*u14;

    %Iterasi 4
    l32 = (a32 - (l31*u12)) / u22;
    l42 = (a42 - (l41*u12)) / u22;

    %Iterasi 5
    u33 = a33 - l31*u13 - l32*u23;
    u34 = a34 - l31*u14 - l32*u24;

    %Iterasi 6
    l43 = (a43 - l41*u13 - l42*u23) / u33;

    %Iterasi 7
    u44 = a44 - l41*u14 - l42*u24 - l43*u34;

    %Determinan
    detL = 1*1*1*1;
    detU = u11*u22*u33*u44;

    detA = detL * detU;

    disp(' ')
    disp('Determinan Matriks A = ');
    disp(detA);
    disp(' ')
    %fprintf('o11 = %f', o11)
End

```

Tampilan *output* program Matlab perhitungan determinan matriks ordo 4x4 metode doolittle:

```

Perhitungan Determinan Matriks Ordo 4x4 MetodeDoolittle.
=====

```

Masukkan nilai a11: 2
 Masukkan nilai a12: 1
 Masukkan nilai a13: 6
 Masukkan nilai a14: 7
 Masukkan nilai a21: 3
 Masukkan nilai a22: 2
 Masukkan nilai a23: 4
 Masukkan nilai a24: 5
 Masukkan nilai a31: 4
 Masukkan nilai a32: 4
 Masukkan nilai a33: 2
 Masukkan nilai a34: 3
 Masukkan nilai a41: 5
 Masukkan nilai a42: 6
 Masukkan nilai a43: 1
 Masukkan nilai a44: 4

Determinan Matriks A =
 19.0000

6. Invers Suatu Matriks

Jika A adalah sebuah matriks bujur sangkar, dan jika matriks B yang berukuran sama bisa didapatkan sedemikian hingga memenuhi:

$$AB = BA = I$$

dimana I adalah matriks identitas, maka A disebut dapat dibalik (invertibel). Untuk selanjutnya invers dari matriks A dinyatakan dengan symbol A^{-1}

Metode Biasa Invers matriks Ordo 2 x 2

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

Matriks A mempunyai invers jika $ad - bc \neq 0$, dimana inversnya dapat ditentukan dengan rumus:

$$A^{-1} = \frac{1}{ad - bc} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix} = \begin{bmatrix} \frac{d}{d - bc} & \frac{-b}{d - bc} \\ \frac{-c}{d - bc} & \frac{a}{d - bc} \end{bmatrix}$$

Contoh:

Diberikan matriks A dan B sebagai berikut:

$$A = \begin{bmatrix} 1 & 2 \\ 1 & 3 \end{bmatrix}, \quad B = \begin{bmatrix} 3 & 2 \\ 2 & 2 \end{bmatrix}$$

Tentukan:

A^{-1} dan B^{-1}

Jawab:

$$\text{Det (A)} = ad - bc = 1 \cdot 3 - 2 \cdot 1 = 3 - 2 = 1$$

$$\text{Det (B)} = ad - bc = 3 \cdot 2 - 2 \cdot 2 = 6 - 4 = 2$$

$$A^{-1} = \frac{1}{1} \begin{bmatrix} 3 & -2 \\ -1 & 1 \end{bmatrix} = \begin{bmatrix} 3 & -2 \\ -1 & 1 \end{bmatrix}$$

$$B^{-1} = \frac{1}{2} \begin{bmatrix} 2 & -2 \\ -2 & 3 \end{bmatrix} = \begin{bmatrix} 1 & -1 \\ -1 & \frac{3}{2} \end{bmatrix}$$

Metode Biasa Invers matriks Ordo 3 x 3

Jika $|A| \neq 0$ maka invers dari A didefinisikan:

$$A^{-1} = \frac{\text{adj}(A)}{|A|}$$

Contoh:

Diberikan matriks A sebagai berikut:

$$A = \begin{bmatrix} 3 & 2 & -1 \\ 1 & 6 & 3 \\ 2 & -4 & 0 \end{bmatrix}$$

Tentukan invers dari A matriks tersebut:

Langkah pertama: Mencari determinan

$\det A = a_{11} C_{11} + a_{12} C_{12} + a_{13} C_{13}$ (ekspansi kofaktor baris pertama)

Minor dari entri a_{11} adalah

$$M_{11} = \begin{bmatrix} 3 & 2 & -1 \\ 1 & 6 & 3 \\ 2 & -4 & 0 \end{bmatrix} = \begin{bmatrix} 6 & 3 \\ -4 & 0 \end{bmatrix} = 0 - (-12) = 12$$

Kofaktor dari a_{11} adalah

$$C_{11} = (-1)^{1+1} M_{11} = M_{11} = 12$$

Minor dari entri a_{12} adalah

$$M_{12} = \begin{bmatrix} 3 & 2 & -1 \\ 1 & 6 & 3 \\ 2 & -4 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 3 \\ 2 & 0 \end{bmatrix} = 0 - 6 = -6$$

Kofaktor dari a_{12} adalah

$$C_{12} = (-1)^{1+2} M_{12} = - M_{12} = -(-6) = 6$$

Minor dari entri a_{13} adalah

$$M_{13} = \begin{bmatrix} 3 & 2 & -1 \\ 1 & 6 & 3 \\ 2 & -4 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 6 \\ 2 & -4 \end{bmatrix} = -4 - 12 = -16$$

Kofaktor dari a_{13} adalah

$$C_{13} = (-1)^{1+3} M_{13} = M_{13} = -16$$

$$\det A = 3 \cdot 12 + 2 \cdot 6 + -1 \cdot (-16) = 36 + 12 + 16 = 64$$

jadi nilai $\det(A)$ dengan menggunakan ekspansi kofaktor baris pertama adalah 64

Langkah kedua:

Mencari Adj (A)

Minor dari entri a_{11} adalah

$$M_{11} = \begin{bmatrix} 3 & 2 & -1 \\ 1 & 6 & 3 \\ 2 & -4 & 0 \end{bmatrix} = \begin{bmatrix} 6 & 3 \\ -4 & 0 \end{bmatrix} = 0 - (-12) = 12$$

Kofaktor dari a_{11} adalah

$$C_{11} = (-1)^{1+1} M_{11} = M_{11} = 12$$

Dengan cara sama untuk mencari kofaktor lainnya dari matriks tersebut. Sehingga diperoleh:

$$C_{11} = 12 \qquad C_{12} = 6 \text{ (buktikan)} \qquad C_{13} = -16 \text{ (buktikan)}$$

$$C_{21} = 4 \text{ (buktikan)} \qquad C_{22} = 2 \text{ (buktikan)} \qquad C_{23} = 16 \text{ (buktikan)}$$

$$C_{31} = 12 \text{ (buktikan)} \qquad C_{32} = -10 \text{ (buktikan)} \qquad C_{33} = 16 \text{ (buktikan)}$$

Sehingga matriks kofaktornya adalah:

$$\begin{bmatrix} 12 & 6 & -16 \\ 4 & 2 & 16 \\ 12 & -10 & 16 \end{bmatrix}$$

dan adjoint (A) adalah:

$$\text{Adj}(A) = \begin{bmatrix} 12 & 6 & -16 \\ 4 & 2 & 16 \\ 12 & -10 & 16 \end{bmatrix}^T = \begin{bmatrix} 12 & 4 & 12 \\ 6 & 2 & -10 \\ -16 & 16 & 16 \end{bmatrix}$$

Langkah Ketiga:

$$A^{-1} = \frac{1}{|A|} \text{adj}(A) = \frac{1}{64} \begin{bmatrix} 12 & 4 & 12 \\ 6 & 2 & -10 \\ -16 & 16 & 16 \end{bmatrix} = \begin{bmatrix} \frac{12}{64} & \frac{4}{64} & \frac{12}{64} \\ \frac{6}{64} & \frac{2}{64} & \frac{-10}{64} \\ \frac{-16}{64} & \frac{16}{64} & \frac{16}{64} \end{bmatrix}$$

$$\text{Matriks Identitas Orde } = \begin{bmatrix} 1_a & 0_c \\ 0_b & 1_d \end{bmatrix}$$

Tentukan Invers Matriks $A = \begin{bmatrix} 2 & 1 \\ 5 & 3 \end{bmatrix}$ dengan transformasi baris elementer

$$A = \left(\begin{array}{cc|cc} 2 & 1 & 1 & 0 \\ 5 & 3 & 0 & 1 \end{array} \right)$$

Langkah 1 a11 Wajib bernilai 1 maka :

$$\frac{1}{2}B_1 = \left(\begin{array}{cc|cc} 2 & 1 & 1 & 0 \\ 5 & 3 & 0 & 1 \end{array} \right) = \left(\begin{array}{cc|cc} 1 & \frac{1}{2} & \frac{1}{2} & 0 \\ 5 & 3 & 0 & 1 \end{array} \right)$$

$$B_{11} =$$

$$a_{11b} = 1/a_{11a}$$

$$a_{12b} = (1/a_{11}) * a_{12a}$$

$$i_{o11b} = (1/a_{11}) * i_{o11a}$$

$$i_{o12b} = (1/a_{11}) * i_{o12a}$$

Langkah 2 a21 Wajib bernilai 0 maka :

$$B_2 - 5B_1 = \left(\begin{array}{cc|cc} 1 & \frac{1}{2} & \frac{1}{2} & 0 \\ 5 & 3 & 0 & 1 \end{array} \right) = \left(\begin{array}{cc|cc} 1 & \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & \frac{1}{2} & -\frac{5}{2} & 1 \end{array} \right)$$

Langkah 3 a12 Wajib bernilai 0 maka :

$$B_1 - B_2 = \left(\begin{array}{cc|cc} 1 & \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & \frac{1}{2} & -\frac{5}{2} & 1 \end{array} \right) = \left(\begin{array}{cc|cc} 1 & 0 & 3 & -1 \\ 0 & \frac{1}{2} & -\frac{5}{2} & 1 \end{array} \right)$$

Langkah 4 a22 Wajib bernilai 1 maka :

$$2B_2 = \left(\begin{array}{cc|cc} 1 & 0 & 3 & -1 \\ 0 & \frac{1}{2} & -\frac{5}{2} & 1 \end{array} \right) = \left(\begin{array}{cc|cc} 1 & 0 & 3 & -1 \\ 0 & 1 & -5 & 2 \end{array} \right)$$

$$A^{-1} = \begin{bmatrix} 3 & -1 \\ -5 & 2 \end{bmatrix}$$

Kode Program Matlab Menghitung Invers Matrix Ordo 2x2 OBE:

```
clc
```

```
disp('Perhitungan Invers Matriks Ordo 2x2 OBE.')
disp('By: Havizul')
disp('=====')
disp(' ')
```

```

a11 = input('Masukkan nilai a11: ');
a12 = input('Masukkan nilai a12: ');
a21 = input('Masukkan nilai a21: ');
a22 = input('Masukkan nilai a22: ');
o11 = 1;
o12 = 0;
o21 = 0;
o22 = 1;

x = 1/a11;
a11 = x*a11;
a12 = x*a12;
o11 = x*o11;
o12 = x*o12;

x = a21;
a21 = a21 - x;
a22 = a22 - (x*a12);
o21 = o21 - (x*o11);
o22 = o22 - (x*o12);

x = a12/a22;
a12 = a12 - (x*a22);
o11 = o11 - (x*o21);
o12 = o12 - (x*o22);

x = 1/a22;
a22 = x*a22;
o21 = x*o21;
o22 = x*o22;

disp(' ')
disp('Nilai Invers Matriks : ')
disp(' ')
fprintf('o11 = %f', o11)
disp(' ')
fprintf('o12 = %f', o12)
disp(' ')
fprintf('o21 = %f', o21)
disp(' ')
fprintf('o22 = %f', o22)
disp(' ')

```

Tampilan Output Program Matlab Menghitung Invers Matrix Ordo 2x2 OBE:

Perhitungan Invers Matriks Ordo 2x2 OBE.
By: Havizul

=====

```

Masukkan nilai a11: 2
Masukkan nilai a12: 1
Masukkan nilai a21: 5
Masukkan nilai a22: 3

```

```

Nilai Invers Matriks :
o11 = 3.000000
o12 = -1.000000
o21 = -5.000000
o22 = 2.000000

```


$$\text{Matriks Identitas Orde } 3 \times 3 = \begin{bmatrix} 1_a & 0_e & 0_i \\ 0_b & 1_d & 0_h \\ 0_c & 0_f & 1_g \end{bmatrix}$$

$$3 \times 3 = \begin{bmatrix} 1_a & 0_e & 0_h \\ 0_b & 1_d & 0_i \\ 0_c & 0_f & 1_g \end{bmatrix}$$

Tentukan Invers Matriks $A = \begin{bmatrix} 1 & 1 & 0 \\ 2 & 3 & 2 \\ 2 & 1 & 3 \end{bmatrix}$ dengan transformasi baris elementer

$$A = \left(\begin{array}{ccc|ccc} 1 & 1 & 0 & 1 & 0 & 0 \\ 2 & 3 & 2 & 0 & 1 & 0 \\ 2 & 1 & 3 & 0 & 0 & 1 \end{array} \right)$$

Langkah 1 a11 Wajib bernilai 1, pada soal a11 sudah bernilai 1 maka

$$\text{lanjut ke langkah berikutnya } \left(\begin{array}{ccc|ccc} 1 & 1 & 0 & 1 & 0 & 0 \\ 2 & 3 & 2 & 0 & 1 & 0 \\ 2 & 1 & 3 & 0 & 0 & 1 \end{array} \right)$$

Langkah 2 a21 Wajib bernilai 0 maka :

$$B_2 - 2B_1 = \left(\begin{array}{ccc|ccc} 1 & 1 & 0 & 1 & 0 & 0 \\ 2 & 3 & 2 & 0 & 1 & 0 \\ 2 & 1 & 3 & 0 & 0 & 1 \end{array} \right) = \left(\begin{array}{ccc|ccc} 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 2 & -2 & 1 & 0 \\ 2 & 1 & 3 & 0 & 0 & 1 \end{array} \right)$$

Langkah 3 a31 Wajib bernilai 0 maka :

$$B_3 - 2B_1 = \left(\begin{array}{ccc|ccc} 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 2 & -2 & 1 & 0 \\ 2 & 1 & 3 & 0 & 0 & 1 \end{array} \right) = \left(\begin{array}{ccc|ccc} 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 2 & -2 & 1 & 0 \\ 0 & -1 & 3 & -2 & 0 & 1 \end{array} \right)$$

Langkah 4 a22 Wajib bernilai 1 maka pada soal a22 sudah bernilai 1 maka

$$\text{lanjut ke langkah berikutnya } \left(\begin{array}{ccc|ccc} 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 2 & -2 & 1 & 0 \\ 0 & -1 & 3 & -2 & 0 & 1 \end{array} \right)$$

Langkah 5 a12 Wajib bernilai 0 maka :

$$B_1 - B_2 = \left(\begin{array}{ccc|ccc} 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 2 & -2 & 1 & 0 \\ 0 & -1 & 3 & -2 & 0 & 1 \end{array} \right) = \left(\begin{array}{ccc|ccc} 1 & 0 & -2 & 3 & -1 & 0 \\ 0 & 1 & 2 & -2 & 1 & 0 \\ 0 & -1 & 3 & -2 & 0 & 1 \end{array} \right)$$

Langkah 6 a32 Wajib bernilai 0 maka :

$$B_3 + B_2 = \left(\begin{array}{ccc|ccc} 1 & 0 & -2 & 3 & -1 & 0 \\ 0 & 1 & 2 & -2 & 1 & 0 \\ 0 & -1 & 3 & -2 & 0 & 1 \end{array} \right) = \left(\begin{array}{ccc|ccc} 1 & 0 & -2 & 3 & -1 & 0 \\ 0 & 1 & 2 & -2 & 1 & 0 \\ 0 & 0 & 5 & -4 & 1 & 1 \end{array} \right)$$

Langkah 7 a33 Wajib bernilai 1 maka :

$$\frac{1}{5}B_3 = \left(\begin{array}{ccc|ccc} 1 & 0 & -2 & 3 & -1 & 0 \\ 0 & 1 & 2 & -2 & 1 & 0 \\ 0 & 0 & 5 & -4 & 1 & 1 \end{array} \right) = \left(\begin{array}{ccc|ccc} 1 & 0 & -2 & 3 & -1 & 0 \\ 0 & 1 & 2 & -2 & 1 & 0 \\ 0 & 0 & 1 & -\frac{4}{5} & \frac{1}{5} & \frac{1}{5} \end{array} \right)$$

Langkah 8 a13 Wajib bernilai 0 maka :

$$B_1 + 2B_3 = \left(\begin{array}{ccc|ccc} 1 & 0 & -2 & 3 & -1 & 0 \\ 0 & 1 & 2 & -2 & 1 & 0 \\ 0 & 0 & 1 & -\frac{4}{5} & \frac{1}{5} & \frac{1}{5} \end{array} \right) = \left(\begin{array}{ccc|ccc} 1 & 0 & 0 & \frac{7}{5} & -\frac{3}{5} & \frac{2}{5} \\ 0 & 1 & 2 & -2 & 1 & 0 \\ 0 & 0 & 1 & -\frac{4}{5} & \frac{1}{5} & \frac{1}{5} \end{array} \right)$$

Langkah 9 a23 Wajib bernilai 0 maka :

$$B_2 - 2B_3 = \left(\begin{array}{ccc|ccc} 1 & 0 & 0 & \frac{7}{5} & -\frac{3}{5} & \frac{2}{5} \\ 0 & 1 & 2 & -2 & 1 & 0 \\ 0 & 0 & 1 & -\frac{4}{5} & \frac{1}{5} & \frac{1}{5} \end{array} \right) = \left(\begin{array}{ccc|ccc} 1 & 0 & 0 & \frac{7}{5} & -\frac{3}{5} & \frac{2}{5} \\ 0 & 1 & 0 & -\frac{8}{5} & \frac{3}{5} & \frac{2}{5} \\ 0 & 0 & 1 & -\frac{4}{5} & \frac{1}{5} & \frac{1}{5} \end{array} \right)$$

$$A^{-1} = \left(\begin{array}{ccc|ccc} 1 & 0 & 0 & \frac{7}{5} & -\frac{3}{5} & \frac{2}{5} \\ 0 & 1 & 0 & -\frac{8}{5} & \frac{3}{5} & \frac{2}{5} \\ 0 & 0 & 1 & -\frac{4}{5} & \frac{1}{5} & \frac{1}{5} \end{array} \right)$$

Kode Program Matlab Menghitung Invers Matriks Ordo 3x3 OBE:

```
clc

disp('Perhitungan Invers Matriks Ordo 3x3 OBE.')
disp('=====')
disp(' ')
%input nilai matriks 3x3
a11 = input('Masukkan nilai a11: ');
a12 = input('Masukkan nilai a12: ');
a13 = input('Masukkan nilai a13: ');
a21 = input('Masukkan nilai a21: ');
a22 = input('Masukkan nilai a22: ');
a23 = input('Masukkan nilai a23: ');
a31 = input('Masukkan nilai a31: ');
a32 = input('Masukkan nilai a32: ');
a33 = input('Masukkan nilai a33: ');

%format matriks OBE 3x3
o11 = 1;
o12 = 0;
o13 = 0;
o21 = 0;
o22 = 1;
o23 = 0;
o31 = 0;
o32 = 0;
o33 = 1;

%langkah 1, a11 wajib bernilai 1
x = a11;
a11 = a11/x;
a12 = a12/x;
a13 = a13/x;
o11 = o11/x;
o12 = o12/x;
o13 = o13/x;

%langkah 2, a21 wajib bernilai 0
x = a21;
a21 = a21 - (x*a11);
```

```
a22 = a22 - (x*a12);  
a23 = a23 - (x*a13);  
o21 = o21 - (x*o11);  
o22 = o22 - (x*o12);  
o23 = o23 - (x*o13);
```

%langkah 3, a31 wajib bernilai 0

```
x = a31;  
a31 = a31 - (x*a11);  
a32 = a32 - (x*a12);  
a33 = a33 - (x*a13);  
o31 = o31 - (x*o11);  
o32 = o32 - (x*o12);  
o33 = o33 - (x*o13);
```

%langkah 4, a22 wajib bernilai 1

```
x = a22;  
a22 = a22/x;  
a23 = a23/x;  
o21 = o21/x;  
o22 = o22/x;  
o23 = o23/x;
```

%langkah 5, a12 wajib bernilai 0

```
x = a12;  
a12 = a12 - (x*a22);  
a13 = a13 - (x*a23);  
o11 = o11 - (x*o21);  
o12 = o12 - (x*o22);  
o13 = o13 - (x*o23);
```

%langkah 6, a32 wajib bernilai 0

```
x = a32;  
a32 = a32 - (x*a22);  
a33 = a33 - (x*a23);  
o31 = o31 - (x*o21);  
o32 = o32 - (x*o22);  
o33 = o33 - (x*o23);
```

%langkah 7, a33 wajib bernilai 1

```
x = a33;  
a33 = a33/x;  
o31 = o31/x;  
o32 = o32/x;  
o33 = o33/x;
```

%langkah 8, a13 wajib bernilai 0

```
x = a13;  
a13 = a13 - (x*a33);  
o11 = o11 - (x*o31);  
o12 = o12 - (x*o32);  
o13 = o13 - (x*o33);
```

%langkah 9, a23 wajib bernilai 0

```
x = a23;  
a23 = a23 - (x*a33);  
o21 = o21 - (x*o31);  
o22 = o22 - (x*o32);  
o23 = o23 - (x*o33);
```

```

%Tampilkan nilai Invers Matriks
disp(' ')
disp('Invers Matriks OBE: ')
disp(' ')
fprintf('o11 = %f', o11)
disp(' ')
fprintf('o12 = %f', o12)
disp(' ')
fprintf('o13 = %f', o13)
disp(' ')
fprintf('o21 = %f', o21)
disp(' ')
fprintf('o22 = %f', o22)
disp(' ')
fprintf('o23 = %f', o23)
disp(' ')
fprintf('o31 = %f', o31)
disp(' ')
fprintf('o32 = %f', o32)
disp(' ')
fprintf('o33 = %f', o33)
disp(' ')

```

Tampilan *Output* Program Matlab Menghitung Invers Matriks Ordo 3x3 OBE:

Perhitungan Invers Matriks Ordo 3x3 OBE.

=====

```

Masukkan nilai a11: 1
Masukkan nilai a12: 1
Masukkan nilai a13: 0
Masukkan nilai a21: 2
Masukkan nilai a22: 3
Masukkan nilai a23: 2
Masukkan nilai a31: 2
Masukkan nilai a32: 1
Masukkan nilai a33: 3

```

Inverse Matriks OBE:

```

o11 = 1.400000
o12 = -0.600000
o13 = 0.400000
o21 = -0.400000
o22 = 0.600000
o23 = -0.400000
o31 = -0.800000
o32 = 0.200000
o33 = 0.200000

```

SISTEM PERSAMAAN LINEAR

1. Metode Eliminasi Gauss

Tentukan nilai x , y dan z pada persamaan linear tiga variabel berikut:

$$x + 2y - 3z = 1$$

$$2x + 5y - 8z = 4$$

$$3x + 8y - 14z = 8$$

Masukkan nilai konstanta pada persamaan tersebut kedalam matriks :

$$\left(\begin{array}{ccc|c} 1 & 2 & -3 & 1 \\ 2 & 5 & -8 & 4 \\ 3 & 8 & -14 & 8 \end{array} \right)$$

Langkah 1 a11 Wajib bernilai 1, pada soal a11 sudah bernilai 1 maka

lanjut ke langkah berikutnya $\left(\begin{array}{ccc|c} 1 & 2 & -3 & 1 \\ 2 & 5 & -8 & 4 \\ 3 & 8 & -14 & 8 \end{array} \right)$

Langkah 2 a21, Wajib bernilai 0 maka :

$$B_2 - 2B_1 = \left(\begin{array}{ccc|c} 1 & 2 & -3 & 1 \\ 2 & 5 & -8 & 4 \\ 3 & 8 & -14 & 8 \end{array} \right) = \left(\begin{array}{ccc|c} 1 & 2 & -3 & 1 \\ 0 & 1 & -2 & 2 \\ 3 & 8 & -14 & 8 \end{array} \right)$$

Langkah 3 a31, Wajib bernilai 0 maka :

$$B_3 - 3B_1 = \left(\begin{array}{ccc|c} 1 & 2 & -3 & 1 \\ 0 & 1 & -2 & 2 \\ 3 & 8 & -14 & 8 \end{array} \right) = \left(\begin{array}{ccc|c} 1 & 2 & -3 & 1 \\ 0 & 1 & -2 & 2 \\ 0 & 2 & -5 & 5 \end{array} \right)$$

Langkah 4 a22, Wajib bernilai 1 maka pada soal a22 sudah bernilai 1 maka

lanjut ke langkah berikutnya $\left(\begin{array}{ccc|c} 1 & 2 & -3 & 1 \\ 0 & 1 & -2 & 2 \\ 0 & 2 & -5 & 5 \end{array} \right)$

Langkah 5 a32, Wajib bernilai 0 maka :

$$B_3 - 2B_2 = \left(\begin{array}{ccc|c} 1 & 2 & -3 & 1 \\ 0 & 1 & -2 & 2 \\ 0 & 2 & -5 & 5 \end{array} \right) = \left(\begin{array}{ccc|c} 1 & 2 & -3 & 1 \\ 0 & 1 & -2 & 2 \\ 0 & 0 & -1 & 1 \end{array} \right)$$

Langkah 6 a33, Wajib bernilai 1 maka :

$$-B_3 = \left(\begin{array}{ccc|c} 1 & 2 & -3 & 1 \\ 0 & 1 & -2 & 2 \\ 0 & 0 & -1 & 1 \end{array} \right) = \left(\begin{array}{ccc|c} 1 & 2 & -3 & 1 \\ 0 & 1 & -2 & 2 \\ 0 & 0 & 1 & -1 \end{array} \right)$$

Dari hasil perhitungan di peroleh gambaran:

$$x + 2y - 3z = 1 \quad \text{Pers 1}$$

$$y - 2z = 2 \quad \text{Pers 2}$$

$$z = -1$$

Subtitusi nilai z ke persamaan (2):

$$y - 2z = 2$$

$$y + 2 = 2$$

$$y = 0$$

subtitusi nilai y dan z ke persamaan (1)

$$x + 2y - 3z = 1$$

$$x + 2(0) - 3(-1) = 1$$

$$x + 0 + 3 = 1$$

$$x + 3 = 1$$

$$x = -2$$

jadi, nilai x, y dan z berturut - turut adalah

$$x = -2 \quad y = 0 \quad z = -1$$

Kode Program Matlab Penyelesaian SPLTV Metode Eliminasi Gauss:

```

clc

disp('Perhitungan Persamaan Linear XYZ Metode Eliminasi Gauss.')
disp('Bentuk persamaan: ')
disp('ax + by + cz = d1')
disp('by + cz = d2')
disp('cz = d3')
disp('=====')
disp(' ')

a1 = input('Masukkan nilai a1: ');
a2 = input('Masukkan nilai a2: ');
a3 = input('Masukkan nilai a3: ');
b1 = input('Masukkan nilai b1: ');
b2 = input('Masukkan nilai b2: ');
b3 = input('Masukkan nilai b3: ');
c1 = input('Masukkan nilai c1: ');
c2 = input('Masukkan nilai c2: ');
c3 = input('Masukkan nilai c3: ');
d1 = input('Masukkan nilai d1: ');
d2 = input('Masukkan nilai d2: ');
d3 = input('Masukkan nilai d3: ');

%Ubah notasi persamaan linear menjadi notasi matriks,

```

```
%untuk mempermudah penulisan kode program agar sinkron
%dengan notasi2 kode program sebelumnya
a11 = a1;
a12 = b1;
a13 = c1;
o11 = d1;
a21 = a2;
a22 = b2;
a23 = c2;
o21 = d2;
a31 = a3;
a32 = b3;
a33 = c3;
o31 = d3;

if a11 == 0
    disp(' ')
    disp('Tukar Baris')
    disp(' ')

    disp(' ');
else
    %a11 wajib bernilai 1
    x = 1/a11;
    a11 = x*a11;
    a12 = x*a12;
    a13 = x*a13;
    o11 = x*o11;
    %ubah nilai a1, b1, c1, dan d1
    a1 = a11;
    b1 = a12;
    c1 = a13;
    d1 = o11;

    %a21 wajib bernilai 0
    x = a21;
    a21 = a21 - (x*a11);
    a22 = a22 - (x*a12);
    a23 = a23 - (x*a13);
    o21 = o21 - (x*o11);

    %a31 wajib bernilai 0
    x = a31;
    a31 = a31 - (x*a11);
    a32 = a32 - (x*a12);
    a33 = a33 - (x*a13);
    o31 = o31 - (x*o11);

    %a22 wajib bernilai 1
    x = a22;
    a22 = a22/x;
    a23 = a23/x;
    o21 = o21/x;
    %ubah nilai b2, c2, dan d2
    b2 = a22;
    c2 = a23;
    d2 = o21;

    %a32 wajib bernilai 0
    x = a32;
```

```

a32 = a32 - (x*a22);
a33 = a33 - (x*a23);
o31 = o31 - (x*o21);

%a33 wajib bernilai 1
x = a33;
a33 = a33/x;
o31 = o31/x;
%ubah nilai c3 dan d3
c3 = a33;
d3 = o31;

%Dapatkan nilai z untuk memulai substitusi
z = d3/c3;

%Substitusi nilai z ke persamaan 2
%y = persDua(z);
y = (d2-(c2*z))/b2;

%Substitusi nilai y dan z ke persamaan 1
%x = persSatu(y, z);
x = (d1 - (b1*y) - (c1*z))/a1;

fprintf('Nilai x = %f', x);
disp(' ');
fprintf('Nilai y = %f', y);
disp(' ');
fprintf('Nilai z = %f', z);
disp(' ');
end

```

Tampilan *Output* Program Matlab Penyelesaian SPLTV Metode Eliminasi Gauss:

Perhitungan Persamaan Linear XYZ Metode Eliminasi Gauss.

Bentuk persamaan:

$$ax + by + cz = d1$$

$$by + cz = d2$$

$$cz = d3$$

=====

```

Masukkan nilai a1: 1
Masukkan nilai a2: 2
Masukkan nilai a3: 3
Masukkan nilai b1: 2
Masukkan nilai b2: 5
Masukkan nilai b3: 8
Masukkan nilai c1: -3
Masukkan nilai c2: -8
Masukkan nilai c3: -14
Masukkan nilai d1: 1
Masukkan nilai d2: 4
Masukkan nilai d3: 8
Nilai x = -2.000000
Nilai y = 0.000000
Nilai z = -1.000000

```


Kode Program Matlab Penyelesaian SPLTV Metode Eliminasi Gauss V2:

```

%Perhitungan SPLTV Metode Eliminasi Gauss Dengan Teknik Looping
clc

disp('Perhitungan Persamaan Linear XYZ Metode Eliminasi Gauss.')
disp('Bentuk persamaan: ')
disp('a1*x + b1*y + c1*z = d1')
disp('b2*y + c2*z = d2')
disp('c3*z = d3')
disp('=====')
disp(' ')

A = input('Masukkan nilai matrix ordo 3x3, A = ');
O = input('Masukkan nilai vektor matriks 3x1, O = ');

%Tahap iterasi
for o = 1:1:3 %Iterasi ordo (3 kolom)
    for m = o:1:3 %Iterasi baris (3 baris)
        x = A(m,o);

        for n = o:1:3 %Iterasi kolom (3 kolom)
            if m == o && n < 3 %a(m,n) = 1
                A(m,n) = A(m,n) / x;
            elseif m == o && n == 3
                A(m,n) = A(m,n) / x;
                O(m,1) = O(m,1) / x ;
            end

            if m ~= o && n < 3 %a(m,n) = 0
                A(m,n) = A(m,n) - (x*A(o,n));
            elseif m ~= o && n == 3 %a(m,n) = 0
                A(m,n) = A(m,n) - (x*A(o,n));
                O(m,1) = O(m,1) - (x*O(o,1));
            end
        end
    end
end

A
O

%Tahap substitusi
z = O(3,1)
y = O(2,1) - (A(2,3)*z)
x = O(1,1) - (A(1,2)*y) - (A(1,3)*z)

```

Tampilan Output Program Matlab Penyelesaian SPLTV Metode Eliminasi Gauss V2:

```

Perhitungan Persamaan Linear XYZ Metode Eliminasi Gauss.
Bentuk persamaan:
a1*x + b1*y + c1*z = d1
b2*y + c2*z = d2
c3*z = d3
=====

Masukkan nilai matrix ordo 3x3, A = [1 2 -3; 2 5 -8; 3 8 -14]
Masukkan nilai vektor matriks 3x1, O = [1; 4; 8]

```

$$A = \begin{pmatrix} 1 & 2 & -3 \\ 0 & 1 & -2 \\ 0 & 0 & 1 \end{pmatrix}$$

$$O = \begin{pmatrix} 1 \\ 2 \\ -1 \end{pmatrix}$$

$$z = -1$$

$$y = 0$$

$$x = -2$$

2. Metode Eliminasi Gauss Jordan

Tentukan nilai x, y dan z pada persamaan linear tiga variabel berikut:

$$x + 2y - 3z = 1$$

$$2x + 5y - 8z = 4$$

$$3x + 8y - 14z = 8$$

Masukkan nilai konstanta pada persamaan tersebut kedalam matriks :

$$\left(\begin{array}{ccc|c} 1 & 2 & -3 & 1 \\ 2 & 5 & -8 & 4 \\ 3 & 8 & -14 & 8 \end{array} \right)$$

Langkah 1 a11 Wajib bernilai 1, pada soal a11 sudah bernilai 1 maka

lanjut ke langkah berikutnya $\left(\begin{array}{ccc|c} 1 & 2 & -3 & 1 \\ 2 & 5 & -8 & 4 \\ 3 & 8 & -14 & 8 \end{array} \right)$

Langkah 2 a21, Wajib bernilai 0 maka :11

$$B_2 - 2B_1 = \left(\begin{array}{ccc|c} 1 & 2 & -3 & 1 \\ 2 & 5 & -8 & 4 \\ 3 & 8 & -14 & 8 \end{array} \right) = \left(\begin{array}{ccc|c} 1 & 2 & -3 & 1 \\ 0 & 1 & -2 & 2 \\ 3 & 8 & -14 & 8 \end{array} \right)$$

Langkah 3 a31, Wajib bernilai 0 maka :

$$B_3 - 3B_1 = \left(\begin{array}{ccc|c} 1 & 2 & -3 & 1 \\ 0 & 1 & -2 & 2 \\ 3 & 8 & -14 & 8 \end{array} \right) = \left(\begin{array}{ccc|c} 1 & 2 & -3 & 1 \\ 0 & 1 & -2 & 2 \\ 0 & 2 & -5 & 5 \end{array} \right)$$

Langkah 4 a22, Wajib bernilai 1 maka pada soal a22 sudah bernilai 1 maka

lanjut ke langkah berikutnya $\left(\begin{array}{ccc|c} 1 & 2 & -3 & 1 \\ 0 & 1 & -2 & 2 \\ 0 & 2 & -5 & 5 \end{array} \right)$

Langkah 5 a12, Wajib bernilai 0 maka :

$$B_1 - 2B_2 = \left(\begin{array}{ccc|c} 1 & 2 & -3 & 1 \\ 0 & 1 & -2 & 2 \\ 0 & 2 & -5 & 5 \end{array} \right) = \left(\begin{array}{ccc|c} 1 & 0 & 1 & -3 \\ 0 & 1 & -2 & 2 \\ 0 & 2 & -5 & 5 \end{array} \right)$$

Langkah 6 a32, Wajib bernilai 0 maka :

$$B_3 - 2B_2 = \left(\begin{array}{ccc|c} 1 & 0 & 1 & -3 \\ 0 & 1 & -2 & 2 \\ 0 & 2 & -5 & 5 \end{array} \right) = \left(\begin{array}{ccc|c} 1 & 0 & 1 & -3 \\ 0 & 1 & -2 & 2 \\ 0 & 0 & -1 & 1 \end{array} \right)$$

Langkah 7 a33, Wajib bernilai 1 maka :

$$-B_3 = \left(\begin{array}{ccc|c} 1 & 0 & 1 & -3 \\ 0 & 1 & -2 & 2 \\ 0 & 0 & -1 & 1 \end{array} \right) = \left(\begin{array}{ccc|c} 1 & 0 & 1 & -3 \\ 0 & 1 & -2 & 2 \\ 0 & 0 & 1 & -1 \end{array} \right)$$

Langkah 8 a13, Wajib bernilai 0 maka :

$$B_1 - B_3 = \left(\begin{array}{ccc|c} 1 & 0 & 1 & -3 \\ 0 & 1 & -2 & 2 \\ 0 & 0 & 1 & -1 \end{array} \right) = \left(\begin{array}{ccc|c} 1 & 0 & 0 & -2 \\ 0 & 1 & -2 & 2 \\ 0 & 0 & 1 & -1 \end{array} \right)$$

Langkah 9 a23, Wajib bernilai 0 maka :

$$B_2 + 2B_3 = \left(\begin{array}{ccc|c} 1 & 0 & 0 & -2 \\ 0 & 1 & -2 & 2 \\ 0 & 0 & 1 & -1 \end{array} \right) = \left(\begin{array}{ccc|c} 1 & 0 & 0 & -2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -1 \end{array} \right)$$

Jadi, Nilai x,y dan z adalah

$$x = -2 \quad y = 0 \quad z = -1$$

Kode Program Matlab Penyelesaian SPLTV Metode Eliminasi Gauss Jordan:

```

clc

disp('Perhitungan Persamaan Linear XYZ Metode Eliminasi Gauss Jordan.')
disp('Bentuk awal persamaan linear: ')
disp('a1*x + b1*y + c1*z = d1')
disp('a2*x + b2*y + c2*z = d2')
disp('a3*x + b3*y + c3*z = d3')
disp('_____')
disp('Bentuk akhir persamaan linear: ')
disp('ax + by + cz = d')
disp('by + cz = d')
disp('cz = d')
disp('_____')
disp('a1*x + b1*y + c1*z = d1')
disp('b2*y + c2*z = d2')
disp('c3*z = d3')
disp('=====')
disp(' ')

%Baris 1
a1 = input('Masukkan nilai a1: ');

```

```

b1 = input('Masukkan nilai b1: ');
c1 = input('Masukkan nilai c1: ');
d1 = input('Masukkan nilai d1: ');
%Baris 2
a2 = input('Masukkan nilai a2: ');
b2 = input('Masukkan nilai b2: ');
c2 = input('Masukkan nilai c2: ');
d2 = input('Masukkan nilai d2: ');
%Baris 3
a3 = input('Masukkan nilai a3: ');
b3 = input('Masukkan nilai b3: ');
c3 = input('Masukkan nilai c3: ');
d3 = input('Masukkan nilai d3: ');

%Ubah notasi persamaan linear menjadi notasi matriks,
%untuk mempermudah penulisan kode program agar sinkron
%dengan notasi2 kode program sebelumnya
a11 = a1;
a12 = b1;
a13 = c1;
o11 = d1;
a21 = a2;
a22 = b2;
a23 = c2;
o21 = d2;
a31 = a3;
a32 = b3;
a33 = c3;
o31 = d3;

if a11 == 0
    disp(' ')
    disp('Tukar Baris')
    disp(' ')

    disp(' ');
else
    %a11 wajib bernilai 1
    %Rumus B1 = B1/a11
    x = a11;
    a11 = a11/x;
    a12 = a12/x;
    a13 = a13/x;
    o11 = o11/x;

    %a21 wajib bernilai 0
    %Rumus B2 = B2 - x.B1 -> x = a21
    x = a21;
    a21 = a21 - (x*a11);
    a22 = a22 - (x*a12);
    a23 = a23 - (x*a13);
    o21 = o21 - (x*o11);

    %a31 wajib bernilai 0
    %Rumus B3 = B3 - x.B1 -> x = a31
    x = a31;
    a31 = a31 - (x*a11);
    a32 = a32 - (x*a12);
    a33 = a33 - (x*a13);
    o31 = o31 - (x*o11);

```

```
%a22 wajib bernilai 1
%Rumus B2 = B2/a22
x = a22;
a22 = a22/x;
a23 = a23/x;
o21 = o21/x;

%a12 wajib bernilai 0
%Rumus B1 = B1 - x.B2 -> x = a12
x = a12;
a12 = a12 - (x*a22);
a13 = a13 - (x*a23);
o11 = o11 - (x*o21);

%a32 wajib bernilai 0
%Rumus B3 = B3 - x.B2 -> x = a32
x = a32;
a32 = a32 - (x*a22);
a33 = a33 - (x*a23);
o31 = o31 - (x*o21);

%a33 wajib bernilai 1
%Rumus B3 = B3/a33
x = a33;
a33 = a33/x;
o31 = o31/x;
%ubah nilai c3 dan d3
a3 = a31;
b3 = a32;
c3 = a33;
d3 = o31;

%a13 wajib bernilai 0
%Rumus B1 = B1 - x.B3 -> x = a13
x = a13;
a13 = a13 - (x*a33);
o11 = o11 - (x*o31);
%ubah nilai baris 1, yaitu a1, b1, c1, dan d1
a1 = a11;
b1 = a12;
c1 = a13;
d1 = o11;

%a23 wajib bernilai 0
%Rumus B2 = B2 - x.B3 -> x = a23
x = a23;
a23 = a23 - (x*a33);
o21 = o21 - (x*o31);
%ubah nilai baris 2, yaitu b2, c2, dan d2
a2 = a21;
b2 = a22;
c2 = a23;
d2 = o21;

%Dapatkan nilai x, y, dan z
%x = o11;
%y = o21;
%z = o31;
x = d1;
```

```

y = d2;
z = d3;

disp(' ');
fprintf('Nilai x = %f', x);
disp(' ');
fprintf('Nilai y = %f', y);
disp(' ');
fprintf('Nilai z = %f', z);
disp(' ');
disp(' ');
end

```

Tampilan *Output* Program Matlab Penyelesaian SPLTV Metode Eliminasi Gauss Jordan:

Perhitungan Persamaan Linear XYZ Metode Eliminasi Gauss Jordan.

Bentuk awal persamaan linear:

$$a1*x + b1*y + c1*z = d1$$

$$a2*x + b2*y + c2*z = d2$$

$$a3*x + b3*y + c3*z = d3$$

Bentuk akhir persamaan linear:

$$ax + by + cz = d$$

$$by + cz = d$$

$$cz = d$$

$$a1*x + b1*y + c1*z = d1$$

$$b2*y + c2*z = d2$$

$$c3*z = d3$$

=====

```

Masukkan nilai a1: 1
Masukkan nilai b1: 2
Masukkan nilai c1: -3
Masukkan nilai d1: 1
Masukkan nilai a2: 2
Masukkan nilai b2: 5
Masukkan nilai c2: -8
Masukkan nilai d2: 4
Masukkan nilai a3: 3
Masukkan nilai b3: 8
Masukkan nilai c3: -14
Masukkan nilai d3: 8

```

Nilai x = -2.000000

Nilai y = 0.000000

Nilai z = -1.000000

3. Aturan Cramer

Tentukan nilai x,y dan z pada persamaan linear tiga variabel berikut:

$$x + 2y - 3z = 1$$

$$2x + 5y - 8z = 4$$

$$3x + 8y - 14z = 8$$

Pembahasan:

SPLTV dalam soal di atas dapat dinyatakan dalam bentuk matriks, yakni:

$$\begin{bmatrix} 1 & 2 & -3 \\ 2 & 5 & -8 \\ 3 & 8 & -14 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 1 \\ 4 \\ 8 \end{bmatrix}$$

Pertama kita cari dulu determinan dari matriks koefisien untuk memastikan apakah Aturan Cramer dapat diterapkan atau tidak. Matriks koefisien dari SPLTV di atas yaitu:

$$\begin{bmatrix} 1 & 2 & -3 \\ 2 & 5 & -8 \\ 3 & 8 & -14 \end{bmatrix}$$

Dengan menggunakan Metode sarrus, kita peroleh determinannya yaitu:

$$D = \begin{pmatrix} 1 & 2 & -3 & | & 1 & 2 \\ 2 & 5 & -8 & | & 2 & 5 \\ 3 & 8 & -14 & | & 3 & 8 \end{pmatrix}$$

$$D = [(1.5.(-14)) + (2.(-8).3) + (-3).2.8] - [(3.5.(-3)) + (8.(-8).1) + ((-14).2.2)]$$

$$D = [(-70 + (-48) + (-48))] - [(-45) + (-64) + (-56)]$$

$$D = (-166) - (-165) = -1$$

Karena $D \neq 0$ maka Aturan Cramer dapat diterapkan. Selanjutnya, kita cari determinan - determinan lainnya yakni

$$\begin{bmatrix} 1 & 2 & -3 \\ 2 & 5 & -8 \\ 3 & 8 & -14 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 1 \\ 4 \\ 8 \end{bmatrix}$$

$$D_x = \begin{pmatrix} 1 & 2 & -3 \\ 4 & 5 & -8 \\ 8 & 8 & -14 \end{pmatrix}$$

$$D_x = \begin{pmatrix} 1 & 2 & -3 & | & 1 & 2 \\ 4 & 5 & -8 & | & 4 & 5 \\ 8 & 8 & -14 & | & 8 & 8 \end{pmatrix}$$

$$D_x = [(1.5.(-14)) + (2.(-8).8) + (-3).4.8] - [(8.5.(-3)) + (8.(-8).1) + ((-14).4.2)]$$

$$D_x = [(-70 + (-128) + (-96))] - [(-120) + (-64) + (-112)]$$

$$D_x = (-294) - (-296) = 2$$

$$\begin{bmatrix} 1 & 2 & -3 \\ 2 & 5 & -8 \\ 3 & 8 & -14 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 1 \\ 4 \\ 8 \end{bmatrix}$$

$$D_y = \begin{pmatrix} 1 & 1 & -3 \\ 2 & 4 & -8 \\ 3 & 8 & -14 \end{pmatrix}$$

$$D_y = \left(\begin{array}{ccc|cc} 1 & 1 & -3 & 1 & 1 \\ 2 & 4 & -8 & 2 & 4 \\ 3 & 8 & -14 & 3 & 8 \end{array} \right)$$

$$D_y = [(1.4.(-14)) + (1.(-8).3) + (-3).2.8] - [(3.4.(-3)) + (8.(-8).1) + ((-14).2.1)]$$

$$D_y = [(-56 + (-24) + (-48))] - [(-36) + (-64) + (-28)]$$

$$D_y = (-128) - (-128) = 0$$

$$\begin{bmatrix} 1 & 2 & -3 \\ 2 & 5 & -8 \\ 3 & 8 & -14 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 1 \\ 4 \\ 8 \end{bmatrix}$$

$$D_z = \begin{pmatrix} 1 & 2 & 1 \\ 2 & 5 & 4 \\ 3 & 8 & 8 \end{pmatrix}$$

$$D_z = \left(\begin{array}{ccc|cc} 1 & 2 & 1 & 1 & 2 \\ 2 & 5 & 4 & 2 & 5 \\ 3 & 8 & 8 & 3 & 8 \end{array} \right)$$

$$D_z = [(1.5.8) + (2.4.3) + 1.2.8] - [(3.5.1) + (8.4.1) + (8.2.2)]$$

$$D_z = (40 + 24 + 16) - (15 + 32 + 32) = 80 - 79 = 1$$

$$D = -1 \quad D_x = 2 \quad D_y = 0 \quad D_z = 1$$

Jadi nilai masing – masing variabel pada persamaan tersebut:

$$x = \frac{D_x}{D} = \frac{2}{-1} = -2$$

$$y = \frac{D_y}{D} = \frac{0}{-1} = 0$$

$$z = \frac{D_z}{D} = \frac{1}{-1} = -1$$

Kode Program Matlab Penyelesaian SPLTV Metode Eliminasi Gauss Jordan:

```

clc

disp('Perhitungan Persamaan Linear XYZ Aturan Cramer.')
disp('Bentuk awal persamaan linear: ')
disp('a1*x + b1*y + c1*z = d1')
disp('a2*x + b2*y + c2*z = d2')
disp('a3*x + b3*y + c3*z = d3')
disp('_____')
disp('SPLTV dalam bentuk matriks:')
disp('|a11 a12 a13| |x|   |o11|')
disp('|a21 a22 a23| |y| = |o21|')
disp('|a31 a32 a33| |z|   |o31|')
disp(' ')
disp('_____')
disp('SPLTV dalam bentuk matriks:')
disp('|a1 b1 c1| |x|   |d1|')
disp('|a2 b2 c2| |y| = |d2|')
disp('|a3 b3 c3| |z|   |d3|')
disp('=====')
disp(' ')

%Baris 1
a1 = input('Masukkan nilai a1: ');
b1 = input('Masukkan nilai b1: ');
c1 = input('Masukkan nilai c1: ');
d1 = input('Masukkan nilai d1: ');
%Baris 2
a2 = input('Masukkan nilai a2: ');
b2 = input('Masukkan nilai b2: ');
c2 = input('Masukkan nilai c2: ');
d2 = input('Masukkan nilai d2: ');
%Baris 3
a3 = input('Masukkan nilai a3: ');
b3 = input('Masukkan nilai b3: ');
c3 = input('Masukkan nilai c3: ');
d3 = input('Masukkan nilai d3: ');

%Mengubah notasi SPLV menjadi notasi Matriks
a11 = a1;
a12 = b1;
a13 = c1;
o11 = d1;

a21 = a2;
a22 = b2;
a23 = c2;
o21 = d2;

a31 = a3;
a32 = b3;
a33 = c3;
o31 = d3;

```

```

%Menghitung determinan metode Sarrus
D = ((a11*a22*a33) + (a12*a23*a31) + (a13*a21*a32)) - ((a31*a22*a13) +
(a32*a23*a11) + (a33*a21*a12))
%D1 = (a11*a22*a33)
%D2 = (a12*a23*a31)
%D3 = (a13*a21*a32)
%D4 = -(a31*a22*a13)
%D5 = -(a32*a23*a11)
%D6 = -(a33*a21*a12)

if D == 0
    disp(' ')
    disp('Determinan bernilai 0. Aturan Cramer tidak dapat ditertapkan.
Perhitungan tidak dapat dilanjutkan.')
    disp(' ');
    disp(' ');
else
    %Menghitung nilai Dx
    Dx = ((o11*a22*a33) + (a12*a23*o31) + (a13*o21*a32)) - ((o31*a22*a13) +
(a32*a23*o11) + (a33*o21*a12))

    %Menghitung nilai Dy
    Dy = ((a11*o21*a33) + (o11*a23*a31) + (a13*a21*o31)) - ((a31*o21*a13) +
(o31*a23*a11) + (a33*a21*o11))

    %Menghitung nilai Dz
    Dz = ((a11*a22*o31) + (a12*o21*a31) + (o11*a21*a32)) - ((a31*a22*o11) +
(a32*o21*a11) + (o31*a21*a12))

    %Menghitung nilai x
    x = Dx/D;

    %Menghitung nilai y
    y = Dy/D;

    %Menghitung nilai z
    z = Dz/D;

    %Tampilkan nilai D,X,Y,Z
    %disp(' ');
    %fprintf('Nilai D = %f', D);
    disp(' ');
    fprintf('Nilai x = %f', x);
    disp(' ');
    fprintf('Nilai y = %f', y);
    disp(' ');
    fprintf('Nilai z = %f', z);
    disp(' ');
    disp(' ');
end

```

Tampilan *Output* Program Matlab Penyelesaian SPLTV Metode Eliminasi Gauss Jordan:

Perhitungan Persamaan Linear XYZ Aturan Cramer.
 Bentuk awal persamaan linear:
 $a1*x + b1*y + c1*z = d1$
 $a2*x + b2*y + c2*z = d2$
 $a3*x + b3*y + c3*z = d3$

SPLTV dalam bentuk matriks:

$$\begin{array}{ccc|c|c} a_{11} & a_{12} & a_{13} & |x| & |o_{11}| \\ a_{21} & a_{22} & a_{23} & |y| & = |o_{21}| \\ a_{31} & a_{32} & a_{33} & |z| & |o_{31}| \end{array}$$

SPLTV dalam bentuk matriks:

$$\begin{array}{ccc|c|c} a_1 & b_1 & c_1 & |x| & |d_1| \\ a_2 & b_2 & c_2 & |y| & = |d_2| \\ a_3 & b_3 & c_3 & |z| & |d_3| \end{array}$$

=====

Masukkan nilai a1: 1
 Masukkan nilai b1: 2
 Masukkan nilai c1: -3
 Masukkan nilai d1: 1
 Masukkan nilai a2: 2
 Masukkan nilai b2: 5
 Masukkan nilai c2: -8
 Masukkan nilai d2: 4
 Masukkan nilai a3: 3
 Masukkan nilai b3: 8
 Masukkan nilai c3: -14
 Masukkan nilai d3: 8

D =
 -1

Dx =
 2

Dy =
 0

Dz =
 1

Nilai x = -2.000000
 Nilai y = -0.000000
 Nilai z = -1.000000

Daftar Pustaka

Anton, Howard. (1997). Aljabar Linear Elementer, Edisi Kelima, terjemahan. Jakarta: Erlangga

Cahyono, Budi. Penggunaan Software Matrix Laboratory (MATLAB) Dalam Pembelajaran Alabar Linear. Jurnal Phenomenon. Volume 1 nomor 1, juli 2013

Widya Luh Putu, Risma Nurul Auliya, *Aplikasi Matlab Dalam Mengembangkan Pembelajaran Matematika Di SMA*. Journal Of Community Service. Volume 3 Issue 2, December 2018