

**LAPORAN PRAKTIKUM
METODE NUMERIK**

Judul: Pengenalan Matlab



**DISUSUN OLEH
ILHAM NUR ROMDONI M0520038**

**PROGRAM INFORMATIKA
FAKULTAS MIPA
UNIVERSITAS SEBELAS MARET
2021**

SCREENSHOT

1. Screenshot Praktikum

a. Variabel

```
>> %Ilham Nur Romdoni M0520038
>>
>> a = 3;
>> b = 7

b =

     7

>> c = a+b;
>> c = a+b

c =

    10

>> a+b

ans =

    10
```

b. Operasi

```
>> %Ilham Nur Romdoni M0520038
>>
>> a=5;
>> b = 10;
>> c = a*b

>> %Ilham Nur Romdoni M0520038
>>
>> a = 6;
>> b = 9;
>> c = a+b

c =

    15

>> x = 9;
>> y = 4;
>> z = x-y

z =

     5

>> %Ilham Nur Romdoni M0520038
>>
>> a=5;
>> b = 10;
>> c = a*b

c =

    50

>> x = 24;
>> y = 36;
>> x/y

ans =

    0.6667

>> x\y

ans =

    1.5000
```

c. Array Matriks

```
>> %Ilham Nur Romdoni M0520038
>>
>> a = [1 2 3]

a =

     1     2     3

>> b = [1 2 3; 4 5 6]

b =

     1     2     3
     4     5     6

>> c = [5 6; 4 3; 2 9]

c =

     5     6
     4     3
     2     9

>> d = [4; 5; 6]

d =

     4
     5
     6
```

d. Operasi Matriks

```
>> %Ilham Nur Romdoni M0520038
>>
>> a = [2 3 4 5; 1 2 3 4; 7 6 5 4]

a =

     2     3     4     5
     1     2     3     4
     7     6     5     4

>> x = [1 3];
>> b = a(x, 2)

b =

     3
     6

>> c = a(2, x)

c =

     1     3

>> d = a(x, :)

d =

     2     3     4     5
     7     6     5     4

>> e = a(:, x)

e =

     2     4
     1     3
     7     5

>> f = zeros(4)

f =

     0     0     0     0
     0     0     0     0
     0     0     0     0
     0     0     0     0
```

```
>> g = ones(3)           >> h = rand(3,4)

g =                       h =

     1     1     1         0.8147    0.9134    0.2785    0.9649
     1     1     1         0.9058    0.6324    0.5469    0.1576
     1     1     1         0.1270    0.0975    0.9575    0.9706

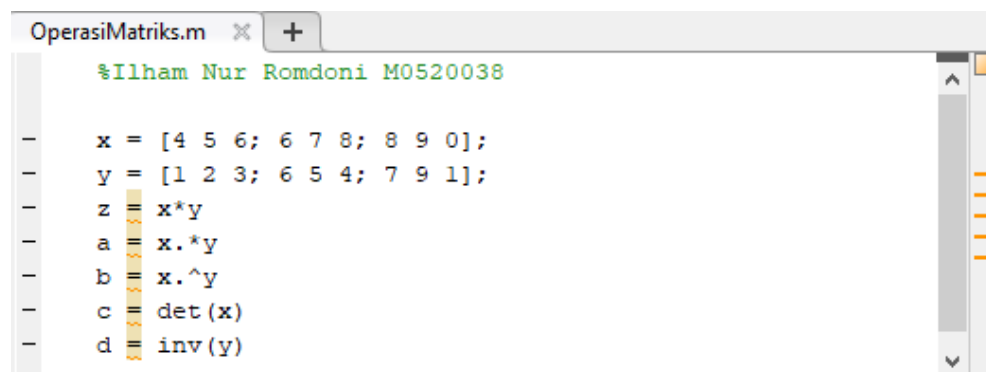
>> g(1:2, 2:3) = a(2:3, 3:4) >> h(:,3) = a(:,1)

g =                       h =

     1     3     4         0.8147    0.9134    2.0000    0.9649
     1     5     4         0.9058    0.6324    1.0000    0.1576
     1     1     1         0.1270    0.0975    7.0000    0.9706
```

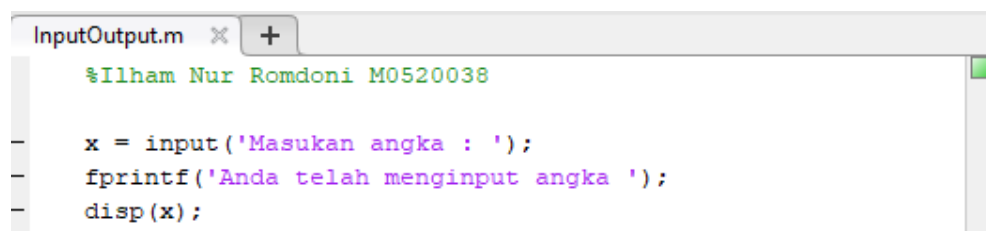
2. Screenshot Source Code

a. Operasi Matriks



```
OperasiMatriks.m  X +
%Ilham Nur Romdoni M0520038
- x = [4 5 6; 6 7 8; 8 9 0];
- y = [1 2 3; 6 5 4; 7 9 1];
- z = x*y
- a = x.*y
- b = x.^y
- c = det(x)
- d = inv(y)
```

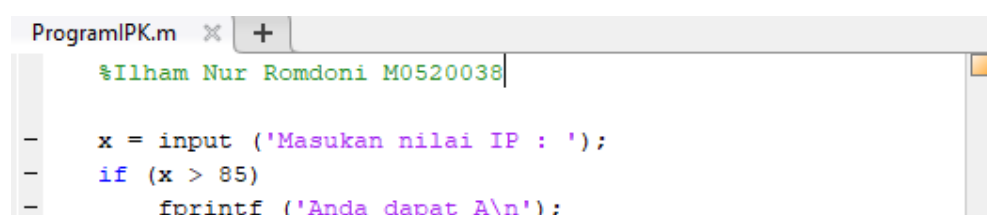
b. Input Output



```
InputOutput.m  X +
%Ilham Nur Romdoni M0520038
- x = input('Masukan angka : ');
- fprintf('Anda telah menginput angka ');
- disp(x);
```

c. Decision

1) Program IPK



```
ProgramIPK.m  X +
%Ilham Nur Romdoni M0520038
- x = input('Masukan nilai IP : ');
- if (x > 85)
-     fprintf('Anda dapat A\n');
```

```

- elseif (x >= 80 & x < 85)
-     fprintf ('Anda dapat A-\n');
- elseif (x >= 75 & x < 80)
-     fprintf ('Anda dapat B+\n');
- elseif (x >= 70 & x < 75)
-     fprintf ('Anda dapat B\n');
- elseif (x >= 65 & x < 70)
-     fprintf ('Anda dapat C+\n');
- else
-     fprintf ('Anda dapat C\n');
- end

```

2) Program Konversi Suhu

```

ProgramKonversiSuhu.m  X  +
%Ilham Nur Romdoni M0520038

- x = input('Masukan nilai suhu Celcius : ');
- if (x > 100)
-     fprintf('Maaf, batas titik didih Celcius 100 derajat Celcius\n');
- elseif (x < 0)
-     fprintf('Maaf, batas titik beku Celcius 0 derajat Celcius\n');
- else
-     y = input('Tentukan konversi suhu : ');
-     switch (y)
-     case 1
-         x = 0.8*x;
-         fprintf('Sukses konversi ke Reamur. Maka nilai R : %g\n', x);
-     case 2
-         x = 1.8*x + 32;
-         fprintf('Sukses konversi ke Fahrenheit. Maka nilai F : %g\n', x);
-     case 3
-         x = x + 273;
-         fprintf('Sukses konversi ke Kelvin. Maka nilai K : %g\n', x);
-     otherwise
-         fprintf('Pilihan Anda tidak ada dalam sistem ini');
-     end
- end
end

```

d. Looping

1) for

```

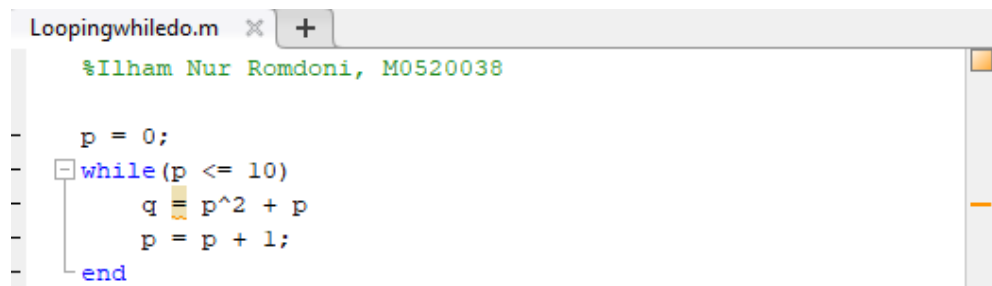
Loopingfor.m  X  +
%Ilham Nur Romdoni, M0520038

% Looping for kondisi 1
- for i = 1:5
-     p = i^2
- end

% Looping for kondisi 2
- for j = 1:0.5:5
-     q = j/2
- end

```

2) while do

A screenshot of a MATLAB script editor window titled 'Loopingwhiledo.m'. The script contains the following code:

```
%Ilham Nur Romdoni, M0520038  
  
p = 0;  
while(p <= 10)  
    q = p^2 + p  
    p = p + 1;  
end
```

ANALISIS

1. Analisis Source Code

Pada *screenshot* di atas menunjukkan *syntax* pada MATLAB.

a. Variabel

Variabel dapat dideklarasikan dengan menuliskan sebuah nama untuk variabel misal “a” lalu diikuti dengan “=” lalu *value* dari variabel tersebut. Tanda “;” digunakan untuk menyimpan variabel terlebih dahulu pada *workspace*, tidak langsung memunculkannya pada *command window*. Dengan tidak menuliskan “;” setelah *value*, maka variabel dan *value*-nya akan langsung dimunculkan setelah menekan “enter”.

b. Operasi

Syntax untuk melakukan operasi penghitungan pada MATLAB dapat dituliskan dengan “*” untuk perkalian, “+” untuk pertambahan, “-” untuk pengurangan. Untuk pembagian dapat dituliskan dengan dua cara yaitu “\” dan “/”, di mana “x\y” adalah x dibagi y sedangkan x/y adalah y dibagi x.

c. Array Matriks

Cara mendeklarasikan suatu matriks pada MATLAB yakni dengan menuliskan nama untuk matriks terlebih dahulu lalu diikuti “=” dan *value* setelahnya. *Value* untuk matriks dimulai dengan “[“ dan diakhiri dengan “]”. Tanda “;” digunakan untuk memisah baris matriks. Misal “[1 2 3; 4 5 6]” akan memunculkan matriks 2 x 3.

d. Operasi Matriks

Dideklarasikan *array* a dan *array* x. Penulisan “e = a(:, x)” pada operasi matriks berarti variabel atau *array* e adalah semua baris pada *array* a pada kolom ke x. Jumlah

baris dideklarasikan pada *value* sebelum “,” dan kolom adalah *value* setelah “,” pada “(:, x)”. Tanda “:” berarti *value* yang digunakan adalah seluruh anggota.

Tetapi pada “g(1 : 2, 2 : 3)”, tanda “:” seperti merepresentasikan kata “hingga”. Jadi “g(1:2, 2:3)” dibaca anggota *array* g pada baris 1 hingga 2 dan kolom 2 hingga 3.

Pada operasi matriks di MATLAB terdapat 3 perintah khusus yaitu zeros, ones dan rand. zeros akan memunculkan matriks dengan anggota semuanya 0. ones memunculkan matriks dengan anggota 1. Sedangkan rand akan memunculkan matriks dengan semua anggota ber-*value* acak.

Pada operasi lain tanda “.” setelah nama *array* seperti pada “a = x.*y” akan memproses operasi secara elemen. Perintah “det” akan menghitung determinan sedangkan perintah “inv” menghitung *invers*.

e. Input Output

Untuk memasukkan data dari *user* pada MATLAB, menggunakan *syntax* “input” di mana nama variabel dituliskan di depan. Seperti pada “x = input(‘Masukkan angka: ‘)”. Sedangkan untuk melakukan *output*, dapat menggunakan “fprintf” untuk mencetak tulisan. Dapat juga menggunakan “disp” yang akan memunculkan *value* dari suatu variabel yang di-*input*-kan.

f. Decision

1) Program IPK

Decision memiliki beberapa *syntax* untuk *decision*. Pada program IPK menggunakan if, elseif dan else. Masing-masing akan menjalankan perintah di dalamnya jika kasus memenuhi syarat. “if” digunakan untuk menentukan kasus awal. “elseif” digunakan untuk menentukan kasus-kasus Lainnya. Sedangkan “else” sebagai kasus default yaitu jika tidak memenuhi semua syarat pada “if” atau “elseif” maka akan menjalankan perintah di “else”. “end” untuk mengakhiri *decision*. Contoh pada *source code* program IPK “else if (x >= 70 & x < 75)” berarti perintah di bawahnya dijalankan jika dan hanya jika x lebih dari sama dengan 70 dan kurang dari 75.

2) Program Konversi Suhu

Pada program ini memasukkan *syntax* untuk *decision* yang lainnya, yaitu “switch case”. Penulisan “switch(y)” digunakan untuk menyimpan dahulu nilai

dari variabel *y* yang di-*input*-kan. “switch” juga menandakan dimulainya *syntax* untuk “case” di mana *value* dari *y* yang akan menentukan *case* mana yang akan menjalankan perintah di bawahnya. Berbeda dengan “if else” yang menggunakan “else” untuk menentukan nilai *default*, pada “switch case” menggunakan “otherwise”.

g. Looping

1) for

Pada kondisi 1, *looping* menggunakan for dituliskan “*i* = 1:5” yang berarti akan memunculkan angka 1 hingga 5. Lalu variabel “*p* = *i*²” di mana *p* akan bernilai dari *value* *i* yang nantinya akan dikuadratkan. “end” mengakhiri *syntax* for.

Kondisi kedua yaitu “*j* = 1:0.5:5” berarti for akan memunculkan bilangan 1 hingga 5 di mana setiap bilangan bertambah 0.5.

2) while do

Dimulai dengan menentukan nilai awal dari variabel yaitu *p* = 0. “while” dituliskan dengan diikuti perintah pengulangannya. Pada baris bawahnya lagi merupakan operasi yang akan dieksekusi. Dan di bawahnya lagi menentukan pertambahan yang dilakukan oleh pengulangan variabel *p*.

2. Analisis Jalannya Program

a. Variabel

Program memunculkan *value* dari variabel yang telah dideklarasikan.

b. Operasi

Program memunculkan hasil dari operasi pembagian, perkalian, pengurangan dan pertambahan.

c. Array Matriks

Program memunculkan *array* matriks dari variabel yang sudah dideklarasikan.

d. Operasi Matriks

Program memunculkan hasil dari operasi matriks. “*z*” adalah hasil perkalian antara matriks *x* dan *y*. “*a*” adalah hasil perkalian elemen dari *x* dan *y*. “*b*” adalah hasil

dari elemen x dipangkatkan y . “c” adalah hasil dari determinan x . Sedangkan “d” adalah hasil dari *invers* matriks y .

```
>> OperasiMatriks
```

```
z =
```

76	87	38
104	119	54
62	61	60

```
a =
```

4	10	18
36	35	32
56	81	0

```
b =
```

4	25	216
46656	16807	4096
2097152	387420489	0

```
c =
```

```
20
```

```
d =
```

-0.4429	0.3571	-0.1000
0.3143	-0.2857	0.2000
0.2714	0.0714	-0.1000

e. Input Output

```
>> InputOutput
```

```
Masukan angka : 3
```

```
Anda telah menginput angka      3
```

Program memunculkan tulisan “Masukkan angka” yang mana *user* diminta untuk meng-*input*-kan sebuah angka. Setelah di-*input*-kan, angka akan masuk pada variabel yang sudah dideklarasikan dan dimunculkan pada sebelah kanan tulisan “Anda telah meng-*input* angka”

f. Decision

1) Program IPK

```
>> ProgramIPK
Masukan nilai IP : 75
Anda dapat B+
```

Program memunculkan tulisan “Masukan nilai IP ; “ yang mana user diminta meng-*input* nilai IP-nya antara 0-100. Karena *user* meng-*input* nilai 75, maka perintah yang diproses adalah dari “elseif (x >=75 & x, 80)” yaitu “fprintf (‘Anda dapat B+\n’)”.

2) Program Konversi Suhu

```
>> ProgramKonversiSuhu
Masukan nilai suhu Celcius : 36.5
Tentukan konversi suhu : 1
Sukses konversi ke Reamur. Maka nilai R : 29.2
```

Program meminta *user* untuk meng-*input*-kan suhu Celsius. Setelah itu muncul “Tentukan konversi suhu:” yang mana dengan memasukkan angka 1 akan mengonversikan ke Reamur, angka 2 mengonversikan ke Fahrenheit, sedangkan angka 3 mengonversikan ke Kelvin. Dengan memilih angka 1 maka operasi pada “case 1” akan dijalankan yaitu “0.8*x”. Maka hasil akhir yang muncul adalah 29.2.

g. Looping

1) for

```
>> Loopingfor

p =          q =          q =
    1          0.5000          1.5000

p =          q =          q =
    4          0.7500          1.7500

p =          q =          q =
    9          1          2

p =          q =          q =          q =
   16          1.2500|          2.2500          2.5000
```

Pada kasus 1, maka program memunculkan *looping* untuk variabel *i* yang mana memunculkan hasil dari bilangan 1 sampai 5 yang dikuadratkan. Sedangkan pada kasus 2, program memunculkan *looping* untuk variabel *j* yang mana memunculkan hasil dari bilangan 1 sampai 5 dengan setiap naik, bilangan ditambahkan 0.5. Setiap bilangan akan dibagi 2 dan hasilnya dimasukkan pada variabel *q*.

2) while do

```
>> Loopingwhiledo
```

q =	q =	q =	
0	12	42	
q =	q =	q =	q =
2	20	56	90
q =	q =	q =	q =
6	30	72	110

Program memunculkan *looping* untuk bilangan dari 0 hingga 10. Bilangan merupakan *value* dari variabel *p*. Setiap bilangan dilakukan operasi yaitu dikuadratkan lalu ditambahkan dengan dirinya sendiri.