

BAB 1

PENGENALAN MATLAB

MATLAB (Matrix Laboratory) adalah sebuah program untuk analisis dan komputasi numerik yang merupakan bahasa pemrograman matematika lanjutan dengan dasar pemikiran menggunakan sifat dan bentuk matriks. MATLAB adalah suatu bahasa pemrograman sederhana dengan fasilitas yang jauh lebih hebat dan lebih mudah digunakan dari bahasa seperti V.BASIC, Pascal, Java, atau C. MATLAB adalah bahasa tingkat tinggi dan interaktif yang memungkinkan untuk melakukan komputasi secara intensif. MATLAB telah berkembang menjadi sebuah environment pemrograman yang canggih yang berisi fungsi-fungsi *built-in* untuk melakukan pengolahan sinyal, aljabar linear dan kalkulasi matematis lainnya. MATLAB juga berisi *toolbox* yang berisi fungsi-fungsi tambahan untuk aplikasi khusus. Penggunaan MATLAB meliputi bidang-bidang:

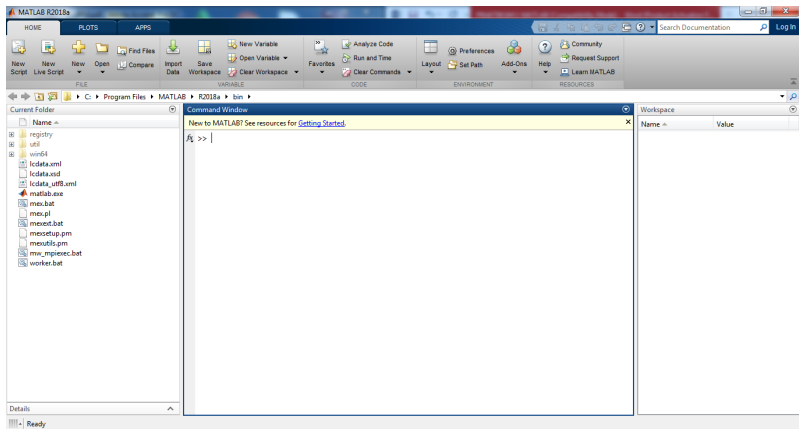
- ❑ Matematika dan Komputasi Algoritma Data
- ❑ Pemodelan, Simulasi dan Pembuatan Prototype
- ❑ Analisis Data, Explorasi, dan Visualisasi Grafik

A. MEMULAI MATLAB

Untuk dapat menjalankan program MATLAB, Anda dapat melakukan hal berikut: Double-click **ikon MATLAB**

Aplikasi Komputer Matematika Menggunakan MATLAB Untuk Tingkat Dasar

(jika ada) pada layar desktop Anda. Selanjutnya, akan muncul tampilan berikut:



Gambar 1.1 Jendela Utama MATLAB

B. WINDOW-WINDOW PADA MATLAB

Ada beberapa macam window yang tersedia dalam MATLAB, yang dapat dijelaskan sebagai berikut:

1) Jendela perintah (Command Window)

Window ini muncul pada saat pertama kali menjalankan/membuka MATLAB. (lihat gambar di atas) di window ini kita dapat melakukan akses ke command-command MATLAB, mengetikan ekspresi-ekspresi MATLAB, mengakses help window dan lain-lain. Kita dapat menuliskan perintah perhitungan sederhana, memanggil fungsi, mencari informasi tentang sebuah fungsi dengan aturan penulisannya (help), demo program, dan sebagainya. Setiap penulisan perintah selalu diawali dengan *prompt* “>>”. Misal,

mencari nilai $0 \leq x \leq 5$ dengan inkrement 0.5, maka pada command window kita dapat mengetikkan:

```
>> x=0:0.5:5
x =
Columns 1 through 9
    0    0.5000    1.0000    1.5000    2.0000
2.5000    3.0000
    3.5000    4.0000
Columns 10 through 11
    4.5000    5.0000
```

2) Jendela ruang kerja (Workspace)

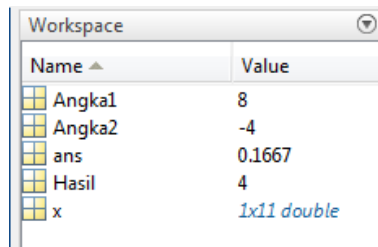
Jendela ini berisi informasi penggunaan variabel di dalam memori MATLAB. Misalkan kita akan menjumlahkan dua buah bilangan, maka pada command window kita dapat mengetikkan:

```
>> Angka1=8
Angka1 =
     8

>> Angka2=-4
Angka2 =
    -4

>> Hasil=Angka1+Angka2
Hasil =
     4
```

Maka pada workspace akan menampilkan variable yang sedang digunakan.



Gambar 1.2. Tampilan Jendela ruang kerja (Workspace)

Untuk melihat variabel yang aktif saat ini, kita dapat menggunakan perintah `who`.

```
>> who
Your variables are:
Angka1  Angka2  Hasil  ans
```

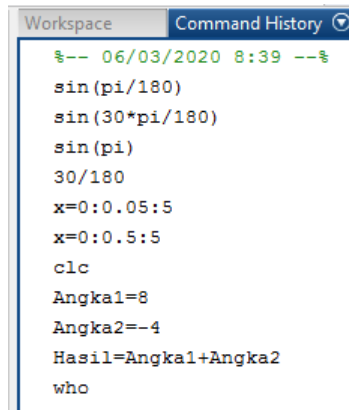
Contoh Command pada Window adalah sebagai berikut:

Command	Fungsi
what	Melihat Nama File Dalam Direktori Aktif
who	Melihat Variabel Yang Sedang Aktif
whos	Menampilkan Nama Setiap Variabel
delete	Menghapus File
clear	Menghapus Variabel
clc	Membersihkan Layar

Tabel 1.1 Command pada Window/Editor MATLAB

3) Jendela history (Command History)

Jendela ini berisi informasi tentang perintah yang pernah dituliskan sebelumnya. Kita dapat mengambil kembali perintah dengan menekan tombol panah ke atas atau mengklik perintah pada jendela histori, kemudian melakukan *copy-paste* ke command window.



```
%-- 06/03/2020 8:39 --%
sin(pi/180)
sin(30*pi/180)
sin(pi)
30/180
x=0:0.05:5
x=0:0.5:5
clc
Angka1=8
Angka2=-4
Hasil=Angka1+Angka2
who
```

Gambar 1.3. Tampilan Command History

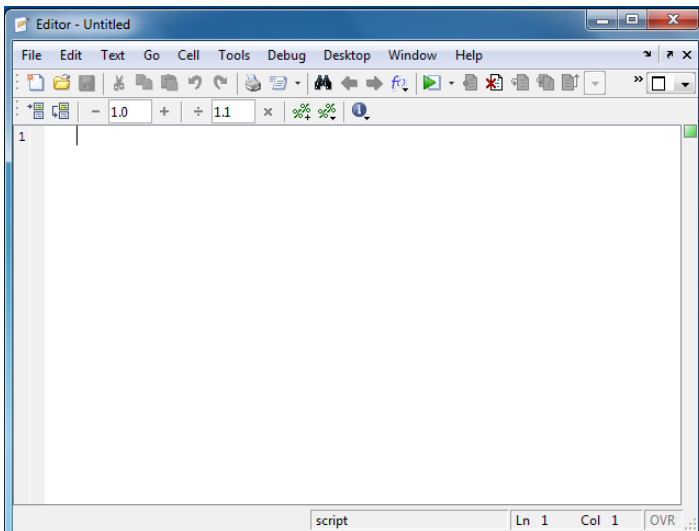
4) Help window

Window ini berisikan petunjuk dari perintah-perintah yang dimiliki MATLAB. Adapun daftar command yang ditampilkan merupakan semua command MATLAB standar dan semua *toolbox* yang diinstall dalam MATLAB. **Toolbox** adalah suatu kumpulan m-file yang telah dibuat untuk tujuan tertentu, seperti halnya sekelompok m-file yang dibuat untuk analisis statistik, numerik dan lain-lain.

5) Window m-file (editor)

Windows ini merupakan tool yang disediakan oleh MATLAB yang berfungsi sebagai editor script MATLAB (listing perintah-perintah yang harus dilakukan oleh MATLAB). Kita dapat mengetikkan barisan perintah dan menyimpannya dalam suatu file, yang dalam MATLAB dikenal dengan nama **m-file**. (file berekstensi ***.m**) Biasanya kita menggunakan m-file untuk membuat suatu program yang terdiri atas barisan perintah dan memanggil m-file tersebut di layar MATLAB command window. Ada dua cara untuk membuka editor ini, yaitu:

1. **File** kemudian pilih **New** → **M-file**
2. Ketik pada command windows: **edit**



Gambar 1.4 Tampilan m-file MATLAB

Secara formal suatu script merupakan suatu file eksternal yang berisi tulisan perintah MATLAB. Tetapi script tersebut bukan merupakan suatu fungsi. Ketika anda menjalankan suatu script, perintah di dalamnya dieksekusi seperti ketika dimasukkan langsung pada MATLAB melalui keyboard. M-file selain dipakai sebagai penamaan file juga bisa dipakai untuk menamakan fungsi, sehingga fungsi fungsi yang kita buat di jendela editor bisa di simpan dengan ekstensi .m sama dengan file yang kita panggil di jendela editor.

Saat kita menggunakan fungsi MATLAB seperti `inv`, `abs`, `cos`, `sin` dan `sqrt`, MATLAB menerima variabel berdasarkan variabel yang kita berikan. Fungsi M-file mirip dengan script file dimana keduanya merupakan file teks dengan ekstensi .m . sebagaimana script M-file, fungsi m-file tidak dimasukkan dalam jendela command window tetapi file tersendiri yang dibuat dengan editor teks.

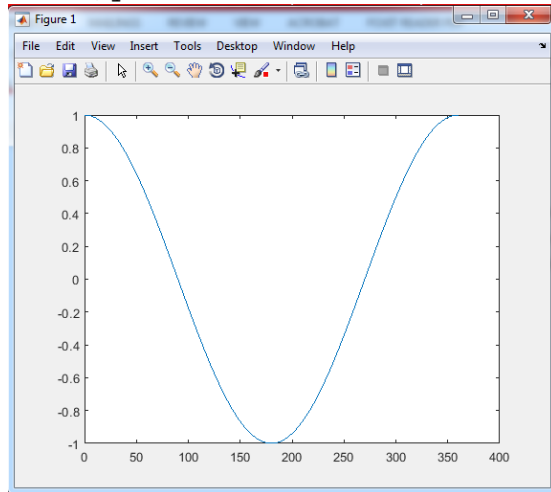
6) Figure Windows

Windows merupakan hasil visualisasi dari script MATLAB. MATLAB memberikan kemudahan bagi programmer untuk mengedit windows ini sekaligus memberikan program khusus untuk itu, sehingga selain berfungsi sebagai visualisasi output yang berupa grafik juga sekaligus menjadi media input yang interaktif.

```
>> x=0:2:360;
```

```
>> y=tan(x*pi/180);
```

```
>> plot(x,y)
```



Gambar 1.5 Tampilan Figure MATLAB

7) Format Tampilan

Mengatur tampilan nilai-nilai pada layar MATLAB. Untuk mengatur tampilan nilai pada layar MATLAB dapat digunakan perintah format. Untuk lebih jelasnya tentang pemakaian perintah format, ketik:

```
>> help format
```

```
format Set output format.
```

```
format with no inputs sets the output  
format to the default appropriate  
for the class of the variable. For  
float variables, the default is  
format SHORT.
```

```
format does not affect how MATLAB  
computations are done. Computations
```


on float variables, namely single or double, are done in appropriate floating point precision, no matter how those variables are displayed.

Computations on integer variables are done natively in integer. Integer variables are always displayed to the appropriate number of digits for the class, for example, 3 digits to display the INT8 range -128:127.

format SHORT and LONG do not affect the display of integer variables.

.....

See also disp, display, isnumeric, isfloat, isinteger.

Reference page for format

Perintah format digunakan untuk mengatur format tampilan output.

```
>> format long
```

Digunakan untuk mengatur format angka ke dalam format long (15 digit)

```
>> format short
```

Digunakan untuk mengatur format angka ke dalam format short (4 digit)

```
>> format short
```

```
>> pi
```

```
ans =
```

```
3.1416
```

```
>> format long
```

```
>> pi
```

```
ans =
```

```
3.141592653589793
```

Dengan cara yang sama, cobalah untuk format short e, long e, short g, long g, hex, bank, +, rat.

Sebagai latihan pertama, cobalah anda jalankan beberapa perintah berikut ini langsung dari prompt MATLAB (>>). Perhatikan apa yang dilakukan oleh perintah-perintah tersebut, catat dan amati.

C. MEMULAI DAN MENGAKHIRI MATLAB

Kita memulai MATLAB dengan mengeksekusi ikon MATLAB di layar komputer ataupun melalui tombol Start di Windows. Setelah proses loading program, jendela utama MATLAB akan muncul. Untuk dapat menjalankan program MATLAB, Anda dapat melakukan hal berikut:

Klik ikon MATLAB (jika ada) pada layar desktop Anda. Ketika kita selesai dengan sesi MATLAB dan ingin keluar, gunakan command **exit** atau **quit**.

```
>> exit
```

atau

```
>> quit
```

D. FIGURE WINDOW

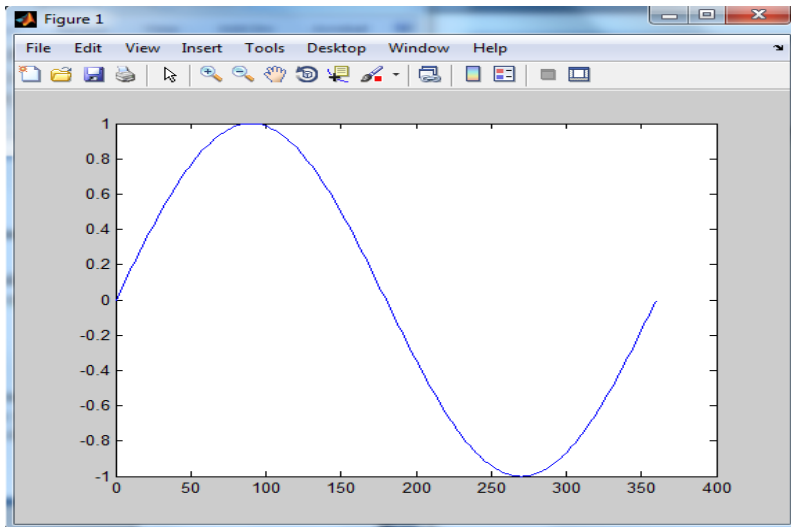
Penggunaan fasilitas grafik dari MATLAB, maka dapat dilihat berikut ini dengan mengetikkan:

Contoh 1.1:

```
>> x=0:2:360;

>> y=tan(x*pi/180);

>> plot(x,y)
```



Grafik 1.1 Grafik Sinus

E. FASILITAS HELP

a. Help dari Command Window

Dari *command window*, kita bisa gunakan: **help**, **helpwin**, dan **doc**. Misalkan kita ingin mengetahui deskripsi dari *command* berikut:

```
>> help sin %Informasi tentang sin.
sin      Sine of argument in radians.
sin(X) is the sine of the elements of X.
See also asin, sind.
Reference page for sin
Other functions named sin

>> help i % Informasi tentang i.
i        Imaginary unit.
```

As the basic imaginary unit $\text{SQRT}(-1)$, i and j are used to enter complex numbers. For example, the expressions $3+2i$, $3+2*i$, $3+2j$, $3+2*j$ and $3+2*\text{sqrt}(-1)$ all have the same value.

Since both i and j are functions, they can be overridden and used as a variable. This permits you to use i or j as an index in FOR loops, etc.

See also j .

Reference page for i

>> help log % informasi tentang log

`log` Natural logarithm.

`log(X)` is the natural logarithm of the elements of X .

Complex results are produced if X is not positive.

See also `loglp`, `log2`, `log10`, `exp`, `logm`, `reallog`.

Reference page for `log`

Other functions named `log`

Perintah **help** tanpa diikuti nama fungsi akan menampilkan daftar semua topik dalam Matlab, yakni daftar kelompok fungsi Matlab. Perintah **help elfun** akan menampilkan daftar fungsi dalam kategori “**elfun**”, yakni fungsi- fungsi elementer seperti **sin**, **exp**, dsb.

>> help sign % Informasi tentang sign

`sign` Signum function.

For each element of X , `sign(X)` returns 1 if the element

is greater than zero, 0 if it equals zero and -1 if it is

less than zero. For the nonzero elements of complex X,
 $\text{sign}(X) = X ./ \text{ABS}(X).$

See also `abs`.

Reference page for `sign`
Other functions named `sign`

>> help plot % Informasi tentang plot.

`PLOT(X,Y)` plots vector Y versus vector X. If X or Y is
`plot` Linear plot.
`plot(X,Y)` plots vector Y versus vector X. If X or Y is a matrix,
then the vector is plotted versus the rows or columns of the matrix,
whichever line up. If X is a scalar and Y is a vector, disconnected
line objects are created and plotted as discrete points vertically at
X.
.....

Example

```
x = -pi:pi/10:pi;  
y = tan(sin(x)) - sin(tan(x));  
plot(x,y,'--rs','LineWidth',2,...  
  
'MarkerEdgeColor','k',...  
  
'MarkerFaceColor','g',...  
      'MarkerSize',10)
```

See also `plottools`, `semilogx`, `semilogy`,
`loglog`, `plotyy`, `plot3`, `grid`,
`title`, `xlabel`, `ylabel`, `axis`, `axes`, `hold`,
`legend`, `subplot`, `scatter`.

Reference page for `plot`
Other functions named `plot`

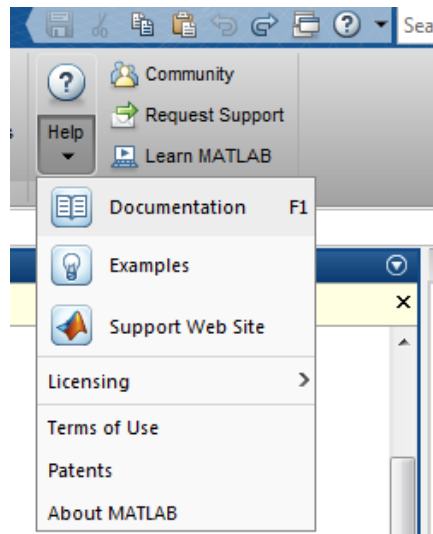
Output dari **help** juga merujuk ke command lain yang berhubungan. Dalam contoh ini: **semilogx**, **semilogy**, **loglog**, dan seterusnya. Untuk melihat deskripsinya bisa kita ketikkan `help semilogx`, `help loglog`, dan sebagainya.

Catatan:

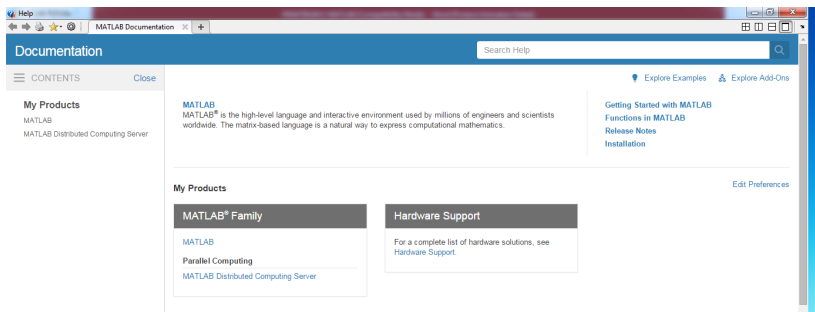
Nama fungsi atau command di dalam help ditampilkan dengan huruf kapital, tetapi ketika kita ketikkan di *command window* harus menggunakan huruf kecil. Contohnya dalam help plot di atas, tertulis PLOT(X,Y), tetapi ketika kita gunakan harus ditulis `plot(x,y)`

b. Help dari Help Browser

Sumber *help* lainnya ialah *help browser*. Anda bisa mengetikkan ***help browser*** di command window, atau dari menu Help → Documentation

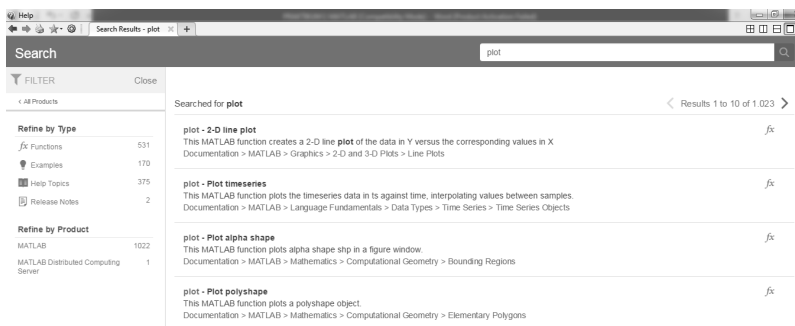


Gambar 1.6 *Help browser Documentation*



Gambar 1.7 Jendela *help browser Documentation*

Ketik **plot** pada **Search help** diperoleh sebagai berikut:



Gambar 1.8 Jendela *help plot*

F. FORMAT TAMPILAN

Setelah membaca keterangan dari petunjuk MATLAB di atas, Anda mengetahui kegunaan dan pemakaian perintah format. Untuk mempraktekkan perintah format, lakukan hal berikut

```
>> x=[2/3 pi; 2.345 1.2345e-6];
>> disp(x)
    0.6667    3.1416
    2.3450    0.0000

>> format short
>> x
x =
    0.6667    3.1416
    2.3450    0.0000

>> format short e
>> x
x =
    6.6667e-01    3.1416e+00
```



```

        2.3450e+00    1.2345e-06
>> format short
>> format compact
>> x
x =
        0.6667        3.1416
        2.3450        0.0000
>> format rat
>> x
x =
           2/3           355/113
        469/200         1/810045
>> format long
>> x
x =
        0.6666666666666667    3.141592653589793
        2.3450000000000000    0.000001234500000
>> format long e
>> disp(x)
        6.666666666666666e-01
        3.141592653589793e+00
        2.345000000000000e+00
        1.234500000000000e-06

```

Perhatikan, bahwa perintah format tidak mengubah nilai-nilai variabel, namun hanya menentukan format bagaimana nilai-nilai variabel ditampilkan. Dari contoh-contoh di atas tahukan Anda kegunaan perintah disp?.

G. KARAKTER SPESIAL MATLAB

1. **Tanda %** merupakan penanda komentar. Keterangan setelah tanda tersebut akan diabaikan dalam proses perhitungan.

Contoh 1.2 :

```
>> y = 2:2:8 % y = [2468];  
y =  
     2     4     6     8
```

2. **Tanda ;** merupakan perintah pembatas yang tidak ditampilkan di jendela kerja, merupakan pemisah kolom dan baris dalam matriks.

Contoh 1.3:

```
A = [1 3 5 ; 5 3 1];
```

3. **Tanda :** merupakan pembatas jangkauan, contohnya :

```
>> B = [0:2:8]  
B =  
     0     2     4     6     8
```

4. **Tanda ‘** merupakan transpose matriks yang merupakan suatu vector kolom

```
>> X = [3 2 4 5;7 6 5 8]  
X =  
     3     2     4     5  
     7     6     5     8
```

```
>> X=X'
X =
     3     7
     2     6
     4     5
     5     8
```

5. **Tanda ... (elipsis)** digunakan untuk menuliskan suatu perintah atau ekspresi yang tidak cukup menempati satu baris, gunakan tanda elipsis (...) diikuti menekan tombol **Enter** atau **Return** untuk melanjutkan penulisan di baris berikutnya.

contohnya:

```
>> P = sin(1)-sin(2)+sin(3)-
sin(4)+sin(5)+cos(6)+...
cos(7)-cos(8)+cos(9)-
cos(10)+cos(11)+cos(12)
P =
2.5070
```

H. FUNGSI KODE PADA MATLAB

Fungsi kode yang dapat diketik pada command windows :

1. **>> what general**

Menunjukkan instruksi-instruksi yang tersedia di direktori general, salah satunya adalah instruksi clear.

2. **>>help general**

Menunjukkan instruksi-instruksi yang tersedia di direktori general dan fungsinya secara umum.

3. **>>help clear**

Menunjukkan penjelasan detail untuk instruksi clear. (Fungsinya untuk apa, syntaxnya untuk apa, fungsi lain yang terkait apa)

4. **>> help ops**

Menunjukkan penulisan operator- operator di dalam Matlab.

5. **>>clc;**

>>P

clc digunakan untuk membersihkan layar, tetapi nilai variable yang tersimpan dimemori tidak akan hilang sehingga dapat ditampilkan kembali ke layer dengan memanggil nama variabelnya.

6. **>>clear;**

>>P

clear digunakan untuk membersihkan layer sekaligus menghapus variable dari memori sehingga kita tidak dapat menampilkan nilai variable ke layer. (muncul pesan ??? **Undefined function or variable 'P'**).

LATIHAN 1 MATLAB



1. Bagaimana cara memulai program MATLAB dan mengakhiri program MATLAB?

.....

.....

.....

.....

.....

.....

.....

2. Tuliskan output dari perintah berikut:

a. `t = 1:5`

.....

.....

b. `t = [1 2 3;4 5 6;7 8 9]`

.....

.....

.....

.....

c. `x = 3+4i`

.....

.....

3. Tuliskan output dari perintah:

```
a. x = 0:pi/4:2*pi;  
   y = sin(x)
```

.....
.....
.....
.....
.....

```
b. x = 0:pi/5:2*pi;  
   y = sin(x)
```

.....
.....
.....
.....
.....

4. Dari soal nomor 3, gambarlah grafik untuk perintah
cos (gunakan perintah plot(x,y))

a.
.....
.....
.....
.....
.....
.....
.....

.....
.....
.....
.....
.....
.....

b.
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

5. $G = 6.67 \cdot 10^{-11}$; $m_1 = 10$; $m_2 = 100$;
 $r = 50$; , tentukan nilai dari $F = \frac{Gm_1m_2}{r^2}$

.....
.....
.....
.....

6. Dari bentuk bentuk format penampilan angka,
jelaskan perbedaan dari format-format penulisan
berikut (short (default), long, shortE,

longE, shortG, longG, shortEng, longEng, +, bank, hex, rat) dan berikan masing-masing contoh.

No	Style	Keterangan	Contoh
1	short		
2	long		
3	shortE		
4	longE		
5	shortG		
6	longG		
7	shortEng		
8	longEng		
9	+		
10	bank		
11	hex		
12	rat		

BAB 2

OPERASI DASAR DAN VARIABEL

Urutan operasi yang dikerjakan dalam suatu ekspresi yang mengikuti aturan prioritas yang biasa (**hirarki**). Aturan tersebut dapat diringkas sebagai berikut:

1. Operasi di dalam kurung akan diselesaikan terlebih dahulu
2. Operasi pangkat
3. Operasi perkalian dan pembagian
4. Operasi penjumlahan dan pengurangan

>> $3 \setminus 25 * 10.5$

>> $25 * (10.5 + 16) / 8$

>> $3 * (2 - 4) / 2.5 \wedge 3 \setminus 25 * 10.5 + 16.8$

A. OPERASI DASAR

Operator aritmatika dasar yang didukung oleh **MATLAB** ialah sebagai berikut:

No	Simbol	Operasi
1	+	Penjumlahan
2	-	Pengurangan
3	*	Perkalian (aturan matriks)
4	.*	Perkalian masing-masing elemen yang bersesuaian (aturan array)
5	/	Pembagian kanan (matriks)
6	./	Pembagian kanan (array)
7	\	Pembagian kiri (matriks)

8	. \	Pembagian kiri (array)
9	^	Perpangkatan (matriks)
10	. ^	Perpangkatan (array)
11	:	Langkah

Tabel 2.1 Operator Aritmatika Dasar MATLAB

B. VARIABEL

Seperti bahasa pemrograman lainnya, MATLAB pun memiliki variabel, tetapi dalam penulisannya, variabel di dalam MATLAB tidak perlu dideklarasikan, karena MATLAB mampu mengenali tipe data dari variabel dari isi variabel tersebut. Aturan penulisan variabel pada MATLAB sama dengan aturan pada bahasa pemrograman lainnya, yaitu bersifat case sensitive, diawali dengan huruf dan selanjutnya boleh menggunakan gabungan huruf angka atau tanda garis bawah. MATLAB mampu mengenali sampai 31 karakter pertama, selanjutnya diabaikan.

Contoh 2.1:

```
>> var_1=6.8
var_1 =
    6.8000

>> var_2=[6 7 8]
var_2 =
     6     7     8
```

Semua tipe data di MATLAB memiliki bentuk yang sama, yaitu array. Array minimal berukuran 0x0 dan dapat

bertambah menjadi array $n \times m$ dimensi dengan sebarang ukuran. MATLAB mempunyai beberapa tipe data dasar (atau class), yaitu: logical, char, numeric, cell, structure, java classes, function handles. Adapun variabel khusus pada MATLAB, yaitu :

No	Variabel	Keterangan
1	ans	“answer”, digunakan untuk menyimpan hasil perhitungan terakhir
2	eps	bilangan sangat kecil mendekati nol yang merupakan batas akurasi perhitungan di MATLAB.
3	pi	konstanta π , 3.1415926...
4	inf	“infinity”, bilangan positif tak berhingga, misalkan $1/0$, 2^{5000} , dsb.
5	NaN	“not a number”, untuk menyatakan hasil perhitungan yang tak terdefinisi, misalkan $0/0$ dan inf/inf .
6	i, j	LATIHAN imajiner, $\sqrt{-1}$, untuk menyatakan bilangan kompleks.
7	flops	Jumlah operasi floating point
8	nargin	Jumlah argumen input suatu fungsi
9	nargout	Jumlah argumen output suatu fungsi
10	realmin	Bilangan real positif terkecil yang dapat digunakan
11	realmax	Bilangan real positif terbesar yang dapat digunakan

Tabel 2.2 Beberapa Variabel Khusus MATLAB

Pemberian nama variabel mengikuti rambu-rambu berikut ini:

- ❑ Gunakan karakter alfabet (A s/d Z, a s/d z), angka, dan garis bawah (_), sebagai nama variabel. Perlu diingat

bahwa MATLAB peka terhadap besar-kecilnya huruf. Misalkan: jumlah, x1, x2, S_21, H_2_in; merupakan nama variable yang valid matlab1, Matlab1, MATLAB1; dianggap sebagai 3 variabel yang berbeda.

- ❑ Jangan gunakan spasi, titik, koma, atau operator aritmatik sebagai bagian dari nama.

C. FUNGSI MATEMATIKA

Berbagi fungsi matematika yang umum kita pergunakan telah terdefinisi di MATLAB, meliputi fungsi tanda, eksponensial, logaritma, trigonometri, pembulatan, dan fungsi yang berkaitan dengan bilangan kompleks. Sebagian fungsi pada MATLAB merupakan *built-in* yaitu bagian yang tidak dapat dipisahkan dari MATLAB, sehingga fungsi-fungsi *built-in* tidak bisa diedit atau dimodifikasi sama sekali. Selain fungsi-fungsi *built-in*, terdapat pula fungsi MATLAB yang didefinisikan dalam **m-file** sehingga masih dapat diedit, misalnya fungsi *bessel*, *sinh*, dan lain-lain.

Fungsi Tanda dan Random	
<code>abs(x)</code>	Menghitung nilai absolut dari x, yaitu $ x $
<code>sign(x)</code>	Fungsi “signum”: bernilai +1 jika x positif, -1 jika x negatif, dan 0 jika x sama dengan nol.
<code>randn(x)</code>	Menghitung nilai x secara random (acak)

Fungsi Eksponensial Dan Logaritma	
<code>sqrt (x)</code>	Menghitung akar pangkat 2 dari suatu bilangan
<code>exp (x)</code>	Pangkat natural dari x, yaitu e^x ($e = 2.718282$)
<code>log (x)</code>	Logaritma natural dari x, yaitu $\ln x$
<code>log10 (x)</code>	Logaritma basis 10 dari x, yaitu $\log_{10} x$
<code>log2 (x)</code>	Logaritma basis 2 dari x, yaitu $\log_2 x$
Fungsi Trigonometri	
<code>sin (x) , cos (x) , tan (x) , cot (x) , sec (x) , csc (x)</code>	Fungsi trigonometri sinus, cosinus, tangent, cotangent, secant, dan cosecant. (x dalam satuan radian)
<code>asin (x) , acos (x) , atan (x) , acot (x) , asec (x) , acsc (x)</code>	Fungsi arcus trigonometri
<code>sinh (x) , cosh (x) , tanh (x) , coth (x) , sech (x) , csch (x)</code>	Fungsi trigonometri-hiperbolik
<code>asinh (x) , acosh (x) , atanh (x) , acoth (x) , asech (x) , acsch (x)</code>	Fungsi arcus trigonometri-hiperbolik
Fungsi Pembulatan, Sisa Pembagian dan Modulo	
<code>round (x)</code>	Pembulatan x ke bilangan bulat terdekat
<code>floor (x)</code>	Membulatkan bilangan x ke bilangan bulat terdekat menuju minus tak berhingga.
<code>ceil (x)</code>	Membulatkan bilangan x ke bilangan bulat terdekat menuju plus tak berhingga.
<code>fix (x)</code>	Membulatkan bilangan x ke bilangan bulat terdekat menuju nol

<code>rem(x, y)</code>	Menghitung sisa pembagian dari x/y
<code>mod(x, y)</code>	Menghitung sisa setelah pembagian (nilai modulus)
Fungsi Bilangan Kompleks	
<code>real(z)</code>	Menghitung komponen riil dari bilangan kompleks z
<code>imag(z)</code>	Menghitung komponen imajiner dari bilangan kompleks z
<code>abs(z)</code>	Menghitung magnitude dari bilangan kompleks z
<code>angle(z)</code>	Menghitung argumen dari bilangan kompleks z
<code>conj(z)</code>	Menghitung konjugasi dari bilangan kompleks z
Fungsi Bilangan Prima dan Nilai Faktorial	
<code>factor(x)</code>	Menghitung faktorisasi prima bilangan asli
<code>primes(x)</code>	Menghitung barisan bilangan prima
<code>factorial(x)</code>	Menghitung nilai faktorial
Fungsi KPK dan FPB	
<code>lcm(x, y)</code>	Menghitung nilai faktor pembagi terkecil dari bilangan x dan y
<code>gcd(x, y)</code>	Menghitung nilai faktor pembagi terbesar dari bilangan x dan y
Fungsi Pengukur Waktu dan Tanggal	
<code>date</code>	Mengukur/menghitung tanggal pada saat ini
<code>clock</code>	Menghasilkan 6 elemen data vektor berupa data dan waktu saat ini hingga bentuk desimal.

	Syntax [year month day hour minute seconds]
datetime	Mengkonversi waktu dan tanggal ke serangkaian angka tanggal

Tabel 2.3 Fungsi Matematika MATLAB

D. OPERASI DASAR MATLAB

Salah satu fungsi MATLAB adalah sebagai kalkulator. Untuk itu, coba jalankan perintah untuk menghitung seperti berikut (setiap akhir pemberian instruksi/perintah, selalu diakhiri dengan penekanan tombol enter). Perhatikan hasilnya, serta variabel penampungnya (variabel penyimpan hasil output). Lakukan perhitungan berikut pada *command window*.

```
>> 2+3
```

```
>> 3/2
```

Perhatikan perbedaan hasil yang diberikan oleh instruksi di atas (3/2) dengan hasil yang diberikan oleh instruksi berikut :

```
>> 3\2
```

Apa yang bisa anda jelaskan ?

Lanjutkan dengan perintah-perintah berikut :

Contoh 2.2:

```
>> 25*10-16
```

```
>> (9+18)/3^2
```

```
>> 3*(2-4)/2.5^3\ (9+18)/3^2
```

E. PENAMAAN VARIABEL

Variabel MATLAB tidak membutuhkan deklarasi ataupun statemen dimensi. Penu lisan variabel dimulai dengan hurup, dapat diikuti dengan hurup atau angka atau underscore. MATLAB hanya dapat mengenal 31 karakter pertama dari nama variabel. Selain berisi bilangan, variabel juga bisa berisi teks. Dalam mendefinisikan variabel teks gunakanlah tanda petik tunggal.

Contoh 2.3:

```
>> baca_ini = 'Contoh variabel berisi teks!';  
>> baca_ini  
baca_ini =  
Contoh variabel berisi teks!
```

Kita tidak boleh salah memperlakukan variabel berisi bilangan dengan yang berisi teks, sebab variabel teks juga bisa terlibat dalam operasi perhitungan.

Contoh 2.4:

```
>> a=7;  
>> b='7';  
>> a/b  
ans =  
0.1273  
>> a+b  
ans =  
62
```


Terlihat bahwa mengoperasikan variabel berisi teks bisa memunculkan hasil perhitungan yang “salah”.

Contoh 2.5:

```
>> Panjang = 10  
>> lebar = 5  
>> Luas_persegi_panjang = Panjang * lebar
```

MATLAB hanya dapat mengenal 31 karakter pertama dari nama variabel.

Contoh 2.6:

```
>> Jumlah_nilai=270  
enter, akan muncul  
Jumlah_nilai =  
270
```

Di sini `Jumlah_nilai` adalah nama variabel yang disimpan pada workspace dengan nilai 270. Bila terdapat variabel baru dengan nama sama, maka secara otomatis nilai variabel lama akan terhapus dan berlaku nilai variabel baru. MATLAB sangat sensitif dalam membaca variabel, ia membedakan antara huruf besar dan huruf kecil. Misalkan: `X` dan `x` adalah dua variabel yang berbeda. Pada satu baris kita dapat mendefinisikan lebih dari satu variabel dengan cara memberikan tanda koma (,) diantara dua variabel.

Contoh 2.7:

```
>> x = 23, X = 73  
x =  
23
```

X =

73

F. FUNGSI MATEMATIKA MATLAB

Perhatikan penggunaan fungsi-fungsi dari tabel 2.3 berikut, Lakukan perintah-perintah berikut:

Contoh 2.8:

```
>> a=pi/2, b=1000, c=-0.5, d=13, e=4,  
f=rand(10)  
>> sign(a)  
>> sqrt(10*b), exp(c), exp(b)  
>> log(exp(c)), log10(b), log2(b+24)  
>> sin(a), cos(a), tan(a/2)  
>> asin(c), acos(f)  
>> round(d/e), floor(d/e), ceil(d/e),  
fix(log2(b+24), rem(d,e), mod(e,f))
```

Perhatikan hasilnya, catat perbedaan-perbedaan yang anda temukan, dan analisislah!.

Selanjutnya bisa anda coba untuk perintah-perintah di bawah ini:

Contoh 2.9:

```
>> a = sin(pi)  
>> b = sin(pi)^2+cos(pi)^2  
>> c = atan(pi)  
>> d = atan(5)
```

Perhatikan hasilnya, catat perbedaan-perbedaan yang anda temukan, dan analisislah!.

Untuk bilangan kompleks yang bentuk umumnya adalah $z=a+bi$, salah satu kelebihan MATLAB adalah tidak memerlukan penanganan khusus untuk bilangan kompleks dan untuk bilangan ini diberi tanda i atau j . **Cobalah instruksi-instruksi tersebut**, kemudian perhatikan hasil yang diberikan. Perhatikan hasil outputnya dan amati kegunaan dari **fungsi-fungsi** yang ada, serta perhatikan variabel penampungnya (variabel penyimpan hasil output).

Untuk operasi Aritmatika atas Bilangan Kompleks, bisa anda perhatikan dan coba **instruksi** seperti di bawah ini :

Contoh 2.10:

```
>> A=3+4i, B = sqrt(2) - i*sqrt(2)
>> real(A), imag(A), real(B), imag(B)
>> abs(A), angle(A), abs(B), angle(B),
conj(A), conj(B),
>> abs(A)*cos(angle(A)), abs(A)*sin(angle(A))
>> angle(A)+conj(A)-real(B)+conj(B)-real(A)-
angle(B)
>> abs(B)*cos(angle(B))+abs(A)*sin(angle(A))
```

Untuk operasi **Fungsi Bilangan Prima dan Nilai Faktorial** bisa anda perhatikan dan coba instruksi seperti di bawah ini:

Contoh 2.11:

```

>> factor(500)
>> n=10000;factor(n)
>> primes(100)
>> n=50; primes(n)
>> factorial(5)
>> n=10; factorial(n)
>> mod(100,3)
>> x=10:15; mod(x,4)
>> y=1:6; mod(x,y)
>> rem(5,2)
>> rem(100,2)
>> rem(100,3)
>> rem(x,4)
>> rem(x,y)

```

Fungsi KPK dan FPB bisa anda perhatikan dan coba instruksi seperti di bawah ini:

Contoh 2.12:

```

>> lcm(3,5)
>> lcm(12,26)
>> lcm(5, 6, 8)
>> x=[4 6 7];y=[5 8 9];
>> lcm(x,y)
>> gcd(15,12)
>> gcd(104,203)
>> gcd(56, 64)
>> gcd(14, 20, 18)

```

Fungsi Pengukur Waktu dan Tanggal bisa anda perhatikan dan coba instruksi seperti di bawah ini:

Contoh 2.13:

```
>> x=[244 144 46]; y=[106 200 120];  
>> gcd(x,y)  
>> n=date  
>> n=datenum(date)  
>> m=datenum(2017,8,25)  
>> k=datenum(date,2020)  
>> clock
```

LATIHAN 2 MATLAB



1. Hitunglah dengan MATLAB, output dari perintah berikut:

a. $12/3, 5$

.....

b. $(3+5/4)^2$

.....

c. $(0,25^2+0,75^2)^{1/2}$

.....

d. $2/(6/0,3)$

.....

2. Buatlah empat variabel berikut:

$$A = 25 \quad B = 50 \quad C = 125 \quad D = 89$$

Hitunglah dengan MATLAB, output dari perintah berikut:

a. $X = A+B^5+(1/C^3)$

.....

.....

b. $Y = A/(D+B)^4$

.....

.....

c. $Z = DA^3/B+C^2$

.....
.....

3. Manakah di antara nama-nama variabel berikut yang valid ?

- a. Luas
- b. kel_1
- c. 2_data
- d. diff:3
- e. Time
- f. time_from_start
- g. 10_hasil_terakhir
- h. nilai-awal

.....
.....
.....
.....
.....
.....
.....
.....
.....

4. Misalkan: $x = \pi/6$, $y = 0,001$; Hitunglah dengan MATLAB, output dari perintah berikut:

- | | | |
|---------------|---------------|------------------|
| a. $\sqrt{y}$ | b. $\tan 3x$ | c. $\ln x$ |
| d. e^{-x} | e. $\log_2 y$ | f. $\log_{10} y$ |

$$g.\ln y$$

$$h.\cos 2x$$

$$i.\sin x$$

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

5. Misalkan: $p = 9+16i$ dan $q = -9+16i$; hitunglah keluarannya di MATLAB.

$$a. r = pq$$

$$b. s = \frac{p}{q}$$

$$c. p - r$$

$$d. R = r + s$$

$$e. p^2$$

$$f. \sqrt{q}$$

$$g. |p|$$

$$h. \angle p$$

$$i. |q|$$

$$j. \angle q$$

$$k. |r|$$

$$l. \angle r$$

$$m. |s|$$

$$n. \angle s$$

.....

.....

.....

.....

.....

.....
.....
.....
.....

6. Hitunglah dengan MATLAB, output dari perintah berikut:

- a. `n=300;factor(n)`
- b. `primes(250)`
- c. `n=70; primes(n)`
- d. `n=15; factorial(n)`
- e. `mod(150,3)`
- f. `x=110:115;y=11:16; mod(x,y)`
- g. `rem(x,y)`

.....
.....
.....
.....
.....
.....
.....
.....

7. Hitunglah dengan MATLAB, output dari perintah berikut:

- a. `x=[8 14 16];y=[6 10 12];`
- b. `lcm(x,y)`

c. `x=[24 14 48]; y=[10 20 12];`

d. `gcd(x,y)`

e. `n=date`

f. `n=datetime(date)`

g. `m=datetime(2025,6,19)`

h. `k=datetime(date,2020)`

i. `clock`

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

BAB 3

TIPE DATA

Tipe data yang digunakan dalam MATLAB adalah **string**, **skalar**, **array** dan **matriks**, sebagai berikut:

- ❑ **String** dalam MATLAB adalah type data yang terdiri atas huruf-huruf dan atau nilai-nilai ASCII yang ditampilkan representasinya. String adalah teks yang diawali dan diakhiri dengan apostrof '.....'.
- ❑ **Skalar** ialah suatu bilangan tunggal, merupakan nama lain dari data numerik. Skalar dapat dimanipulasi menggunakan fungsi.
- ❑ **Vektor** ialah sekelompok bilangan yang tersusun 1-dimensi. Dalam MATLAB biasanya disajikan sebagai vektor-baris atau vektor-kolom.
- ❑ **Array** merupakan matriks yang hanya memiliki satu baris. MATLAB menangani array secara intuitif. Untuk membuat array dalam MATLAB, yang perlu dilakukan hanyalah mengetikkan kurung kotak kiri, memasukkan elemen-elemen dengan dipisahkan oleh spasi atau koma, kemudian menutup array dengan kurung kotak kanan.
- ❑ **Matriks** ialah sekelompok bilangan yang tersusun dalam segi-empat 2-dimensi. Juga merupakan bentuk utama dari MATLAB, seperti halnya array,

matriks juga didefinisikan elemen demi elemen Di dalam MATLAB, matriks didefinisikan dengan jumlah baris dan kolomnya. Di MATLAB terdapat pula matriks berdimensi 3, 4, atau lebih.

Sebenarnya, semua data bisa dinyatakan sebagai matriks. Skalar bisa dianggap sebagai matriks satu baris – satu kolom (matriks 1×1). MATLAB juga memiliki banyak fungsi *built-in* untuk bekerja memanipulasi tipe data tersebut. Khusus untuk **Matriks** akan di jelaskan di bab selanjutnya.

Array atau disebut juga larik adalah bentuk dasar penyimpanan data pada MATLAB. Array dapat berupa array kosong dengan dimensi 0×0 , hingga array dengan n-dimensi. Dengan array, kita dapat membuat vektor, matrik, menggunakan subscript arrays, dan penggunaan khusus untuk iterasi. Sebuah matriks adalah sebuah array dua dimensi dengan elemen dapat berupa riil maupun kompleks.

A. Mengkonstruksi array

Untuk membentuk array pada MATLAB, dimulai dengan kurung siku buka, masukan nilai dipisahkan dengan koma atau spasi, dan diakhiri dengan kurung tutup. Misalnya:

$A = [2, 3 ; 4 \ 5]$, merupakan perintah untuk membuat array

dua dimensi $A = \begin{bmatrix} 2 & 3 \\ 4 & 5 \end{bmatrix}$

MATLAB menyediakan fasilitas untuk membuat array sederhana dengan cara-cara berikut:

No	Mengkonstruksi Array	Keterangan
1	<code>x=[x₁ x₂ x₃ . . . x_n]</code>	Membentuk vektor baris x dengan elemen yang disebutkan
2	<code>x=first:last</code>	Membentuk vektor baris x dimulai dengan first, dinaikan satu, berakhir sampai atau sebelum last.
3	<code>x=first:increment:last</code>	Membentuk vektor baris x dimulai dengan first, dinaikan sebesar increment, berakhir sampai atau sebelum last.
4	<code>x=linspace(first,last,n)</code>	Membentuk vektor baris x dimulai dengan first, berakhir pada last, dengan n elemen.
5	<code>x=logspace(first,last,n)</code>	Membuat vektor baris x yang memiliki spasi logaritmik, dimulai dengan 10^{first} , berakhir di 10^{last} , mempunyai n elemen

Tabel 3.1 Mengkonstruksi Array

B. Pengalamatan Elemen (Array) dan manipulasi array

Ketika sebuah array didefinisikan, MATLAB menyediakan cara yang *powerful* untuk menyisipkan, mengekstrak, dan mengatur kembali bagian-bagiannya dengan

mengidentifikasi indeks bagian tersebut. Pengetahuan mengenai hal tersebut merupakan kunci untuk memanfaatkan MATLAB secara efisien. Teknik-teknik pengalamatan array disajikan dalam tabel berikut.

No	Pengalamatan Array	Keterangan
1	$A(r, c)$	Pengalamatan elemen array di A ditentukan dengan vektor indeks baris di r dan vektor indeks kolom di c
2	$A(r, :)$	Pengalamatan subarray di A ditentukan dengan vektor indeks baris r dan semua kolom.
3	$A(:, c)$	Pengalamatan subarray di A ditentukan dengan semua baris dan vektor indeks kolom c
4	$A(:)$	Pengalamatan semua elemen A sebagai vektor kolom diambil kolom per kolom
5	$A(i)$	Pengalamatan subarray di A ditentukan oleh vektor indeks tunggal dari elemen tersebut di I, seolah A adalah vektor kolom $A(:)$
6	$A(x)$	Pengalamatan subarray di A ditentukan oleh array logika x. x harus berisi nilai 0 dan 1 dan harus berukuran sama dengan A.

Tabel 3.2 Pengalamatan Array

C. Matematika Array

Operasi matematis antara skalar dan array, meliputi penjumlahan, pengurangan, perkalian dan pembagian, beroperasi pada semua elemen array. Sedangkan operasi matematis antara array dengan array tidak sesederhana itu. Akan tetapi untuk array yang berukuran sama, operasi penjumlahan, pengurangan, perkalian, dan pembagian dengan

MATLAB, berlaku elemen per elemen. Operasi matematis pada array dirangkum dalam tabel berikut.

No	Matematika Array Elemen per Elemen	
	Data ilustratif: $a = [a_1 \ a_2 \ a_3 \ \dots \ a_n]$, $b = [b_1 \ b_2 \ b_3 \ \dots \ b_n]$, $c = \text{suatu skalar}$	
1	Penjumlahan skalar	$a+c = [a_1+c \ a_2+c \ a_3+c \ \dots \ a_n+c]$
2	Perkalian skalar	$a*c = [a_1*c \ a_2*c \ a_3*c \ \dots \ a_n*c]$
3	Penjumlahan array	$a+b = [a_1+b_1 \ a_2+b_2 \ a_3+b_3 \ \dots \ a_n+b_n]$
4	Perkalian array	$a.*b = [a_1.*b_1 \ a_2.*b_2 \ a_3.*b_3 \ \dots \ a_n.*b_n]$
5	Pembagian kanan array	$a./b = [a_1/b_1 \ a_2/b_2 \ a_3/b_3 \ \dots \ a_n/b_n]$
6	Pembagian kiri array	$a.\backslash b = [a_1.\backslash b_1 \ a_2.\backslash b_2 \ a_3.\backslash b_3 \ \dots \ a_n.\backslash b_n]$
7	Perpangkatan array	$a.^c = [a_1.^c \ a_2.^c \ a_3.^c \ \dots \ a_n.^c]$ $c.^a = [c.^{a_1} \ c.^{a_2} \ c.^{a_3} \ \dots \ c.^{a_n}]$ $a.^b = [a_1.^{b_1} \ a_2.^{b_2} \ a_3.^{b_3} \ \dots \ a_n.^{b_n}]$

Tabel 3.3 Matematika Array

Dalam operasi ini yang harus diperhatikan adalah **ukuran dimensi dari operasi elemen per elemen haruslah sama**. Menghilangkan titik sebelum tanda $*$, $/$, dan $^$ adalah sebuah kesalahan serius karena pernyataan kemudian akan menyebutkan sebuah operasi matriks skalar, bukan operasi elemen per elemen.

D. Ukuran Array

Fungsi-fungsi yang digunakan untuk mengetahui ukuran array diringkas dalam table berikut:

No	Ukuran Array	
1	Whos	Menampilkan variabel yang ada di workspace dan ukurannya
2	s=size(A)	Menghasilkan vektor dua elemen s, dengan elemen pertama adalah Banyaknya baris di a dan elemen kedua adalah banyaknya kolom Di a
3	[r,c]=size(A)	Menghasilkan dua skalar r dan c yang berturut-turut berisi Banyaknya baris dan kolom di a
4	r=size(A,1)	Menghasilkan banyaknya baris array A dalam variabel r
5	c=size(A,1)	Menghasilkan banyaknya kolom array A dalam variabel c
6	n=length(A)	Menghasilkan max(size(A)) pada variabel n

Tabel 3.4 Ukuran Array

Setelah di jelaskan tentang keempat tipe data yaitu String, Skalar, Array dan, Matriks. Untuk mengetahui lebih jauh tentang tipe-tipe data ini, cobalah anda berikan instruksi-instruksi berikut.

E. STRING

Contoh 3.1:

```
>> saya_1 = 'LATIHAN MATLAB';
>> vartest_1 = 'test aja ....';
```

bandingkan dengan contoh berikut ini :


```
>> saya_2 = "LATIHAN MATLAB";
>> vartest_2 = "test aja ...";
```

Apa yang anda temukan? Jelaskan!

```
>> size(saya_1)
ans =
1    16
>> whos
      Name              Size              Bytes   Class
Attributes
      ans              1x2                16
double
      saya_1          1x16                32   char
      saya_2          1x1                166
string
      vartest_1       1x13                26   char
      vartest_2       1x1                150
string
```

Setiap karakter dalam suatu string adalah satu elemen dalam array, dengan setiap elemennya sebesar 2 byte. Untuk melihat representasi ASCII karakter string dapat dilakukan dengan melakukan operasi aritmatika terhadap string atau mengkonversikannya menggunakan fungsi `double`.

Contoh 3.2:

```
>> double(saya_1)
ans =
Columns 1 through 16
```

```

80    114    97    107    116    105    107    117
109    32    77    65    84    76    65
    66
>> abs(saya_1)
ans =
    Columns 1 through 16
80    114    97    107    116    105    107    117
109    32    77    65    84    76    65
    66

```

Fungsi `char` menyediakan transformasi balikan :

```

>> char(saya_1)
ans =
    'LATIHAN MATLAB'

```

Karena string merupakan array numerik dengan atribut khusus, string dapat dimanipulasi dengan menggunakan semua metode manipulasi array yang tersedia dalam MATLAB. Sebagai berikut:

a. Diambil beberapa

Contoh 3.3:

```

>> saya = 'LATIHAN MATLAB';
>> t = saya(11:16)
t =
    'MATLAB'

```

string juga dialamati seperti array. Dari contoh di atas elemen ke 11 sampai ke 16 memuat kata *MATLAB*.

b. Dibalik

Jika kata tersebut dibalik maka perintahnya dapat diketikkan sebagai berikut:

```
>> t = saya(16:-1:11)
t =
'BALTAM'
```

c. Ditranspose

Jika penggunaan operator transpose, maka kata MATLAB akan dibaca dalam format kolom yaitu:

```
>> t=saya(11:16) '
t =
6x1 char array
'M'
'A'
'T'
'L'
'A'
'B'
```

d. Digabung

Penggabungan string mengikuti aturan penggabungan array:

Contoh 3.4:

```
>> a = 'Jika anda rajin belajar,'
a =
'Jika anda rajin belajar,'

>> b = 'maka anda akan dapat nilai yang
bagus'
b =
```

```
'maka anda akan dapat nilai yang bagus'
maka anda akan dapat nilai yang bagus
>> c = [a b]
c =
'Jika anda rajin belajar,maka anda akan dapat
nilai yang bagus'
```

F. SKALAR

Variabel-variabel yang telah dipelajari digunakan untuk mewakili satu nilai (satu data). Dalam hal ini variabel yang hanya mewakili satu nilai ini dalam MATLAB disebut skalar. Untuk mengetahui kerja MATLAB dengan tipe data skalar, bisa anda lakukan perintah seperti di bawah.

Perhatikan cara mendefinisikan skalar dengan ataupun tanpa kurung siku.

Contoh 3.5:

```
>> skalar1 = 3.1415
skalar1 =
3.1415
>> skalar2 = [2.71828]
skalar2 =
2.7183
```

Contoh vektor-baris dan vektor-kolom

Contoh 3.6:

```
>> vektor1=[3,5,7]
```

```
vektor1 =
3 5 7
>> vektor2=[2;4;6]
vektor2 =
2
4
6
```

Contoh 3.7:

```
>> x = 0:30:360;
>> y = [x; sin(x*pi/180)];
>> fprintf ('%6.0f %10.4f\n', y)
0      0.0000
    30      0.5000
    60      0.8660
    90      1.0000
   120      0.8660
   150      0.5000
   180      0.0000
   210     -0.5000
   240     -0.8660
   270     -1.0000
   300     -0.8660
   330     -0.5000
   360     -0.0000
```

Perhatikan nilai yang dihasilkan merupakan skalar.

G. ARRAY

Variabel yang dapat mewakili beberapa nilai sekaligus dalam MATLAB disebut **array**. Array (larik) dapat juga diartikan sebagai deretan nilai yang mempunyai tipe data sama.

a. Mengkonstruksi Array

Untuk suatu array yang mempunyai nilai-nilai yang membentuk pola yang teratur, dapat dikonstruksi dengan sintak:

$$x = \text{first}:\text{increment}:\text{last}$$

Dalam sintak tersebut **first** menyatakan nilai awal dari array dan **last** menyatakan nilai akhir dari array. Sedangkan **increment** menyatakan **bilangan (konstanta)** yang harus dijumlahkan kepada **nilai array sebelumnya** untuk memberikan **nilai array berikutnya**. Cara lain untuk mengkonstruksi array menggunakan kata `linspace` dan `logspace`.

Lakukan perintah berikut:

Contoh 3.8:

```
>> A=[-15 -10 -5 0 1] %array A mempunyai 5
anggota
>> B=[1-2i -4 3+2i] %array B mempunyai 3
anggota
>> C=[2 2*pi sqrt(2) 2-3j] %array B mempunyai
4 anggota
```

```
>> x=2:7  
>> y=3:0.1:4  
>> z=2:-5:-42
```

Perhatikan hasilnya, catat perbedaan-perbedaan yang anda temukan, dan analisislah!.

Lakukan perintah berikut untuk linspace dan logspace.

Contoh 3.9:

```
>> X=linspace(-4,3,2)  
>> X=linspace(-4,3,4)  
>> X=linspace(-4,3,6)  
>> Y=logspace(2,4,5)  
>> Y=logspace(-2,4,4)
```

Perhatikan hasilnya, catat perbedaan-perbedaan yang anda temukan, dan analisislah!.

b. Pengalamatan Array

1) Menampilkan nilai dari anggota array tertentu

Dalam matematika, untuk menyatakan nilai x yang kelima misalnya, ditulis dengan x_5 . Sedangkan dalam MATLAB, untuk menunjuk pada nilai dari anggota array tertentu ditulis dengan sintak:

Nama_array(indeks)

Dalam sintak tersebut, indeks mempunyai tipe integer positif seperti 1, 2, 3, dst. Jadi untuk menunjuk anggota kelima

dan array x maka ditulis dengan x(5), untuk menunjuk anggota ke 10 dari array y ditulis dengan y(10).

Contoh 3.10:

```
>> x = [-5 -4 -3 -2 -1 0 1 2 3 4 5 6]
x      =
      -5      -4      -3      -2      -1      0      1      2      3      4
           5      6
>> x(5)      %anggota kelima dari x
ans =
      -1
>> y=2*x+3
y      =
      -7      -5      -3      -1      1      3      5      7      9      11
           13     15
>> y(10)      %anggota kesepuluh dari y
ans =
      11
```

2) Menampilkan anggota array dalam range tertentu

MATLAB juga memberikan fasilitas untuk menampilkan anggota array dalam range atau interval tertentu dengan menggunakan tanda titik dua /colon (:). Untuk menampilkan anggota array dalam range tertentu sintaknya:

Nama_array(first:last)

Contoh 3.11:


```
>> x(2:7) %menampilkan anggota ke 2 sampai ke
7 dari x
ans =
    -4    -3    -2    -1     0     1
>> y(6:11) %menampilkan anggota ke 6 sampai
ke 11 dari y
ans =
     3     5     7     9    11    13
```

3) Memanggil anggota array dalam range tertentu

MATLAB juga menyediakan fasilitas untuk memanggil anggota array dalam range tertentu dengan indeks anggota array yang akan ditampilkan tidak harus urut naik satu persatu. Untuk keperluan sintak yang digunakan:

Nama_array(first:increment:last)

Dalam sintak tersebut, hitung merupakan ‘**penghitung**’ indeks berikutnya yang harus dituju dengan cara menambahkan indeks_sebelum + hitung = indeks_berikut. Dengan demikian untuk indeks kedua berarti indeks_awal + hitung. Dalam hal ini hitung harus berupa bilangan integer positif atau negatif.

Contoh 3.12:

```
>>x(2:1:7) %menampilkan anggota ke 2 sampai
ke 7 dari x
ans =
```

-4 -3 -2 -1 0 1

4) Pengalamatan array dalam matriks

```
>> A=[1 2 3;4 5 6;7 8 9];  
>> A(2,3) %Pengalamatan elemen array di  
matriks A pada  
baris ke-2 dan kolom ke-3  
>> A(1,:) %Pengalamatan subarray di matriks A  
pada  
baris ke-1 dan semua kolom  
>> A(:,3) %Pengalamatan subarray di matriks A  
pada  
Semua baris dan kolom ke-3  
>> A(:) %Pengalamatan semua elemen di  
matriks A sebagai  
vektor kolom diambil kolom per kolom
```

c. Matematika Array

MATLAB menyediakan operasi hitung antara skalar dengan array. Operasi hitung yang **Untuk lebih menambah pemahaman Anda lakukan perintah berikut:**

Contoh 3.13:

```
>> Y(10:-1:2)  
>> X(1:3:10)  
>> X+Y  
>> X-2*Y  
>> X/Y
```

```
>> X\Y
>> Y^2
>> X*2
>> Y.^2
>> X.*2
>> X.\2
>> Y./2
```

Perhatikan hasilnya, catat perbedaan-perbedaan yang anda temukan, dan analisislah!.

d. Ukuran array

Cobalah perintah-perintah dibawah ini pada command window!!! Lalu amati hasilnya!

Contoh 3.14:

```
>> A=[1 2 3 4;5 6 7 8]
>> s=size(A)
>> [r,c]=size(A)
>> r=size(A,1)
>> c=size(A,2)
>> length(A)
```

LATIHAN 3 MATLAB



1. Buatlah sebuah deret angka (array) yang dimulai dari angka 5 sampai dengan 100 dengan selisih dari dua angka yang berdekatan adalah 5. Tuliskan Perintah dan hasil outputnya di MATLAB.

.....
.....
.....

2. Diketahui $s = [-2 \ 0 \ 1 \ 3 \ 5]$ dan $w = [-7 \ 4 \ 3 \ -2 \ 6 \ 8]$. Hitunglah dengan MATLAB, output dari perintah berikut:

a) $s(3)$

.....

b) $s(1:3:5)$

.....

c) $w(2:5)$

.....

d) $w(6:-2:1)$

.....

3. Hitunglah dengan MATLAB, output dari perintah berikut:

a) $\text{satu} = (8:-3:-5)$

.....

.....

b) `dua=linspace(0,30,7)`

.....
.....

c) `tiga=logspace(0,2,5)`

.....
.....

d) `empat=-4:3`

.....
.....

4. Diketahui array-array `s` dan `w` sebagaimana dalam soal nomor 2. Apakah hasil dari perintah-perintah berikut ini.

a) `p = [w s]`

.....
.....
.....

b) `r = [s w(5:-1:2)]`

.....
.....
.....

c) `k = [w(3) s(1:3)]`

.....
.....
.....

d) `m = [s(2:2:5) w(6:-1:3)]`

.....
.....
.....

5. Diketahui skalar $s=-3$ dan $k=4$, serta array $h=[-3 \ 0 \ 1 \ 2]$ dan $p=[1 \ -1 \ -2 \ 3]$. Apakah hasil dari perintah berikut ini?

a) $s*h$

.....

b) $p*k$

.....

c) $p.^s$

.....

d) $k./h$

.....

6. Diketahui

$x=\text{linspace}(0,2*\pi,11)$, $y=\sin(x)$, serta $z=\cos(x)$. Carilah anggota-anggota array y dan array z , serta bentuk Grafiknya.

.....

.....

.....

.....

.....

.....

.....

.....

7. Diketahui $A = [\text{linspace}(-4, 0, 5); \text{linspace}(4, 4, 5); \text{linspace}(2, 8, 5)]$
 dan $B = [-1 \ 0 \ -1 \ 2 \ -2; (2:2:10); \text{linspace}(2, -2, 5)]$. Hitunglah :

a) $A+B$

.....

.....

.....

b) $3A-2B$

.....

.....

.....

c) $2B'$

.....

.....

.....

d) AB

.....

.....

8. Diketahui array $Z1 = [2-3i \quad 5 \quad 7i; -4+6i \quad 2+3i \quad 2-i]$ dan array $Z2 = [5-4i \quad -2-5i \quad 2i; 3-7i \quad 8+5i \quad -3-4i]$. Apakah hasil dari perintah-perintah berikut ini?

a) $Z1+Z2$

.....

b) $Z1'$

.....

c) $Z2'$

.....

d) $Z2.*Z1$

.....

e) $Z1.'./Z2'$

.....

BAB 4

MATRIKS DAN SPL

Matriks merupakan bentuk utama dari MATLAB, seperti halnya array, matriks juga didefinisikan elemen demi elemen. Matriks didefinisikan dengan kurung siku ([]) dan biasanya dituliskan baris per-baris. Tanda koma (,) digunakan untuk memisahkan kolom, dan titik-koma (;) untuk memisahkan baris. Kita juga bisa menggunakan spasi untuk memisahkan kolom dan menekan Enter ke baris baru untuk memisahkan baris.

Awalnya MATLAB digunakan untuk menyederhanakan perhitungan matriks dan aljabar linier yang muncul pada berbagai aplikasi. Skalar bisa dianggap sebagai matriks satu baris-satu kolom (matriks 1×1), dan vektor bisa dianggap sebagai matriks 1-dimensi yaitu satu baris- n kolom, atau n baris-1 kolom (matriks $1 \times n$ atau $n \times 1$). Semua perhitungan di MATLAB dilakukan dengan matriks.

A. FUNGSI MATRIKS PADA MATLAB

Untuk membantu mempercepat operasi matriks, MATLAB menyediakan fungsi-fungsi khusus yang berlaku pada matriks yang ditunjukkan pada tabel berikut ini.

No.	Fungsi	Keterangan
1.	<code>det(A)</code>	Determinan matriks A
2.	<code>inv(A)</code>	Invers dari matriks A

3.	A'	Transpose dari matriks A
4.	$\text{eig}(A)$	Memperoleh kolom vektor yang mengandung nilai eigen dari matriks bujur sangkar A
5.	$[V, D] = \text{eig}(A)$	Memperoleh nilai eigen dari matriks matriks diagonal D dan matriks V yang kolomnya berkorespondensi dengan eigen vektor kanan, sehingga $A \cdot V = V \cdot D$
6.	$\text{orth}(A)$	Orthogonalisasi matriks A
7.	$\text{pinv}(A)$	Pseudo invers dari A
8.	$\text{poly}(A)$	Polinom karakteristik dari A
9.	$\text{qr}(A)$	Dekomposisi ortogonal-segitiga
10.	$\text{rank}(A)$	Rank A, banyaknya baris (kolom) dari A yang bebas linear
11.	$\text{rref}(A)$	Bentuk eselon baris tereduksi dari A
12.	$\text{trace}(A)$	Jumlah elemen diagonal dari A

Tabel 4.1 Fungsi Matriks

B. MENYATAKAN MATRIKS

Sebagaimana dalam array baris atau array kolom, MATLAB juga menyediakan berbagai perintah (command) khusus untuk melakukan manipulasi anggota-anggota matriks.

No.	Perintah	Keterangan
1.	$A(r, s)$	Menunjuk elemen baris ke-r kolom ke-s dari matriks A
2.	$A(:, s)$	Menunjuk kolom ke s dari matriks A
3.	$A(r, :)$	Menunjuk baris ke r dari matriks A
4.	$A(:)$	Menyatakan matriks A sebagai vektor (array) kolom, dengan susunan kolom per kolom.
5.	$A(s)$	Anggota ke s dari A yang ditulis dalam bentuk array kolom.

Tabel 4.2 Perintah Manipulasi Anggota Matriks

Di samping menyediakan perintah untuk melakukan manipulasi anggota array (matriks), MATLAB juga menyediakan perintah-perintah yang berhubungan dengan dimensi atau ukuran suatu matriks. Tabel di bawah menunjukkan perintah untuk menanyakan dimensi matriks.

No.	Perintah	Keterangan
1.	<code>whos</code>	Mendisplay semua variabel dan dimensinya yang digunakan dalam workspace
2.	<code>size(A)</code>	Menyatakan dimensi dari A
3.	<code>[r c]=size(A)</code>	Mendapatkan skalar r dan c, di mana r=banyaknya baris A dan c=banyaknya kolom A
4.	<code>size(A,1)</code>	Menyatakan banyaknya baris dari A
5.	<code>size(A,2)</code>	Menyatakan banyaknya kolom dari A
6.	<code>length(A)</code>	Mendapatkan sebuah skalar yang menyatakan maksimum di antara banyaknya baris dan banyaknya kolom matriks A

Tabel 4.3 Perintah Dimensi Matriks

C. OPERASI MATRIKS PADA MATLAB

Operasi yang bisa dilakukan dalam matriks adalah operasi aljabar biasa, misalnya penjumlahan, pengurangan, perkalian dan pembagian serta beberapa operasi khusus.

No.	Perintah	Keterangan
1.	<code>A+B</code>	Menjumlahkan matriks A dengan B
2.	<code>A-B</code>	Mengurangkan matriks A dengan B
3.	<code>A*B</code>	Mengalikan matriks A dengan B

4.	A/B	Membagi matriks A dengan B
5.	$A.*B$	Mengalikan elemen-elemen matriks A dengan elemen-elemen matriks B yang lokasinya sama
6.	$A./B$	Membagi elemen-elemen matriks A dengan elemen-elemen matriks B yang lokasinya sama
7.	$A.^B$	Mempangkatkan elemen-elemen matriks A dengan elemen-elemen matriks B yang lokasinya sama

Tabel 4.4 Operasi Matriks

D. MATRIKS-MATRIKS KHUSUS

Dalam teori matriks dikenal ada beberapa matriks khusus, seperti matriks identitas, matriks nol, dan sebagainya. Perintah untuk membuat matriks-matriks khusus dalam MATLAB.

No.	Perintah	Keterangan
1.	<code>zeros(n)</code>	Membuat matriks nol berdimensi $n \times n$
2.	<code>zeros(m,n)</code>	Membuat matriks nol berdimensi $m \times n$
3.	<code>ones(n)</code>	Membuat matriks satu berdimensi $n \times n$
4.	<code>eye(n)</code>	Membuat matriks identitas berdimensi $n \times n$
5.	<code>eye(m,n)</code>	Membuat matriks identitas berdimensi $m \times n$
6.	<code>[]</code>	Matriks kosong (empty matrix)
7.	<code>diag(n)</code>	Matriks kolom yang berisi elemen-elemen diagonal dari suatu matriks tertentu.

8.	<code>triu(n)/tril(n)</code>	Matriks yang elemen-elemen diatas/ dibawah diagonal utama berisi angka nol.
----	------------------------------	---

Tabel 4.5 Operasi Perintah Membuat Matriks Khusus

E. SISTEM PERSAMAAN LINEAR (SPL)

Salah satu aplikasi matriks adalah untuk mendapatkan penyelesaian dari sistem persamaan linear. Dengan MATLAB, metode untuk mendapatkan penyelesaian sistem persamaan linear menjadi semakin mudah. Dalam subbab ini khususnya dibahas untuk sistem persamaan linear yang mempunyai penyelesaian tunggal. Pandanglah sistem persamaan linear, yang dapat ditulis dalam notasi matriks sebagai berikut :

$$AX = B$$

dengan A adalah matriks $n \times n$:

$$A = \begin{pmatrix} a_{11} & a_{12} & \text{L} & a_{1n} \\ a_{21} & a_{22} & \text{L} & a_{2n} \\ a_{31} & a_{32} & \text{L} & a_{3n} \\ \text{M} & \text{M} & \text{M} & \text{M} \\ a_{n1} & a_{n2} & \text{L} & a_{nn} \end{pmatrix}$$

dan X dan B adalah vektor-vektor n komponen:

$$X = (x_1, x_2, x_3, \dots, x_n)^t ; B = (b_1, b_2, b_3, \dots, b_n)^t$$

Apabila semua nilai $b_i = 0$ untuk $i = 1, 2, 3, \dots, n$, maka SPL diatas disebut sebagai **sistem homogen**. Jika terdapat

$b_k \neq 0$ untuk suatu $1 \leq k \leq n$, maka SPL diatas disebut sebagai **sistem tak homogen. Terdapat Tiga Macam SPL**

- 1) **Apabila Jumlah persamaan sama dengan jumlah variabel (matriks bujur sangkar).**

Contoh 4.1:

$$3x_1 + 2x_2 + x_3 = 12$$

$$2x_1 + 7x_2 + 2x_3 = 28$$

$$8x_1 + 2x_2 - 7x_3 = 4$$

- 2) **Apabila Terdapat lebih BANYAK persamaan dari pada variabel (kasus berlebihan) → disebut Penyelesaian Kuadrat Terkecil**

$$x = A \setminus b$$

Contoh 4.2:

$$x_1 + x_2 - x_3 = 1$$

$$-2x_1 - 6x_2 + 4x_3 = -2$$

$$-x_1 - 3x_2 + 3x_3 = 1$$

$$x_1 + 4x_2 + 5x_3 = 4$$

- 3) **Apabila Terdapat lebih SEDIKIT persamaan dari pada variabel (kasus kekurangan) → disebut Penyelesaian Normal Minimum**

Contoh 4.3:

$$x_1 + x_2 - x_3 = 1$$

$$-2x_1 - 6x_2 + 4x_3 = -2$$

Karena matriks merupakan dasar dari pemrograman MATLAB, maka penyusunan suatu matriks dalam MATLAB menjadi sangat penting. Seperti halnya array, matriks juga didefinisikan elemen demi elemen, untuk mengetahui lebih jauh tentang matriks dan Sistem Persamaan Linier (SPL), cobalah **Anda** berikan instruksi-instruksi berikut.

F. MENYUSUN MATRIKS

Jalankan perintah-perintah berikut, perhatikan hasilnya dan perbedaan-perbedaan yang diberikan antara perintah yang satu dengan perintah yang lainnya.

Contoh 4.4:

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$$

a. Menyusun Matriks

Ada **5 cara** untuk menyusun matriks yakni :

Cara 1:

```
>>a=[1 2 3;4 5 6;7 8 9]
```

Cara 2:

```
>>b=[1,2,3;4,5,6;7,8,9]
```

Cara 3 :

```
>>c=[1 2 3 (tekan enter)
5 5 6 (tekan enter)
7 8 9];
```

Cara 4:

Aplikasi Komputer Matematika Menggunakan MATLAB Untuk Tingkat Dasar

```
>>d1=[1 2 3];
>>d2=[4 5 6];
>>d3=[7 8 9];
>>d=[a1;a2;a3;]
```

Cara 5 :

```
>>e=input('Masukkan matrik= ');
>>Masukkan matrik=[1 2 3;4 5 6;7 8 9]
>>disp(e)
```

Sekarang, lanjutkan perintah berikut :

```
>>      A(1,1)=1;      A(1,2)=2;      A(2,2)=4;
A(2,3)=7;
>> A
```

Apa hasilnya ? Mengapa demikian?. Jelaskan

- b. Mencari Determinan, Invers, dan Transpose Matriks, Lakukan perintah berikut:**

Contoh 4.5:

```
>> B = [3 -3 1 ; -3 5 -2 ; 1 -2 1]
>> k = det(B)
>> l = inv(B)
>> m = B'
```

- c. Mencari Nilai Eigen dan Vektor Eigen Matriks, Lakukan perintah berikut:**

Contoh 4.6:

```
>> n = eig(B)
>> [V,D] = eig(B)
```


G. MENYATAKAN MATRIKS

a. Mengetahui Ukuran Matriks

Jalankan dan perhatikan perintah berikut, amati dan catat hasilnya:

Contoh 4.7:

```
>> A = zeros(2,3)
>> [baris kolom] = size(A)
>> barismatriks = size(A,1)
>> kolommatriks = size(A,2)
>> size(A,2)
```

b. Mengambil Sebagian Elemen Dari Matriks

Cobalah contoh berikut, perhatikan hasilnya, catat dan analisislah !

```
>> a = A(:,2)
>> b = A(1,:)
```

c. Membalik Urutan Kolom/Baris Dari Matriks

```
>> n = size(A,2);
>> a = A(:,n:-1:1);
```

Perhatikan hasilnya, dan analisislah !. Lanjutkan dengan perintah berikut :

```
>> j = size(A,1) ;
>> b = A(j:-1:1,:)
```

H. OPERASI MATRIKS MATLAB

Perhatikan operasi yang dilakukan terhadap dua matriks berikut, catat hasilnya dan analisislah!:

Contoh 4.8:

```
>> a = [1 2 3; 4 5 6; 7 8 9];  
>> b = [2 5 1; 3 8 2; 5 8 6];  
>> a + b  
>> a.*b  
>> a * b  
>> a./b  
>> a.^b; a'
```

Dengan mengambil contoh matriks A sebarang berordo sebarang, lakukan operasi-operasi berikut, dan catat hasilnya:

```
>> det(A)  
>> inv(A)
```

Bila diketahui matriks B sebarang yang berordo sebarang, lakukan operasi-operasi berikut, dan catat hasilnya :

```
>> B'  
>> A+B  
>> A-B  
>> A*B  
>> A/B  
>> B/A  
>> A.*B  
>> A./B  
>> A.\B
```

```
>> A.^B
```

a. Pembagian Matrik Kanan

Contoh 4.9:

```
xA= c
x = c/A
>> A = [1 2 3;2 5 4;4 3 1]
A = 1 2 3
2 5 4
4 3 1
>> c = [20 15 -8]
c = 20 15 -8
>> x = c/A
x = -8.6667 3.0952 5.6190
>>x = c*inv(A)
```

b. Pembagian Matriks Kiri

```
Ax = c
x = A\c
```

Contoh 4.10:

```
>> A = [1 2 3;2 5 4;4 3 1]
A = 1 2 3
2 5 4
4 3 1
>> c = [20;15;-8]
c = 20
```

```

15
-8
>> x = A\c
x = -1.0000
-4.7143
10.1429
>> x = inv(A)*c

```

I. MATRIKS KHUSUS MATLAB

MATLAB mempunyai sekelompok fungsi yang berbentuk matriks khusus, beberapa matriks khusus tersebut adalah :

a. Matriks Nol

Matriks yang semua elemennya berisi angka nol. Perintah MATLAB untuk matriks Nol adalah **zeros(m,n)**, dengan m adalah ukuran baris dan n ukuran kolom. Jika ukuran baris dan kolom sama, maka cukup ditulis **zeros(m)** saja. Sebagai contoh, perhatikan ilustrasi berikut ini:

Contoh 4.11:

```

>> A = zeros(3,2)
A =
     0     0
     0     0
     0     0
>> B = zeros(3)
B =
     0     0     0

```

0	0	0
0	0	0

b. Matriks Satu

Matriks yang semua elemennya berisi angka satu. Perintah MATLAB untuk matriks Satu adalah **ones(m,n)**, dengan m adalah ukuran baris dan n ukuran kolom. Jika ukuran baris dan kolom sama, maka cukup ditulis **ones(m)** saja. Sebagai contoh, perhatikan ilustrasi berikut ini:

Contoh 4.11:

```
>> A = ones(3,2)
```

A =

1	1
1	1
1	1

```
>> B = ones(3)
```

B =

1	1	1
1	1	1
1	1	1

c. Matriks Identitas

Matriks yang diagonal utamanya berisi angka satu dan angka nol pada tempat yang lainnya. Perintah MATLAB untuk matriks Identitas adalah **eye(m,n)**, dengan m adalah ukuran baris dan n ukuran kolom. Jika ukuran baris dan kolom sama, maka cukup ditulis **eye(m)** saja. Sebagai contoh, perhatikan ilustrasi berikut ini:

Contoh 4.12:

```
>> A = eye(3,2)
```

```
A =
```

```
1      0
0      1
0      0
```

```
>> B = eye(3)
```

```
B =
```

```
1      0      0
0      1      0
0      0      1
```

d. Matriks Diagonal

Matriks kolom yang berisi elemen-elemen diagonal dari suatu matriks tertentu. Perintah MATLAB untuk matriks Diagonal adalah **diag(A)**, dengan A adalah matriks yang diambil elemen-elemen diagonal utamanya Sebagai contoh, perhatikan ilustrasi berikut ini:

Contoh 4.13:

```
>> diag(A)
```

```
ans =
```

```
1
1
```

```
>> diag(B)
```

```
ans =
```

```
1
1
```

e. Matriks Segitiga Atas/Bawah

Matriks yang elemen-elemen diatas/ dibawah diagonal utama berisi angka nol. Perintah MATLAB untuk matriks Segitiga Atas/Bawah adalah **triu(A)/tril(A)**, dengan A adalah matriks awal. Sebagai contoh, perhatikan ilustrasi berikut ini:

Contoh 4.14:

```
>> A = [1 2 3 ; 4 5 6 ; 7 8 10]
```

A =

1	2	3
	4	5
		6
7	8	10

```
>> triu(A)
```

ans =

1	2	3
0	5	6
0	0	10

```
>> tril(a)
```

ans =

	1	0	0
	4	5	0
	7	8	10

J. SISTEM PERSAMAAN LINIER (SPL) MATLAB

Jika diketahui sebuah SPL berikut:

Contoh 4.15:

$$x_1 + x_2 - x_3 = 1$$

$$-2x_1 - 6x_2 + 4x_3 = -2$$

$$-x_1 - 3x_2 + 3x_3 = 1$$

SPL di atas dapat dinyatakan dalam bentuk persamaan matriks $Ax = b$ dengan

$$A = \begin{bmatrix} 1 & 1 & -1 \\ -2 & -6 & 4 \\ -1 & -3 & 3 \end{bmatrix}; \quad b = \begin{bmatrix} 1 \\ -2 \\ 1 \end{bmatrix} \text{ dan vektor } x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

✚ Matriks memiliki solusi bila *determinan*-nya tidak sama nol: $\det(A) \neq 0$

$$Ax = b$$

$$x = A \backslash b$$

$$>> A = [1 \ 1 \ -1; -2 \ -6 \ 4; -1 \ -3 \ 3]$$

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

$$-2 \ -6 \ 4$$

$$-1 \ -3 \ 3$$

$$>> c = [1; -2; 1]$$

$$c = \begin{bmatrix} 1 \\ -2 \\ 1 \end{bmatrix}$$

$$-2$$

$$1$$

$$>> x = A \backslash b$$

$$>> y = \text{inv}(A) * b$$

a. Terdapat Tiga Macam SPL

1) Apabila Jumlah persamaan sama dengan jumlah variabel (matriks bujur sangkar).

Ada dua cara penyelesaian:

$$x = \text{inv}(A) * b$$

$$x = A \setminus b \text{ (pembagian kiri matriks)}$$

Contoh 4.16:

$$3x_1 + 2x_2 + x_3 = 12$$

$$2x_1 + 7x_2 + 2x_3 = 28$$

$$8x_1 + 2x_2 - 7x_3 = 4$$

```
>> A = [3 2 1; 2 7 2; 8 2 -7]
```

```
>> b = [12; 28; 4]
```

```
>> x = A\b
```

```
x =
```

```
    2.4570
```

```
   -1.3907
```

```
    3.4106
```

```
>> y = inv(A) * b
```

```
y =
```

```
    2.4570
```

```
   -1.3907
```

```
    3.4106
```

2) Apabila Terdapat lebih BANYAK persamaan dari pada variabel (kasus berlebihan) → disebut Penyelesaian Kuadrat Terkecil

Contoh 4.17:

$$\begin{aligned}x_1 + x_2 - x_3 &= 1 \\ -2x_1 - 6x_2 + 4x_3 &= -2 \\ -x_1 - 3x_2 + 3x_3 &= 1 \\ x_1 + 4x_2 + 5x_3 &= 4\end{aligned}$$

```
>> A = [1 1 -1;-2 -6 4;-1 -3 3;1 4 5]
>> b = [1;-2;1;4]
>> x = A\b
x =
    1.5370
    0.0354
    0.4952
>> y = inv(A)*b
Error using inv
Matrix must be square.
```

3) Apabila Terdapat lebih SEDIKIT persamaan dari pada variabel (kasus kekurangan) → disebut penyelesaian Normal Minimum

 Terdapat penyelesaian yang tak terbatas

 Matlab menghitung dua di antaranya

```
>> x = A\b           % solusi dengan jumlah nol
terbanyak
```

```
>> xn = pinv(A)*b    % solusi normal minimum
```

Contoh 4.18:

$$\begin{aligned}x_1 + x_2 - x_3 &= 1 \\ -2x_1 - 6x_2 + 4x_3 &= -2\end{aligned}$$

```
>> A = [1 1 -1;-2 -6 4]
```

```
>> b = [1;-2]
```

```
>> x = A\b
```

```
x =
```

```
    1
```

```
    0
```

```
    0
```

```
>> xn = pinv(A)*b
```

```
xn =
```

```
    0.8333
```

```
   -0.1667
```

```
   -0.3333
```

LATIHAN 4 MATLAB



1. Jika diketahui matriks

$$A = \begin{pmatrix} 5 & 4 & 9 & 3 \\ 2 & 7 & 7 & 4 \\ 0 & 5 & 4 & 2 \\ 2 & 2 & 3 & 12 \end{pmatrix}$$

a) Apakah matrik di atas termasuk matriks singular?

.....
.....

b) Tentukan determinannya?

.....
.....
.....
.....

c) Tentukan tranpose dari matrik A!

.....
.....
.....
.....

d) Tentukan inversnya!

.....
.....

e) $A^{-1}A$

2. Dalam MATLAB, apakah hasil dari perintah-perintah berikut ini

Jika diketahui matriks

$B = \begin{bmatrix} 16 & 3 & 2 & 13; & 5 & 10 & 11 & 8; & 9 & 6 & 7 & 12; & 4 & 15 \\ 14 & 1 \end{bmatrix};$

a) $B(4, 2)$

b) $B(:, 2)$

c) $B(4, :)$

.....
.....
d) $B(:, 3) = [7; 7; 7; 7]$

.....
.....
.....
.....
e) $B(3, :) = []$

.....
.....
.....
.....
3. Apakah hasil dari perintah-perintah berikut ini?

a) $2 * \text{ones}(3, 4) + \text{eye}(3, 4)$

.....
.....
.....
.....
b) $\text{zeros}(4, 3) + 7 * \text{eye}(4, 3) - \text{ones}(4, 3)$

.....
.....
.....
.....
c) $3 * \text{eye}(3) + 2 * \text{ones}(3)$

.....

4. Buatlah matriks-matriks berikut dengan command ones, zeros, dan eye:

$$A = \begin{pmatrix} 8 & 0 & 0 & 0 \\ 0 & 8 & 0 & 0 \\ 0 & 0 & 8 & 0 \\ 0 & 0 & 0 & 8 \end{pmatrix} \quad B = \begin{pmatrix} 8 & 8 & 0 & 0 \\ 8 & 8 & 0 & 0 \\ -8 & 0 & 0 & 8 \\ 0 & -8 & 8 & 0 \end{pmatrix}$$

.....

5. Apakah hasil dari perintah-perintah berikut ini, Jika diketahui matriks

$P = [16 \ 5 \ 9 \ 4; 3 \ 10 \ 6 \ 15; 2 \ 11 \ -7 \ 14; 13 \ 8 \ 12 \ -1];$

$Q = [5 \ 2 \ 0 \ 2; 4 \ 7 \ 5 \ -2; 9 \ 7 \ 4 \ 3; 2 \ 4 \ 2 \ 12];$

$R = [8 \ 5 \ 4 \ -2; 2 \ -5 \ 6 \ 11; 2 \ 6 \ 7 \ 8; 3 \ 6 \ 10 \ 3];$

a) $P - 2Q^t$

.....

.....

b) PQR

.....

.....

.....

.....

c) RPQ

.....

.....

.....

.....

6. Untuk matriks P , Q , dan R dari soal nomor 5, hitunglah:

a) $\det(PQ)$

.....

.....

.....

.....

b) $\det(PQR)$

.....

.....

.....

.....

c) $\text{trace}(RPQ)$

.....

.....

.....

.....

7. Selesaikanlah SPL-SPL Berikut:

a) $-x_1 + 7x_2 + 5x_3 = 12$

$6x_1 + 3x_2 - 2x_3 = 3$

$8x_1 + \quad x_3 = 10$

$4x_1 - 4x_2 + 2x_3 = -9$

.....

.....

.....

.....

.....

.....

.....

b) $-2x_1 + x_2 + 5x_3 = 1$

$3x_2 - x_3 = 4$

$8x_1 + 2x_2 = 5$

.....

.....

.....

.....

.....

.....

.....

$$c) -x_1 + 7x_2 + 5x_3 = 12$$

$$6x_1 + 3x_2 - 2x_3 = 3$$

$$8x_1 + \quad x_3 = 10$$

$$4x_1 - 4x_2 + 2x_3 = -9$$

.....

.....

.....

.....

.....

.....

.....

BAB 5

KALKULUS DASAR

Penggunaan MATLAB pada bab ini pada dasarnya cukup banyak, namun dalam hal ini kita hanya mempelajari mengenai: Limit, Differensial, dan Integral. Materi-materi tersebut tentu tidak asing lagi buat Anda, akan tetapi bagaimana penggunaan materi-materi tersebut di dalam MATLAB. Untuk lebih lanjut pelajari materi berikut.

A. LIMIT

Secara harfiah, limit berarti batas. Dalam matematika, limit dapat dipandang sebagai batas, nilai yang mendekati (approach). Jika dikaitkan dalam fungsi, limit fungsi x di a dapat dikata nilai fungsi untuk x mendekati a , sedekat mungkin, tapi tidak pernah tepat di a .

$$\lim_{x \rightarrow a} f(x) = L$$

Berarti untuk setiap $\varepsilon > 0$ terdapat nilai $\delta > 0$ sedemikian sehingga $|f(x) - L| < \varepsilon$ dan $0 < |x - a| < \delta$.

1) Limit Kiri dan Limit Kanan

Telah kita ketahui nilai fungsi $f(x)$ di $x = a$ dapat didekati dari kiri maupun dari kanan. Nilai limit $f(x)$ untuk x menuju a dari arah kiri ($x \rightarrow a^-$ atau $x < a$) disebut limit kiri, sedangkan nilai limit $f(x)$ untuk x menuju a dari arah kanan ($x \rightarrow a^+$ atau $x > a$) disebut limit kanan. Suatu fungsi $f(x)$

dikatakan memiliki limit di $x = a$ jika dan hanya jika memiliki limit kiri dan limit kanan dan limit kirinya sama dengan limit kanannya.

Suatu fungsi $f(x)$ dikatakan memiliki limit di $x = a$ jika dan hanya jika

- $\lim_{x \rightarrow a^-} f(x)$ ada, yakni L^-
- $\lim_{x \rightarrow a^+} f(x)$ ada, yakni L^+
- $\lim_{x \rightarrow a^-} f(x) = \lim_{x \rightarrow a^+} f(x)$ atau $L^- = L^+$

2) Limit Fungsi Aljabar

Misalkan grafik dari $y = \frac{1}{x}$, maka untuk x menuju 0, y akan terus membesar ke atas (untuk daerah kanan) dan ke bawah (untuk daerah kiri). Nah, berapakah nilai y untuk $x = 0$? Jika kita melihat pola bahwa semakin kecil nilai x (dengan mengabaikan tanda positif dan negatif) maka y akan semakin besar, maka dapat disimpulkan untuk $x = 0$, $y = \infty$.

Dalam menyelesaikan persoalan limit fungsi aljabar, biasanya dapat diselesaikan dengan substitusi langsung. Misalkan $\lim_{x \rightarrow 2} x^2 + 3$, langsung saja substitusikan nilai $x = 2$ dalam $f(x) = x^2 + 3$ yang menghasilkan $f(2) = (2)^2 + 3 = 7$. Seperti yang dijelaskan pada subbab 1, untuk fungsi kontinyu, bila $x \rightarrow a$ maka pastilah $f(x) \rightarrow f(a)$.

3) Limit Fungsi Trigonometri

Pada bagian sebelumnya telah dipelajari cara menyelesaikan problem limit fungsi aljabar. Nah, sekarang

saatnya untuk mempelajari limit fungsi trigonometri. Ada tiga bentuk yang sering muncul mengenai limit fungsi trigonometri, yakni $\lim_{x \rightarrow 0} \sin x$, $\lim_{x \rightarrow 0} \cos x$, dan $\lim_{x \rightarrow 0} \tan x$.

a. Limit Fungsi Sinus

Untuk limit $\sin \theta$ untuk θ menuju nol diperoleh

$$\lim_{\theta \rightarrow 0} \sin \theta = \theta$$

Dari hasil ini dapat pula diperoleh

$$\lim_{\theta \rightarrow 0} \frac{\sin \theta}{\theta} = \lim_{\theta \rightarrow 0} \frac{\theta}{\sin \theta} = \frac{\theta}{\theta} = 1$$

b. Limit Fungsi Cosinus

Untuk limit $\cos \theta$ untuk θ menuju nol ialah

$$\lim_{\theta \rightarrow 0} \cos \theta = 1$$

Berdasarkan hasil ini dapat pula diperoleh

$$\lim_{\theta \rightarrow 0} \frac{1 - \cos \theta}{\theta} = \lim_{\theta \rightarrow 0} \left(\frac{1}{\theta} - \frac{\cos \theta}{\theta} \right) = \lim_{\theta \rightarrow 0} \left(\frac{1}{\theta} - \frac{1}{\theta} \right)$$

$$\lim_{\theta \rightarrow 0} \frac{1 - \cos \theta}{\theta} = \lim_{\theta \rightarrow 0} \frac{\cos \theta - 1}{\theta} = 0$$

c. Limit Fungsi Tangen

Untuk limit $\tan \theta$ untuk θ menuju nol berlaku:

$$\lim_{\theta \rightarrow 0} \tan \theta = \theta$$

Dari hasil ini kembali dapat pula diperoleh

$$\lim_{\theta \rightarrow 0} \frac{\tan \theta}{\theta} = \lim_{\theta \rightarrow 0} \frac{\theta}{\tan \theta} = \frac{\theta}{\theta} = 1$$

Untuk menghitung limit suatu fungsi, perintah MATLAB yang dipergunakan adalah `limit()`. Beberapa

perintah Matlab untuk limit suatu fungsi, dapat dilihat dalam Tabel berikut:

No.	Perintah Matlab	Keterangan
1.	<code>limit(f)</code>	Memberikan limit dari f dengan menggunakan $x=0$ sebagai titik limit atau $\lim_{x \rightarrow 0} f(x)$
2.	<code>limit(f,a)</code> atau <code>limit(f,x,a)</code>	Memberikan limit dari f dengan menggunakan $x=a$ sebagai titik limit atau $\lim_{x \rightarrow a} f(x)$
3.	<code>limit(f,x,'left')</code>	Memberikan limit dari f dengan menggunakan $x=a^-$ sebagai titik limit yaitu limit kiri atau $\lim_{x \rightarrow a^-} f(x)$
4.	<code>limit(f,x,'right')</code>	Memberikan limit dari f dengan menggunakan $x=a^+$ sebagai titik limit yaitu limit kanan atau $\lim_{x \rightarrow a^+} f(x)$

Tabel 5.1 Perintah Menghitung Limit Suatu Fungsi

B. DIFERENSIAL (TURUNAN)

Turunan (derivatif) dapat dianggap sebagai suatu operasi matematis yang menyatakan interpretasi nilai sebuah faktor dari faktor lain yang berdimensi lebih besar (ordenya turun satu). Karena faktor yang berdimensi kecil itu merupakan suatu ‘titik’ pada dimensi yang lebih besar, maka turunan erat hubungannya dengan limit. Namun demikian,

pemahaman di atas hanyalah pernyataan berdasarkan hal-hal praktis yang nampak pada operasi turunan. Secara formal, turunan dapat didefinisikan sebagai perubahan nilai fungsi terhadap perubahan nilai input untuk selang yang sangat kecil (menuju nol).

Misalkan suatu fungsi $y = f(x)$ yang terdefinisi pada selang terbuka I yang memuat semua bilangan riil. Bila nilai x berubah dari $x = x_1$ ke $x = x_2$, maka nilai fungsi juga akan berubah dari $y_1 = f(x_1)$ ke $y_2 = f(x_2)$. Dengan demikian perubahan nilai fungsi terhadap perubahan nilai x didefinisikan sebagai:

$$\frac{y_2 - y_1}{x_2 - x_1} = \frac{f(x_2) - f(x_1)}{x_2 - x_1}$$

Jika dipilih nilai x_2 yang mendekati x_1 , maka secara matematis didefinisikan perbandingan ini ialah

$$\lim_{x_2 \rightarrow x_1} \frac{f(x_2) - f(x_1)}{x_2 - x_1}, \text{ Asal limitnya ada.}$$

Dengan substitusi $x_2 = x_1 + \Delta x$, yang bila $x_2 \rightarrow x_1$ mengakibatkan $\Delta x \rightarrow 0$, maka turunan pertama fungsi f di titik $x = x_1$ dapat dituliskan dalam bentuk:

$$f'(x_1) = \lim_{\Delta x \rightarrow 0} \frac{\Delta y}{\Delta x} = \lim_{\Delta x \rightarrow 0} \frac{f(x_1 + \Delta x) - f(x_1)}{\Delta x}$$

Jika limitnya ada, dikatakan fungsi f mempunyai turunan di $x = x_1$. Sebaliknya jika limitnya tidak ada, dikatakan f tidak diturunkan di $x = x_1$, dimana $\frac{\Delta y}{\Delta x}$ disebut hasil bagi diferensial. Turunan pertama suatu fungsi $y = f(x)$

dapat dinotasikan dengan notasi dari Newton dan Lagrange yakni $f'(x)$, $\frac{d}{dx}f(x)$ atau notasi Leibniz yakni $\frac{dy}{dx}$.

Sedangkan nilai turunan di suatu titik tertentu (misalnya di $x = x_1$) dinotasikan dengan $f'(c)$ atau $\left. \frac{dy}{dx} \right|_{x=c}$.

Berikut diberikan rangkuman rumus-rumus sederhana untuk menghitung turunan fungsi yang mudah dihafal dan digunakan sebagaimana yang telah didapatkan sebelumnya.

1. Turunan fungsi konstan.

Jika $f(x) = c$, c suatu konstanta, maka $f'(x) = 0$; $\forall x \in \mathbb{R}$

2. Turunan fungsi identitas.

Jika $f(x) = x$, maka $f'(x) = 1$; $\forall x \in \mathbb{R}$

3. Turunan fungsi pangkat.

Jika $f(x) = x^n$, maka $f'(x) = n.x^{n-1}$; $n \in \mathbb{Z}^+$; $x \neq 0$

4. Aturan jumlah.

Jika f dan g adalah fungsi-fungsi terdiferensialkan, maka $(f + g)'(x) = f'(x) + g'(x)$

5. Aturan selisih.

Jika f dan g adalah fungsi-fungsi terdiferensialkan, maka $(f - g)'(x) = f'(x) - g'(x)$

6. Aturan hasil kali

Jika f dan g adalah fungsi-fungsi terdiferensialkan, maka $(f \cdot g)'(x) = f'(x) g(x) + f(x) g'(x)$

Atau biasa dinyatakan $u \cdot v \frac{dy}{dx} = u'v + uv'$

7. Aturan hasil bagi.

Jika f dan g adalah fungsi-fungsi terdiferensialkan dengan $g(x) \neq 0$, maka

$$\left(\frac{f}{g}\right)'(x) = \frac{f'(x) g(x) - f(x) g'(x)}{(g(x))^2}$$

Atau biasa dinyatakan $\frac{u}{v} \frac{dy}{dx} = \frac{u'v - uv'}{v^2}$

8. Rumus turunan untuk fungsi trigonometri

Jika $f(x) = \sin x$, maka $f'(x) = \cos x$

Jika $f(x) = \cos x$, maka $f'(x) = -\sin x$

Jika $f(x) = \tan x$, maka $f'(x) = \sec^2 x$

Jika $f(x) = \cot x$, maka $f'(x) = -\operatorname{cosec}^2 x$

Jika $f(x) = \sec x$, maka $f'(x) = \sec x \tan x$

Jika $f(x) = \operatorname{cosec} x$, maka $f'(x) = -\operatorname{cosec} x \cot x$

Untuk menghitung differensial suatu fungsi, perintah MATLAB yang dipergunakan adalah `diff()`. Beberapa perintah Matlab untuk differensial suatu fungsi pada Tabel berikut:

No.	Perintah Matlab	Keterangan
1.	<code>diff (f)</code>	Memberikan turunan dari f dengan memperhatikan variabel independent.

2.	<code>diff(f,'t')</code>	Memberikan turunan dari f dengan memperhatikan variabel 't'.
3.	<code>diff(f,n)</code>	Memberikan turunan ke n dari f dengan memperhatikan variabel independent.
4.	<code>diff(f,'t',n)</code>	Memberikan turunan ke n dari f dengan memperhatikan variabel t.

Tabel 5.2 Perintah Menghitung Differensial Suatu Fungsi

C. INTEGRAL (ANTI-TURUNAN)

Integral merupakan lawan (invers) dari turunan, sehingga notasinya pun dibuat sedemikian sehingga menjadi lawan dari notasi integral. Jika pada turunan diberikan notasi berbentuk pembagi, semisal d/dx , maka notasi integral berbentuk pengali, semisal dx .

$$\begin{aligned} \text{Notasi turunan fungsi x:} \quad & \frac{d}{dx} f(x) \\ & \text{atau} \quad \frac{dy}{dx} \end{aligned}$$

$$\begin{aligned} \text{Notasi integral fungsi x:} \quad & \int f(x) dx \\ & \text{atau} \quad \int y dx \end{aligned}$$

Notasi khusus pada operasi integral berupa simbol S (dari kata **sum**, **penjumlahan**) yang digayakan seperti halnya penambahan **simbol “d”** pada operasi turunan. Itu adalah notasi baku dari integral, meskipun hanya dengan menuliskan $f(x)dx$ sudah cukup bagi kita untuk mengetahui bahwa yang dimaksud ialah integral dari $f(x)$ terhadap x .

Dalam berbagai problem matematis, biasanya terdapat berbagai bentuk integral fungsi. Berikut beberapa di antaranya.

1. $\int k \, du = ku + C$
2. $\int u^n \, du = \frac{1}{n+1} u^{n+1} + C ; n \neq -1$
3. $\int \frac{1}{u} \, du = \ln|u| + C ; n \neq -1$
4. $\int e^u \, du = e^u + C$
5. $\int a^u \, du = \frac{a^u}{\ln a} + C$
6. $\int \sin u \, du = -\cos u + C$
7. $\int \cos u \, du = \sin u + C$
8. $\int \sec^2 u \, du = \tan u + C$
9. $\int \sec u \tan u \, du = \sec u + C$
10. $\int \csc^2 u \, du = -\cot u + C$
11. $\int \csc u \cot u \, du = -\csc u + C$
12. $\int \tan u \, du = \ln|\cos u| + C$
13. $\int \cot u \, du = \ln|\sin u| + C$
14. $\int \sec u \, du = \ln|\sec u + \tan u| + C$
15. $\int \csc u \, du = -\ln|\csc u + \cot u| + C$
16. $\int \frac{1}{\sqrt{a^2 - u^2}} \, du = \sin^{-1}\left(\frac{u}{a}\right) + C$
17. $\int \frac{1}{u\sqrt{u^2 - a^2}} \, du = \frac{1}{a} \sec^{-1}\left(\frac{|u|}{a}\right) + C$
18. $\int \frac{1}{a^2 + u^2} \, du = \frac{1}{a} \tan^{-1}\left(\frac{u}{a}\right) + C$

Untuk menghitung integral suatu fungsi, perintah Matlab yang dipergunakan adalah `int()`. Beberapa perintah

Matlab untuk menghitung integral suatu fungsi, dapat dilihat dalam Tabel berikut ini :

No.	Perintah Matlab	Keterangan
1.	<code>int (f)</code>	Memberikan integral dari f dengan memperhatikan variabel independent.
2.	<code>int (f, 't')</code>	Memberikan integral dari f dengan memperhatikan variabel 't'.
3.	<code>int (f, a, b)</code>	Memberikan integral ke n dari f dengan memperhatikan variabel independent atas interval [a,b].
4.	<code>int (f, 't', a, b)</code>	Memberikan integral dari f dengan memperhatikan variabel t atas interval [a,b].

Tabel 5.3 Perintah Menghitung Integral Suatu Fungsi

Dari ketiga topik diatas, masih banyak sekali materi mengenai kalkulus, oleh sebab itu di serahkan kepada pembaca untuk mempelajari dan mengembangkannya lebih lanjut. Diharapkan ketiga topik diatas dapat menjadi bekal yang cukup dalam mengembangkan materi kalkulus. Selanjutnya, mari kita melakukan LATIHAN mengenai ketiga materi sebagai berikut:

D. LIMIT MATLAB

Contoh 5.1:

```
>> syms x
>> y=4*x+5
```

Limit dari y

```
>> limit(y,x,3)
```

Atau biasa dilakukan tanpa mendefinisikan fungsinya

```
>> limit((4*x+5),x,3)
```

Sampai di sini, pahami hasil yang diberikan! Apa yang anda temukan, catat dan analisislah!.

Untuk mengetahui limit dari fungsi diatas maka perlu diketahui limit kanan dan limit kiri. Apabila limit kiri tidak sama dengan limit kanan maka disimpulkan bahwa fungsi tersebut tidak mempunyai limit.

Contoh 5.2:

```
>> syms x
```

```
>> y=(2*x^2+3*x)/(sqrt(x^2-x))
```

Limit kiri

```
>> limit(y,x,2,'left')
```

Limit kanan

```
>> limit(y,x,2,'right')
```

Karena **limit kiri** sama dengan **limit kanan** maka disimpulkan y mempunyai limit di $x = 2$

```
>> limit(y,x,2)
```

Sampai di sini, pahami hasil yang diberikan! Apa yang anda temukan, catat dan analisislah!. Kemudian lanjutkan dengan perintah berikut:

Contoh 5.3:

Tentukan turunan dari $y = \tan(x)$. Turunan fungsi didefinisikan sebagai y'

```
>> syms x h
>> y=tan(x)
```

Turunan y

```
>> limit((subs(y,x+h)-y)/h,h,0)
```

Atau bisa dilakukan langsung tanpa pendefinisian terlebih dahulu

```
>> syms h n x
>> limit((tan(x+h)-tan(x))/h,h,0)
```

Pahami hasil yang diberikan! Apa yang anda temukan, catat dan analisislah!.

E. DIFFERENSIAL MATLAB

Contoh 5.4:

```
>> F1 = sym('6*x^3-4*x^2+b*x-5');
>> diff(F1)
ans = 18*x^2-8*x+b

>> diff(F1,2)
```

```

ans = 36*x-8

>> diff(F1,'b')
ans = x

>> F2 = sym('sin(a)');
>> diff(F2)
ans = cos(a)

>> F3 = sym('(1-t^3)/(1+t^4)');
>> diff(F3)
ans = -3*t^2/(1+t^4)-4*(1-t^3)/(1+t^4)^2*t^3

>> simplify(diff(F3))
ans = t^2*(-3+t^4-4*t)/(1+t^4)^2

```

Penggunaan perintah MATLAB untuk menghitung differensial suatu fungsi, dapat dilihat pada Tabel berikut:

F u n g s i	Matlab	Hasil (ans)
F1 = sym('6*x^3-4*x^2+b*x-5')	diff(F1)	$18x^2-8x+b$
	diff(F1,2)	$36x-8$
	diff(F1,'b')	x
F2 = sym('sin(a)')	diff(F2)	cos(a)
F3 = sym('(1-t^3)/(1+t^4)')	diff(F3)	$-\frac{3t^2}{(1+t^4)} - \frac{4(1-t^3)}{(1+t^4)^2 t^3}$

	<code>simplify(diff(F3))</code>	$\frac{t^2(-3+t^4-4t)}{(1+t^4)^2}$
--	---------------------------------	------------------------------------

Pahami hasil yang diberikan! Apa yang anda temukan, catat dan analisislah!.

F. INTEGRAL MATLAB

Contoh 5.4:

```
>> f1 = sym('6*x^3-4*x^2+b*x-5');
>> int(f1)
ans = 3/2*x^4-4/3*x^3+1/2*b*x^2-5*x

>> f2 = sym('sin(a)');
>> int(f2)
ans = -cos(a)

>> f3 = sym('sqrt(x)');
>> int(f3)
ans = 2/3*x^(3/2)

>> int(f3,'a','b')
ans = 2/3*b^(3/2)-2/3*a^(3/2)

>> int(f3,'0.5','0.6')
ans = 0.7413640730107750934739311128097e-1

>> f4 = sym('2*x');
>> int(f4,1,3)
ans = 8
```


Penggunaan perintah MATLAB untuk menghitung integral suatu fungsi polinomial, dapat dilihat pada Tabel berikut :

F u n g s i	Perintah Matlab	Hasil (ans)
f1 = sym('6*x^3- 4*x^2+b*x-5');	int(f1)	$\frac{3}{2}x^4 - \frac{4}{3}x^3 + \frac{1}{2}bx^2 - 5x$
f2 = sym('sin(a)');	int(f2)	$-\cos(a)$
f3 = sym('sqrt(x)') ;	int(f3)	$\frac{2}{3}x^{(3/2)}$
	int(f3,'a','b')	$\frac{2}{3}b^{(3/2)} - \frac{2}{3}a^{(3/2)}$
	int(f3,'0.5','0.6')	0.7413640e
f4 = sym('2*x');	int(f4,1,3)	8

LATIHAN 5 MATLAB



1. Tentukan nilai limit dari fungsi berikut di dalam MATLAB

a) $\lim_{x \rightarrow \sqrt{2}} \frac{x^2 - 2}{x - \sqrt{2}}$

.....

.....

.....

.....

.....

b) $\lim_{x \rightarrow 0} \left(\frac{2}{x-2} - \frac{8}{x^2-4} \right)$

.....

.....

.....

.....

.....

c) $\lim_{x \rightarrow \frac{\pi}{4}} \frac{\cos 2x}{\cos x - \sin x}$

.....

.....

.....

.....

.....

d) $\lim_{x \rightarrow 0} \frac{\cos x - \cos 5x}{x \tan 2x}$

.....

.....

.....

.....

e) $\lim_{x \rightarrow 2} \frac{\sin(x-2)}{x^2 - 3x + 2}$

.....

.....

.....

.....

.....

2. Tentukan $\lim_{x \rightarrow 0^-} f(x)$, $\lim_{x \rightarrow 0^+} f(x)$, $\lim_{x \rightarrow 1^-} f(x)$, dan $\lim_{x \rightarrow 1^+} f(x)$ jika diberikan:

$$f(x) = \begin{cases} 2x-1, & x < 1 \\ \frac{x-1}{x^2-1}, & x > 1 \end{cases}$$

.....

.....

.....

.....

.....

.....

.....

3. Tentukan $\lim_{x \rightarrow 1} f(x)$ jika diberikan:

$$f(x) = \begin{cases} 2x - 1, & x < 1 \\ x^3, & x > 1 \end{cases}$$

.....

.....

.....

.....

.....

.....

.....

4. Tentukan $\lim_{x \rightarrow -1^-} f(x)$, $\lim_{x \rightarrow 0} f(x)$, dan $\lim_{x \rightarrow 3} f(x)$ jika diberikan:

$$f(x) = \begin{cases} 2x - 1, & x \leq 0 \\ \frac{x^2 + 3x}{x^2 - 3x}, & 0 < x < 3 \\ -5x + 1, & x \geq 3 \end{cases}$$

.....

.....

.....

.....

.....

.....
.....
.....

5. Tentukan $\frac{dy}{dx}$ dari persamaan:

a) $y = \tan(x^2)$

.....
.....
.....
.....
.....

b) $y = \sin^2(\cos^2(x^2))$

.....
.....
.....
.....
.....

6. Tentukan Turunan pertama dan kedua dari fungsi berikut:

a) $y = \tan(6x)$

.....
.....
.....
.....
.....

b) $y = \sec(2x)\cos(x)$

.....

.....

.....

.....

.....

7. Tentukan turunan parsial dari fungsi $g = \cos(ab)$

a) Turunan parsial $g(x, b)$ terhadap b dan turunan parsial terhadap a

.....

.....

.....

.....

.....

b) Turunan parsial $g(x, b)$ kedua terhadap b dan turunan parsial kedua terhadap a

.....

.....

.....

.....

.....

8. Selesai integral tak tentu berikut:

$$\int (x^2 - 3x + 2)^3 (2x - 3) dx$$

.....

.....

.....

.....

.....

.....

9. Selesaikan integral tentu berikut:

$$\int_1^4 \left(\frac{s^2 - 8}{s^2} \right) ds$$

.....

.....

.....

.....

.....

.....

.....

10. Selesaikan integral tentu berikut:

$$\int_0^6 \sqrt{4x+1} \, dx$$

.....

.....

.....

.....

.....

.....

.....

.....

DAFTAR PUSTAKA

- Arhami, M., & Desiani, A. 2014. *Pemrograman Matlab*. Yogyakarta.
- Carlson, B.C. 1977., *Special Functions of Applied Mathematics*, Academic Press, New York.
- Cheney W. and D. Kincaid, 1999., *Numerical Mathematics and Computing*, Fourth edition, Brooks/Cole Publishing Company, Pacific Grove.
- Davis P.J. and P. Rabinowitz, 1975., *Methods of Numerical Integration*, Academic Press, New York.
- Fausett, L.V., 1999., *Applied Numerical Analysis Using MATLAB*, Prentice Hall, Upper Saddle River, NJ.
- Hanselman D. and B. Littlefield, 1998., *Mastering MATLAB 5. A Comprehensive Tutorial and Reference*, Prentice Hall, Upper Saddle River, NJ.
- Hanselman, D. & Littlefield, B. 2000. *Mastering MATLAB, A Comprehensive Tutorial and Reference*. Prentice-Hall International, Inc.
- Hanselman D. and B. Littlefield, 2000, *MATLAB Bahasa Komputansi Teknis*, Penerbit ANDI Yogyakarta
- Kuswari Hernawati. 2010. *Aplikasi Komputer*. Yogyakarta: FMIPA Universitas Negeri Yogyakarta.
- Lindfield G. and J. Penny, 1995., *Numerical Methods Using MATLAB*, Ellis Horwood, New York,.
- Mathews J.H. and K.D. Fink, 1999, *Numerical Methods Using MATLAB*, Third edition, Prentice Hall, Upper Saddle River, NJ.

MATLAB, *The Language of Technical Computing. Using MATLAB*, Version 5, The MathWorks, Inc., 1997.

Neuman, Edward, *Tutorial MATLAB*, Department of Mathematics Southern Illinois University at Carbondale

Sri Andayani. 2009. *Handout of Computer Application*. Yogyakarta: FMIPA Universitas Negeri Yogyakarta.

Team Labkomputer UMM. 2011. *Modul Praktikum MATLAB*. Malang: Universitas Muhammadiyah Malang

Teguh Widiarsono. 2005. *Tutorial Praktis Belajar Matlab*. Jakarta.

Zahnur. 2013. *Modul Praktikum Komputasi Matematika*. Aceh: Universitas Syiah Kuala.

PROFIL PENYUSUN



Malim Muhammad adalah Dosen Pendidikan Matematika Fakultas Keguruan dan Ilmu Pendidikan Universitas Muhammadiyah Purwokerto yang lahir di Medan, 07 Februari 1986. Bidang Ilmu yang dikuasai adalah Data Science, Big Data, Komputasi Statistik, Aplikasi Komputer Matematika, Statistika Deskriptif, Statistika Inferensia, dan Pemodelan Matematika. Beliau sudah mengeluarkan Buku Ajar Statistika Inferensia Tahun 2017. Beliau aktif mengikuti seminar baik Nasional dan Internasional serta menghasilkan banyak artikel ilmiah terindeks Scopus, WoS, Sinta dan Lainnya. Beliau pernah mendapatkan presetasi sebagai Pemakalah Terbaik (Best Paper) pada Seminar Nasional Pendidikan Matematika Universitas Ahmad Dahlan (SENDIKMAD) 2018.