

Dasar-dasar Pemrograman Matlab

A.Firmansyah

firman03@gmail.com

Lisensi Dokumen:

Copyright © 2003-2007 IlmuKomputer.Com

Seluruh dokumen di IlmuKomputer.Com dapat digunakan, dimodifikasi dan disebarkan secara bebas untuk tujuan bukan komersial (nonprofit), dengan syarat tidak menghapus atau merubah atribut penulis dan pernyataan copyright yang disertakan dalam setiap dokumen. Tidak diperbolehkan melakukan penulisan ulang, kecuali mendapatkan ijin terlebih dahulu dari IlmuKomputer.Com.

1. Pendahuluan

Matlab merupakan bahasa pemrograman yang hadir dengan fungsi dan karakteristik yang berbeda dengan bahasa pemrograman lain yang sudah ada lebih dahulu seperti Delphi, Basic maupun C++. Matlab merupakan bahasa pemrograman level tinggi yang dikhususkan untuk kebutuhan komputasi teknis, visualisasi dan pemrograman seperti komputasi matematik, analisis data, pengembangan algoritma, simulasi dan pemodelan dan grafik-grafik perhitungan.

Matlab hadir dengan membawa warna yang berbeda. Hal ini karena matlab membawa keistimewaan dalam fungsi-fungsi matematika, fisika, statistik, dan visualisasi. Matlab dikembangkan oleh MathWorks, yang pada awalnya dibuat untuk memberikan kemudahan mengakses data matrik pada proyek LINPACK dan EISPACK. Saat ini matlab memiliki ratusan fungsi yang dapat digunakan sebagai *problem solver* mulai dari simple sampai masalah-masalah yang kompleks dari berbagai disiplin ilmu.

2. Lingkungan Kerja Matlab

2.1 Beberapa Bagian dari Window Matlab

- **Current Directory**

Window ini menampilkan isi dari direktori kerja saat menggunakan matlab. Kita dapat mengganti direktori ini sesuai dengan tempat direktori kerja yang diinginkan. Default dari alamat direktori berada dalam folder works tempat program files Matlab berada.

- **Command History**

Window ini berfungsi untuk menyimpan perintah-perintah apa saja yang sebelumnya dilakukan oleh pengguna terhadap matlab.

- **Command Window**

Window ini adalah window utama dari Matlab. Disini adalah tempat untuk menjalankan fungsi, mendeklarasikan variable, menjalankan proses-proses, serta melihat isi variable.

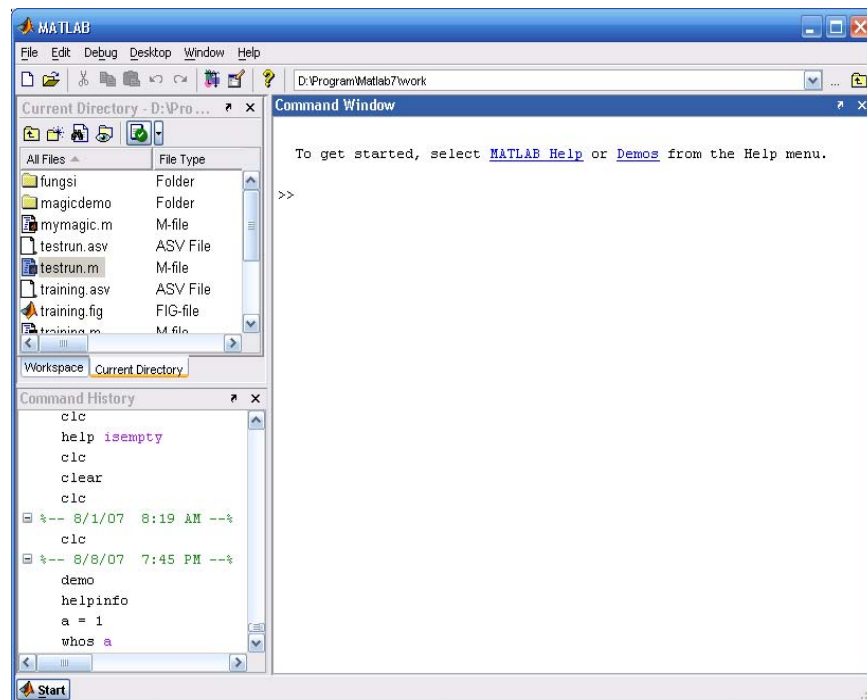
Komunitas eLearning IlmuKomputer.Com

Copyright © 2003-2007 IlmuKomputer.Com


- **Workspace**

Workspace berfungsi untuk menampilkan seluruh variabel-variabel yang sedang aktif pada saat pemakaian matlab. Apabila variabel berupa data matriks berukuran besar maka user dapat melihat isi dari seluruh data dengan melakukan double klik pada variabel tersebut. Matlab secara otomatis akan menampilkan window “array editor” yang berisikan data pada setiap variabel yang dipilih user

Gambar berikut menampilkan tampilan antar muka dari matlab versi 7.0



2.2 Getting Help

Matlab menyediakan fungsi help yang tidak berisikan tutorial lengkap mengenai Matlab dan segala keunggulannya. User dapat menjalankan fungsi ini dengan menekan tombol  pada toolbar atau menulis perintah 'helpwin' pada command window. Matlab juga menyediakan fungsi demos yang berisikan video tutorial matlab serta contoh-contoh program yang bisa dibuat dengan matlab

2.3 Interrupting dan Terminating dalam Matlab

Untuk menghentikan proses yang sedang berjalan pada matlab dapat dilakukan dengan menekan tombol **Ctrl-C**. Sedangkan untuk keluar dari matlab dapat dilakukan dengan menuliskan perintah **exit** atau **quit** pada command window atau dengan menekan menu exit pada bagian menu file dari menu bar.

3. Variabel Pada Matlab

Matlab hanya memiliki dua jenis tipe data yaitu Numeric dan String. Dalam matlab setiap variabel akan disimpan dalam bentuk matrik. User dapat langsung menuliskan variabel baru tanpa harus mendeklarasikannya terlebih dahulu pada command window

Contoh pembuatan variabel pada matlab:

```
>> varA = 1000
varA =
    1000
>> varB = [45 2 35 45]
varB =
    45     2    35    45
>> varC = 'test variabel'
varC =
    test variabel
```

Penamaan variabel pada matlab bersifat caseSensitif karena itu perlu diperhatikan penggunaan huruf besar dan kecil pada penamaan variabel. Apabila terdapat variabel lama dengan nama yang sama maka matlab secara otomatis akan *me-replace* variabel lama tersebut dengan variabel baru yang dibuat user.

3.1 Matriks

Dapat diasumsikan bahwa didalam matlab setiap data akan disimpan dalam bentuk matriks. Dalam membuat suatu data matriks pada matlab, setiap isi data harus dimulai dari kurung siku '[' dan diakhiri dengan kurung siku tutup ']'. Untuk membuat variabel dengan data yang terdiri beberapa baris, gunakan tanda 'titik koma' (;) untuk memisahkan data tiap barisnya.

Contoh pembuatan data matriks pada matlab:

```
>> DataMatriks = [1 2 3;4 5 6]
DataMatriks =
     1     2     3
     4     5     6
```

Matlab menyediakan beberapa fungsi yang dapat kita gunakan untuk menghasilkan bentuk-bentuk matriks yang diinginkan. Fungsi-fungsi tersebut antara lain:

- zeros : untuk membuat matriks yang semua datanya bernilai 0
- ones : matriks yang semua datanya bernilai 1
- rand : matriks dengan data random dengan menggunakan distribusi uniform
- randn : matriks dengan data random dengan menggunakan distribusi normal
- eye : untuk menghasilkan matriks identitas

contoh penggunaan fungsi-fungsi diatas:

```
>> a = zeros(2,3)
a =
```

```

0 0 0
0 0 0

>> b = ones(1,3)

b =
    1    1    1

>> c = rand(2,2)

c =
    0.9501    0.6068
    0.2311    0.4860

>> d = rand (1,4)

d =
    0.8214    0.4447    0.6154    0.7919

>> e = eye(3,3)

e =
    1    0    0
    0    1    0
    0    0    1

```

Untuk memanggil isi dari suatu data matriks, gunakan tanda kurung ‘()’ dengan isi indeks dari data yang akan dipanggil. Contoh penggunaan :

```

>> c(2,2)

ans =
    0.4860

```

Untuk pemanggilan data berurutan seperti a(1,2,3) dapat disingkat dengan menggunakan tanda titik dua ‘:’ sehingga menjadi a(1:2). Penggunaan tanda titik dua ‘:’ juga dapat digunakan untuk memanggil data matriks perbaris atau perkolom.

Contoh penggunaan:

- c(2:5) = memanggil data matrik baris 2 sampai baris 5
- a(1,:) = memanggil data matriks pada baris pertama
- b(:,3) = memanggil data matriks pada kolom ketiga

4. Operator

Beberapa penggunaan operator aritmatika antara dua operand (A dan B) ditunjukkan pada tabel berikut ini

Operasi	Bentuk Aljabar	Bentuk Matlab	Contoh
Perkalian	$A \times B$	$A * B$	$5*3$
Pembagian	$A \div B$	$A \backslash B$	$2\backslash 3$
Penambahan	$A + B$	$A + B$	$1+2$
Pengurangan	$A - B$	$A - B$	$4-3$
Eksponensial	A^B	$A ^ B$	4^3

5. Fungsi Matematika lainnya

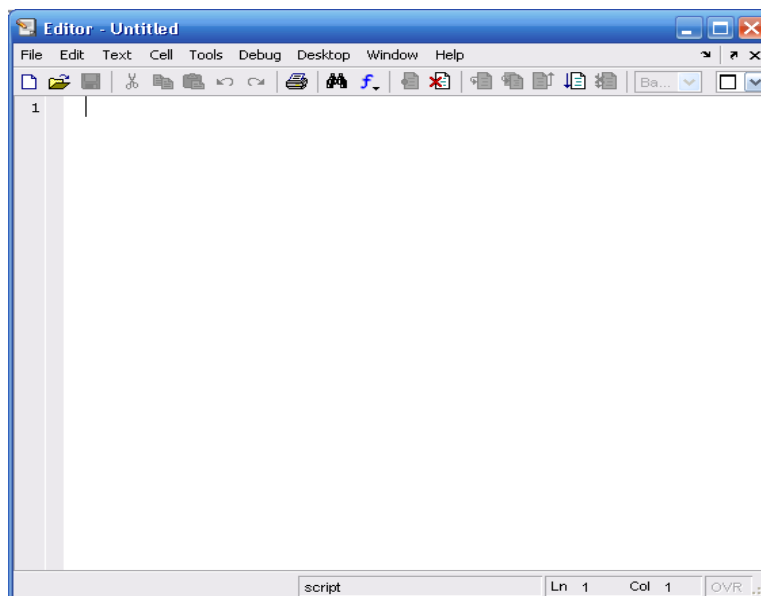
Beberapa fungsi matematika lainnya yang dapat kita gunakan untuk operasi matematika antara lain sebagai berikut:


- $\text{abs}(x)$: fungsi untuk menghasilkan nilai absolut dari x
- $\text{sign}(x)$: fungsi untuk menghasilkan nilai -1 jika $x < 0$, 0 jika $x = 0$ dan 1 jika $x > 0$
- $\text{exp}(x)$: untuk menghasilkan nilai eksponensial natural, e^x
- $\text{log}(x)$: untuk menghasilkan nilai logaritma natural x , $\ln x$
- $\text{log10}(x)$: untuk menghasilkan nilai logaritma dengan basis 10, $\log_{10} x$
- $\text{sqrt}(x)$: untuk menghasilkan akar dari nilai x , \sqrt{x}
- $\text{rem}(x,y)$: untuk menghasilkan nilai modulus (sisa pembagian) x terhadap y

6. M File

Di dalam matlab, kita dapat menyimpan semua script yang akan digunakan dalam file pada matlab dengan ekstensi .M. M-File dapat dipanggil dengan memilih menu **file->new->M-File**.

Contoh gambar M-File



Di dalam M-File, kita dapat menyimpan semua perintah dan menjalankannya dengan menekan tombol  atau mengetikkan nama M-File yang kita buat pada command window.

6.1 Fungsi

Di dalam M File, kita dapat menuliskan fungsi-fungsi yang berisikan berbagai operasi sehingga menghasilkan data yang diinginkan.

Bentuk penulisan nama fungsi

```
Function [Nilai keluaran ] = namaFungsi (nilai masukan)

% operasi dari fungsi
% ...
% ...
```

Contoh penggunaan:

fungsi yang akan dibuat bernama 'testfungsi' memiliki tiga nilai masukan 'c,d,e' dan dua nilai keluaran 'a,b':

```
function [a,b] = testFungsi(c,d,e)

%operasi yang dijalankan
a = c + d +e;
b = c * d *e;
```

Selanjutnya Fungsi tersebut akan dijalankan melalui command window dengan nilai masukan '10,2,4'. Perhatikan penulisan kurung siku '[']' pada nilai keluaran dan kurung biasa '(')' pada nilai masukan.

```
>> [a,b] = testFungsi(10,2,3)

a =

    15

b =

    60
```

6.2 Flow Control

Matlab memiliki empat macam statement yang dapat digunakan untuk mengatur aliran data pada fungsi yang akan dibuat

1. If, Else, Elseif

Bentuk dasar penggunaan statement jenis ini adalah sebagai berikut:

```
if ekspresi1
    statements1;

elseif ekspresi2
    statements2;

else
    statements3;

end
```

Ekspresi akan bernilai 1 jika benar dan bernilai 0 jika salah.

Contoh penggunaan:

```
function testFungsi(A,B)
if A > B
    disp('A lebih besar dari B')
elseif A == B
    disp('A sama dengan B')
else
    disp('A lebih kecil dari B')
end
```

Fungsi disp digunakan untuk menampilkan pesan pada command window.
Fungsi tersebut setelah dijalankan melalui command window:

```
>> testFungsi(1,2)
    A lebih kecil dari B
>> testFungsi(2,2)
    A sama dengan B
>> testFungsi(2,3)
    A lebih kecil dari B
```

2. Switch

Bentuk dasar penggunaan statement switch

```
switch switch_ekspresi
case case_ekspresi1
    statement1
case case_ekspresi2
    statement2
...
...
otherwise
    statementN
end
```

Contoh penggunaan:

```
function testFungsi(x)
switch x
case 1
    disp('x is 1')
case {2,3,4}
    disp('x is 2, 3 or 4')
case 5
    disp('x is 5')
otherwise
    disp('x is not 1, 2, 3, 4 or 5')
end
```

Hasil setelah dijalankan

```
>> testFungsi(2)
x is 2, 3 or 4

>> testFungsi(1)
x is 1

>> testFungsi(5)
x is 5

>> testFungsi(6)
x is not 1, 2, 3, 4 or 5
```

3. while

Statement while digunakan untuk aliran data yang bersifat perulangan.

Bentuk dasar penggunaan while

```
while ekspresi
    statements
    ...
end
```

Contoh penggunaan

```
function testFungsi(x)
%selama nilai x kurang dari 10
while x < 10
    disp('nilai saat ini : ');
    %tampilkan nilai x
    x
    %increment nilai x
    x = x+1;
end
```

Hasil setelah dijalankan

```
>> testFungsi(6)
nilai saat ini :
x =
     6
nilai saat ini :
x =
     7
nilai saat ini :
x =
     8
nilai saat ini :
x =
     9
```


4. for

Bentuk dasar penggunaan bentuk for:

```
for index = start:increment:stop
    statement
    ...
    ...
    statement
end
```

Default dari nilai increment (penambahan nilai setiap perulangan) jika tidak ditentukan oleh user adalah 1.

Contoh fungsi :

```
function a = testFungsi
for y = 1:10
    a(y) = y^2;
end
```

Hasil setelah fungsi dijalankan

```
>> a = testFungsi
a =
     1     4     9    16    25    36    49    64    81   100
```

5. Operator

Berikut ini adalah jenis-jenis operator pada matlab yang dapat digunakan untuk operasi ekspresi pada statement yang membutuhkan perbandingan seperti **if** atau **while**.

Operator	Keterangan
$A < B$	A lebih kecil dari B
$A > B$	A lebih besar dari B
$A \leq B$	A lebih kecil atau sama dengan B
$A \geq B$	A lebih besar atau sama dengan B
$A == B$	A sama dengan B
$A \sim B$	A tidak sama dengan B

6. Penutup

Disini telah diulas, dasar-dasar dari matlab. Sebenarnya masih banyak fasilitas-fasilitas lain yang dimiliki oleh matlab. Tugas-tugas matematika yang membutuhkan analisis ataupun perhitungan yang kompleks dan rumit dapat kita implementasikan dengan mudah dengan memanfaatkan fasilitas yang terdapat di matlab. Penggunaan dan manipulasi grafik serta pemrograman GUI pada matlab, insyaAllah akan penulis bahas pada artikel berikutnya.

Dibalik ini semua, penulis yakin masih banyak kekurangan dibalik tulisan ini. Pertanyaan, Saran maupun kritik dari pembaca, penulis harapkan sebagai masukan agar dapat menulis lebih baik ke depannya.

Referensi

1. Andrew knight, *Basics of Matlab and Beyond*, CHAPMAN & HALL/CRC, 1999.
2. Gunaidi Abdia Away, *the shortcut of Matlab*, Informatika Bandung, Juni 2006
3. *Matlab User's Guide*, The Math Works inc, 1989.

Biografi Penulis

Penulis bernama lengkap Ahmad Firmansyah dilahirkan di Palembang, 28 Juni 1985, merupakan anak keempat dari empat bersaudara. Penulis menyelesaikan pendidikan di Jurusan Teknik Informatika-Institut Teknologi Sepuluh Nopember Surabaya pada tahun 2007. Di Jurusan Teknik Informatika ini, Penulis mengambil bidang minat Sistem Bisnis Cerdas. Pernah dipercaya menjadi asisten mata kuliah pengenalan pola dan asisten teknologi Database di PIKTI.

Penulis dapat dihubungi melalui email ke firman03@gmail.com atau via YM dengan id: [xenin3](#)