



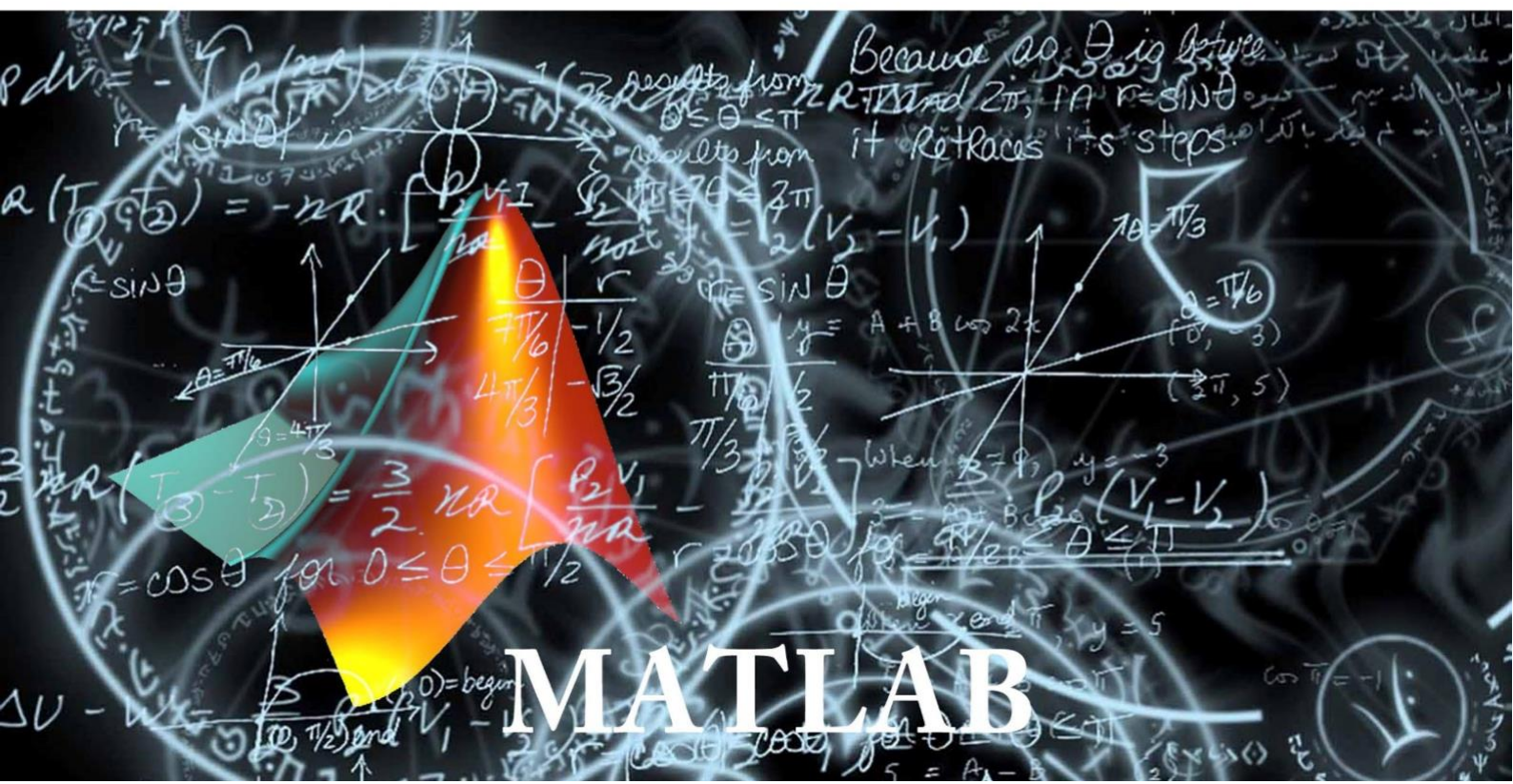
**MATLAB**

# Modul Praktikum Algoritma dan Pemrograman

Penyusun:

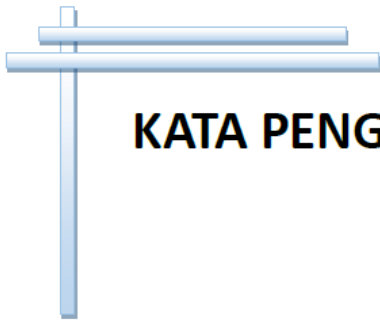
Juhari, M.Si

Hisyam Fahmi, M.Kom



## DAFTAR ISI

	Halaman
KATA PENGANTAR .....	2
MODUL 1. Pendahuluan .....	3
MODUL 2. Operasi Lembar Kerja .....	6
2.1 Memulai Matlab .....	7
2.2 Beberapa Fungsi Dalam Matlab .....	7
MODUL 3. Operasi Fungsi .....	11
MODUL 4. Beberapa Fungsi Dalam Program Matlab .....	15
4.1 Operasi Array, Vektor, dan Matrik .....	15
4.2 Pembentukan <i>array</i> dan matriks .....	15
MODUL 5. Kontrol Flow .....	19
MODUL 6. Menggambar Grafik.....	26
MODUL 7. Analisis Data.....	35
MODUL 8. Guide Matlab .....	40



## KATA PENGANTAR

Puji syukur kehadiran Allah S.W.T karena atas anugerah dan karuniaNya penulis dapat menyelesaikan modul praktikum “Algoritma dan Pemrograman”. Modul ini dibuat untuk membantu mahasiswa mengimplementasikan algoritma yang ditemui dalam beberapa mata kuliah, Program Komputer, Analisis Numerik, Persamaan Diferensial Biasa, Persamaan Diferensial Parsial, Statistika, dan mata kuliah lain ke dalam suatu bahasa program.

Dalam modul ini dijelaskan bagaimana proses pengimplementasian itu dilakukan dalam Matlab. Dengan beberapa fungsi–fungsi khusus yang sudah “*built-in*” dalam Matlab *Library*, mahasiswa diharapkan dapat dengan mudah membuat program dalam bahasa non-prosedural yang bersifat singkat dan lugas namun dapat mengatasi semua masalah – masalah kompleks dalam Matematika.

Selanjutnya dalam kesempatan ini penulis menyampaikan banyak terima kasih kepada yang terhormat:

1. Dekan Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang;
2. Ketua Program Studi Matematika yang telah memberikan motivasi dan rekomendasi penggunaan modul praktikum ini dalam beberapa acara perkuliahan mata kuliah;
3. Semua pihak yang terlibat langsung maupun tak langsung dalam penyusunan modul ini.

Semoga bantuan rielnnya mendapat balasan yang setimpal dari Allah S.W.T. dan akhirnya penulis berharap agar modul ini memberikan manfaat bagi mahasiswa dan pembaca pada umumnya, oleh karena itu kritik dan saran masih penulis harapkan untuk penyempurnaan dikemudian hari.

Malang, Juli 2022

# MODUL 1

## Pendahuluan

**Tujuan Khusus Praktikum :** *Mahasiswa dapat memahami kegunaan software Matlab dan juga mengetahui keunggulan-keunggulan software*

MATLAB (Matrix Laboratory) 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
- Pembentukan Algorithm
- Akuisisi Data
- Pemodelan, simulasi dan Pembuatan Prototype
- Analisis Data, Explorasi, dan Visualisasi Grafik
- Keilmuan dan Bidang Rekayasa

Beberapa keunggulan Matlab terletak pada:

### 1. Aspek Komputasional

- Analisa matrik dan manipulasinya
- Reduksi data dan pengolahan data statistik
- FFT, statistic korelasi dan kovarian
- Pendukung matrik “sparse”
- Fungsi trigonometri dan beberapa fungsi kompleks lainnya
- Fungsi Bessel, beta, dan fungsi kepadatan lainnya
- Persamaan diferensial linier dan nonlinier

### 2. Aspek Grafik dan Visualisasi

- 2-D scatter, grafik garis, poligon dan mesh, counter, grafik polar, dan plot histogram
- 3-D scatter, grafik garis, poligon, mesh dan plot “wireframe”
- Grafik dengan variasi permukaan disertasi dengan animasi gambar dan suara

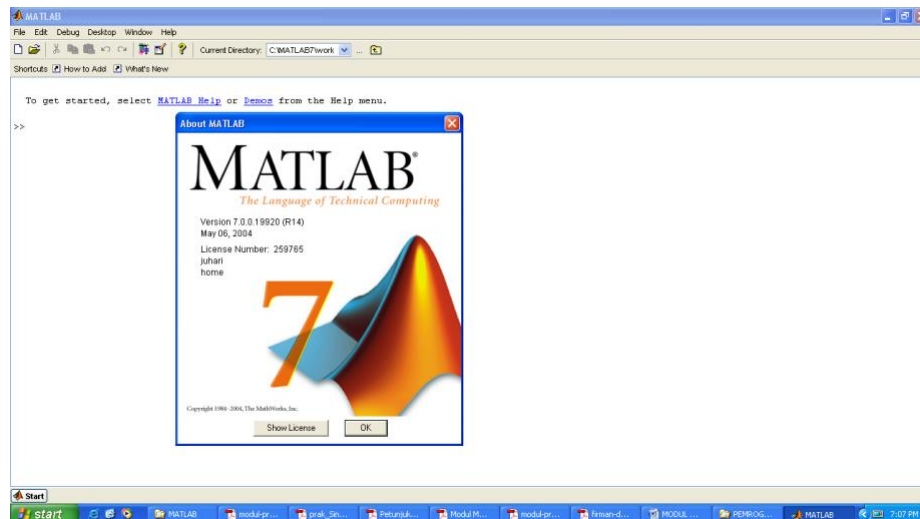


### 3. Aspek Pemrograman

- Struktur control (FOR, WHILE dan IF)
- Manipulasi string
- Input file berupa ASCII dan biner
- Debugging
- Dapat berinteraksi dengan bahasa pemrograman C

### 4. Aspek GUI (Graphical User Interface)

- Menu pull-down dan pop-up
- Push buttons, radio buttons, check boxes, sliders, dan dialog tek yang dapat diedit
- Mouse events dan callbacks



Gambar 1.1 Menu dan symbol utama Matlab.

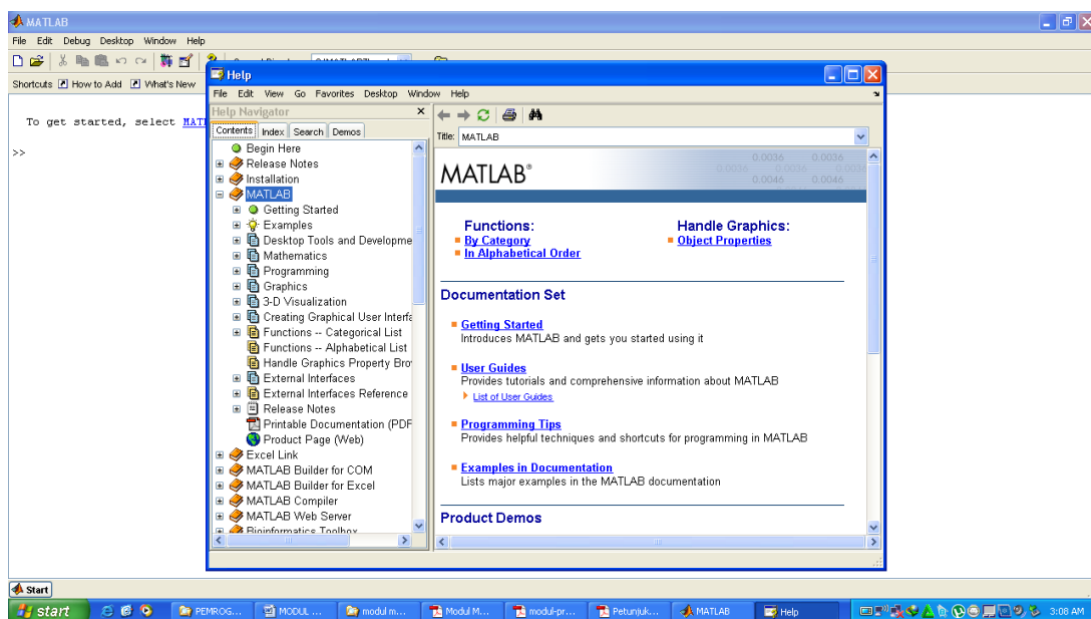
Disamping sebagai compiler, Matlab juga dilengkapi dengan Toolbox software yang merupakan satu-satu kecanggihan Matlab. Software ini mencakup berbagai masalah – masalah besar dalam teknologi tingkat tinggi, diantaranya adalah:

- **Control system Toolbox**, merupakan kumpulan fungsi-fungsi Matlab untuk pemodelan, analisis dan desain system kontrol otomatis.
- **Financial Toolbox**, merupakan software untuk menyelesaikan beberapa masalah keuangan dari masalah yang sederhana sampai masalah yang cukup kompleks.
- **Frequency Domain System Identification Toolbox**, merupakan tools khusus untuk mengidentifikasi system dinamis linier dari respon waktu dan frekuensi.
- **Fuzzy Logic Toolbox**, merupakan software untuk mengembangkan desain “fuzzy” dari tahap “setup” sampai diagnose.

- **Signal Processing Toolbox**, merupakan tools untuk menyelesaikan masalah besar dalam analisa bispektral, model signal linier dan nonlinier, transformasi FFT dan DCT serta visualisasi spectrum.
- **Spectral Analysis Toolbox**, merupakan tools untuk menganalisa signal dengan menggunakan “cumulant” atau spectral dengan order tinggi.
- **Image Processing Toolbox**, merupakan software khusus dalam matlab untuk desain filter, analisa citra, manipulasi warna dan lain-lain yang berkenaan dengan visualisasi citra.
- **Statistics Toolbox**, merupakan software yang menangani masalah-masalah stokastik.
- **System Identification Toolbox**, merupakan software untuk melakukan aktifitas desain system dinamis yang berdasarkan pada input dan output data.

## **PRAKTIKUM PERTAMA**

Silahkan pelajari sendiri tentang Matlab dengan menekan tombol F1 pada keyboard !



Gambar 1.2 Matlab Help

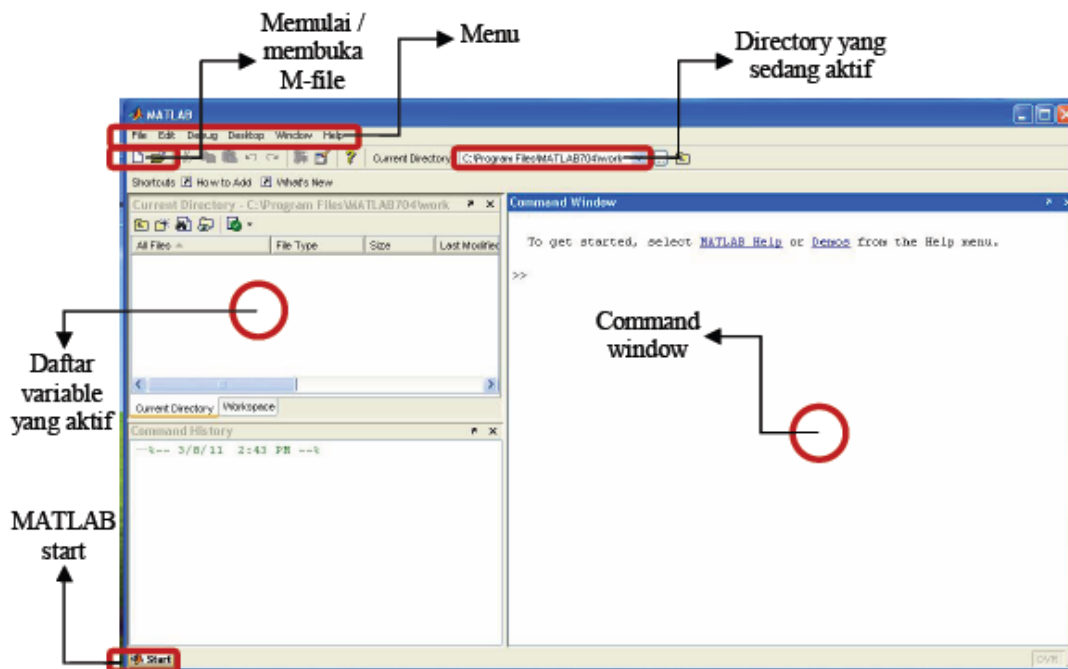
## MODUL 2

### Operasi Lembar Kerja

**Tujuan Khusus Praktikum :** Mahasiswa mengetahui bagaimana mendefinisikan dan menjalankan fungsi-fungsi yang ada dalam Matlab.

#### 2.1 Memulai Matlab

Setelah melakukan instalasi MATLAB pada PC, perhatikan icon MATLAB pada tampilan desktop kemudian “doubleclick” pada icon tersebut. Selanjutnya akan muncul tampilan seperti pada gambar berikut ini.



Gambar 2.1 Lembar kerja untuk menjalankan fungsi-fungsi Matlab

Pada tampilan awal MATLAB, terlihat beberapa jendela yang merupakan bagian penting di dalam MATLAB, antara lain :

a. Jendela perintah (Command Window)

Pada command window, semua perintah matlab dituliskan dan dieksekusi. 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 '>>'.

b. Jendela ruang kerja (Workspace)

Jendela ini berisi informasi penggunaan variabel di dalam memori MATLAB.

c. **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 cypaste ke command window.

## **2.2 Beberapa Fungsi Dalam Matlab**

Beberapa fungsi yang dapat dipakai dalam Matlab untuk menyelesaikan beberapa masalah dapat diringkas dalam penjelasan berikut:

1. **Fungsi pengatur umum**

>> *help* fungsi : untuk mengetahui petunjuk pemakaian suatu fungsi  
>> *type* file.m : untuk menampilkan isi dari M-File  
>> *pwd* : untuk mengetahui subdirektori aktif  
>> *cd* a\ data : memindahkan suatu direktori aktif ke direktori lain yaitu A dalam subdirektori data  
>> *dir* : untuk menampilkan isi direktori  
>> *!ren* file 1.txt file 1.m : merubah nama file1.txt menjadi file 1.m

2. **Fungsi pengatur variabel dan areal kerja**

>> *save* filename : untuk menyimpan variabel dalam file.mat  
>> *load* filename : untuk memanggil data yang disimpan dalam file.mat  
>> *clear* : untuk menghapus variabel terdefinisi  
>> *pack* : untuk memampatkan pemakaian memory lembar kerja  
>> *size*(A) : untuk mengetahui ordo matrik A  
>> *max*(A) atau *min*(A) : untuk mengetahui nilai terbesar dan terkecil dari elemen matrik A  
>> *length*(A) : menginformasikan bilangan terbesar dari ordo matrik A  
>> *clc*: membersihkan layar lembar kerja

3. **Operator numerik dan matrik**

>>  $\pm$  : penjumlahan dan pengurangan  
>>  $*$ ,  $^$  : perkalian dan perpangkatan  
>>  $/$ ,  $\backslash$  : pembagian kanan untuk bilangan dan pembagian kiri untuk matrik dan vektor  
>>  $'$  : transpose vektor atau matrik

4. **OPERATOR array**



>>  $\pm$  : penjumlahan dan pengurangan  
 >>  $.*, .^$  : perkalian dan perpangkatan  
 >>  $./, .\backslash$  : pembagian kanan untuk bilangan dan pembagian kiri untuk matrik dan vektor  
 >>  $'$  : transpose vektor atau matrik

Penambahan titik dalam operator array disebabkan adanya operasi sederetan bilangan dalam waktu yang bersamaan. Contoh array  $x = 0:0.1:10$

#### 5. Operator logika dan relasional

>>  $<, <=$  : lebih kecil dan lebih kecil sama dengan  
 >>  $>, >=$  : lebih besar dan lebih besar sama dengan  
 >>  $=$  : sama atau ekuivalen  
 >>  $\sim$  : tidak sama atau tidak ekuivalen  
 >>  $\&, |, \sim$  : dan, atau, tidak

#### 6. Penulisan fungsi matematika

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

#### 7. Fungsi M-file

>>  $disp('karakter')$  : menampilkan karakter (string)  
 >>  $num2str$  : mengkonversi numerik menjadi string  
 >>  $input$  : meminta user memberikan input  
 >>  $pause$  : menghentikan program sampai user menekan <ENTER>  
 >>  $pause(n)$  : berhenti selama n detik

## **PRAKTIKUM 2A**

1. Bukalah program Matlab pada computer anda !
2. Ketikkan sintak perintah-perintah berikut ini dengan huruf kecil diikuti dengan menekan tombol ENTER !
3. Amati hasil tampilan di layar untuk setiap sintak perintah !
4. Tuliskan fungsi dari setiap sintak perintah tersebut pada kolom yang disediakan pada tabel praktikum !

No.	Sintak	Fungsi
1	Intro	
2	Demo	
3	help input	
4	help disp	
5	Pwd	
6	d=pi	
7	eps	
8	Realmin	
9	Realmax	
10	n/0	
11	round(d)	
12	floor(d)	
13	0/0	
14	format rat	
15	exp(5)	
16	e=input('nilai e=')	
17	disp('nilai yang anda masukkan : ');e	
18	format bank	
19	100/8	
20	f='belajar pemrograman'	
21	double(f)	
22	char(f)	

## **PRAKTIKUM 2B**

1. Bukalah M-File baru dengan menu *File-New-M-File*
2. Ketikkan program sederhana ini dan simpan dengan nama “praktikum1B” !
3. Jalankan program dengan menu *Tools-Run*, jika disimpan di folder pribadi ubah directory dengan menu *File-Set Path-Browse* pada jendela kerja (*command window*).
4. Amati hasil program, jika ada pesan kesalahan lihat pesan di jendela kerja dan perbaiki program di *M-File*, jangan lupa simpan dan jalankan lagi.

```
% Kelipatan 7 kurang dari 1000 habis dibagi 7,  
% sisa 1 jika dibagi 2, 3, 4, 5 dan 6  
  
n=7:7:1000; % semua kelipatan 7 yang kurang dari 1000  
number=length(n) % jumlah penyelesaian yang mungkin  
  
n(mod(n,2)~=1)=[]; % membuang yang bukan solusi  
number=length(n)  
n(mod(n,3)~=1)=[]; % mengesetnya sama dengan matriks kosong  
number=length(n)  
n(mod(n,4)~=1)=[]; % fungsi mod menghitung sisa hasil bagi  
number=length(n)  
n(mod(n,5)~=1)=[];  
number=length(n)  
n(mod(n,6)~=1)=[];  
number=length(n)  
n
```

## **PRAKTIKUM 2C**

1. Buatlah satu *M-File* program utama untuk menentukan sudut  $x$  dari nilai

$$\cos x = 0; \frac{1}{2}; \frac{1}{2}\sqrt{2}; \frac{1}{2}\sqrt{3}; 1$$

$$\sin x = 0; \frac{1}{2}; \frac{1}{2}\sqrt{2}; \frac{1}{2}\sqrt{3}; 1$$

$$\tan x = 0; \frac{1}{2}; \frac{1}{2}\sqrt{2}; \frac{1}{2}\sqrt{3}; 1$$

Petunjuk : berlatihlah dengan memecahkan soal ini secara bersamaan dalam satu *listing* program. Gunakan manipulasi *array* untuk menyatakan hasilnya dalam bentuk baris.

## MODUL 3

### Operasi Fungsi

**Tujuan Khusus Praktikum :** *Mahasiswa dapat menulis program dalam programming editor dan dapat menjalankannya dalam lembar kerja Matlab*

Ada dua cara Matlab dalam mendefinisikan suatu fungsi. Pertama secara langsung, yaitu dengan memberikan sintak perintah **inline** dalam M-File program utama atau bisa juga pada jendela kerja secara langsung. Sintak perintah ini membutuhkan nama fungsi, definisi fungsi, dan nama variable bebas sebagai data masukan fungsi, dimana dua terakhir ditulis terpisah oleh tanda koma, dengan diapit oleh tanda petik satu (*apostrof*) dan dalam tanda kurung.

```
f = inline('definisi fungsi', 'variabel1','variabel2',...)
```

Perintah fungsi ini dapat dijalankan dengan mengetikkan nama fungsi diikuti nilai variabelnya dalam tanda kurung:

```
f(nilai1,nilai2,...)
```

atau dengan menggunakan sintak perintah **feval** yang diikuti dengan nama fungsi dan nilai variabel yang terpisah dengan tanda koma dalam tanda kurung:

```
feval(f,nilai1,nilai2,...)
```

cara kedua adalah cara tidak langsung, yaitu dengan mendefinisikan fungsi pada *M-file* yang lain, terpisah dengan *M-file* program utama. *M-file* fungsi ini harus disimpan dengan nama sesuai dengan nama fungsinya dan pada directory yang sama pula dengan program utamanya. *M-file* fungsi harus diawali dengan sintak perintah **function** dan diikuti dengan nama variabel output, nama fungsi, nama variabel inputannya:

```
function varoutput = namafungsi(varinput1,varinput2,...)
```

kemudian diikuti dengan definisi fungsinya. Cara kedua ini di khususkan untuk definisi fungsi yang cukup panjang sehingga tidak cukup dalam satu baris sebagaimana cara pertama.

Untuk menjalankan M-file fungsi ini dilakukan sama dengan cara sebelumnya, yaitu dengan langsung mengetikkan nama fungsinya yang diikuti oleh nilai variabel inputnya, ataupun dengan sintak **feval**, dimana nama fungsinya diapit tanda petik satu.

### **PRAKTIKUM 3A**

1. Bukalah program Matlab pada computer anda !
2. Ketikkan sintak perintah-perintah berikut ini dengan huruf kecil diikuti dengan menekan tombol ENTER ! (abaikan tampilan warning dari sintak no.5)
3. Jika ditulis pada M-File pisahkan untuk no.5 tersendiri !
4. Amati hasil tampilan di layar untuk setiap sintak perintah !
5. Tuliskan fungsi dari setiap sintak perintah tersebut pada kolom yang disediakan pada tabel praktikum !

No.	Sintak	Keterangan
1	<code>f = inline('x^2+x-7', 'x')</code>	
2	<code>f(0)</code>	
3	<code>f(-5)</code>	
4	<code>feval(f,6)</code>	
5	<code>function y = g(x)</code> <code>y = x^3+5*x^2+2*x-5;</code>	
6	<code>g(8)</code>	
7	<code>feval('g',-7)</code>	
8	<code>f(0)+ g(0)</code>	
9	<code>f(g(0))</code>	
10	<code>g(f(0))</code>	
11	<code>sin(30)</code>	
12	<code>sin(30*pi/180)</code>	
13	<code>sin(30*pi/180)^2</code>	
14	<code>Sin((30*pi/180)^2)</code>	
15	<code>cos(sin(pi))</code>	



### **PRAKTIKUM 3B**

1. Bukalah M-File baru dengan menu *File-New-M-File*
2. Ketikkan *M-File function* dibawah ini dan simpan dengan nama “akar”
3. Bukalah M-File baru dengan menu *File-New*
4. Ketikkan program utama ini dan simpan dengan nama “praktikum2B” !
5. Jalankan program dengan menu *Tools-Run*, jika disimpan di folder pribadi ubah directory dengan menu *File-Set Path-Browse* pada jendela kerja (*command window*).
6. Amati hasil program, jika ada pesan kesalahan lihat pesan di jendela kerja dan perbaiki program di *M-File*, jangan lupa simpan dan jalankan lagi.

```
Function[x1,x2]=akar(a,b,c)           %fungsi dengan nama akar
                                       %tiga input a,b,c
                                       %output x1 dan x2

d = b^2-4*a*c;
x1 = (-b + sqrt(d))/2*a;
x2 = (-b - sqrt(d))/2*a;
```

```
%praktikum 2 program computer matlab
%program operasi fungsi dengan menggunakan M-file function
clc;clear;format short

disp('=====')
disp('pencari akar real persamaan kuadrat')
disp('          f(x)=ax^2+b*x+c = 0          ')
disp('          JUHARI          ')
disp('          016010057          ')
disp('=====')

a = input ('masukkan nilai koefisien a=');
b = input ('masukkan nilai koefisien b=');
c = input ('masukkan nilai koefisien c=');

disp('persamaan kuadrat yang anda masukkan adalah : ')
disp([num2str(a),'*x^2+',num2str(b),'*x+', num2str(c),'=0'])
disp(['akar pertama yaitu',num2str(x1)])
disp(['akar pertama yaitu',num2str(x2)])

disp('=====')
disp('          terima kasih          ')
disp('jika anda ingin mencoba lagi ketikkan praktikum2B')
disp('=====')
```

## PRAKTIKUM 3C

1. Ketikkan sintak perintah untuk mendefinisikan fungsi-fungsi berikut ini pada *command window*:

$$h(x) = \left(\frac{x}{2,5}\right)^3 - \frac{2}{5}x - \cos(x\pi)$$

$$k(x, y) = x^5 + 3x^4y - 2xy + 8x - 5y + 7$$

2. Berikan perintah pada *command window* untuk mengetahui nilai fungsi dari  $h(6.5)$ ,  $h(-3)$ ,  $h(0)$ ,  $k(0,0)$ ,  $k(4,5)$ ,  $k(-7,-9)$
3. Buatlah dua M-file function untuk mendefinisikan kedua fungsi diatas dan simpang masing-masing sesuai dengan nama fungsinya !
4. Buatlah satu M-file program utama untuk menghitung nilai dari kedua fungsi tersebut dari data yang dimasukkan secara input, dengan susunan perintah:
  - a. Menampilkan deskripsi program
  - b. Meminta dua nilai bilangan riil (decimal) tak nol sebagai data masukan input
  - c. Tampilkan nilai dari kedua fungsi tersebut !
5. Simpan M-file program anda dengan nama “praktikum2C” !
6. Jalankan program anda !

```
%latihan program computer Matlab praktikum 3C
```

This image shows a blank sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.

## MODUL 4

### Beberapa Fungsi Dalam Program Matlab

**Tujuan Khusus Praktikum :** Mahasiswa dapat menjalankan fungsi-fungsi Matlab untuk menyelesaikan masalah tertentu yang ditulis dalam programming editor

#### 4.1 Operasi Array, Vektor, dan Matrik

*Array* adalah kumpulan data-data skalar yang dinyatakan dalam bentuk baris, kolom dan gabungan antar keduanya. Kumpulan data dengan deret yang tidak teratur mengharuskan pemakai untuk menuliskan data satu per satu. Kumpulan data dengan deret yang teratur dapat diekspresikan dalam bentuk array, sehingga memungkinkan pemakai untuk tidak menuliskannya satu per satu.

*Matriks* adalah *array* yang dibangun dari kumpulan persamaan linier. Operasi matrik tidak seperti *array* biasa, melainkan system operasi aljabar matriks. 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.

#### 4.2 Pembentukan *array* dan matriks

*Array* dan matriks ditampilkan dalam bentuk yang sama tetapi representasi internalnya berbeda. Berikut ini adalah perintah untuk membentuk *array*.

- `x=m:n`, membuat baris dengan elemen awal  $m$ , kenaikan 1 dan elemen akhir  $n$
- `x=m:k:n`, membuat baris dengan elemen awal  $m$ , kenaikan  $k$  dan elemen akhir  $n$
- `x=linspace(m,n,k)`, membuat baris dengan elemen awal  $m$  dan elemen akhir  $n$  dengan jumlah elemen sebanyak  $k$
- `x=logspace(m,n,k)`, membuat baris dengan elemen awal  $m$  dan elemen akhir  $n$  dengan jumlah elemen sebanyak  $k$  dalam skala logaritma.
- `x=ones(m)`, membuat *array* segiempat ukuran  $m \times m$  dengan semua elemennya bernilai 1
- `x=ones(m,n)`, membuat *array* segiempat ukuran  $m \times n$  dengan semua elemennya bernilai 1

#### **PRAKTIKUM 4A**

1. Bukalah program Matlab pada komputer anda !
2. Ketikkan sintak perintah-perintah berikut ini dengan huruf kecil diikuti dengan menekan tombol ENTER !
3. Amati hasil tampilan di layar untuk setiap sintak perintah !
4. Tuliskan fungsi dari setiap sintak perintah tersebut pada kolom yang disediakan pada tabel praktikum !

No.	Sintak	Keterangan
1	<code>fix(3.84)</code>	
2	<code>floor(5.32)</code>	
3	<code>ceil(5.32)</code>	
4	<code>x=1:2:20</code>	
5	<code>y=rand(2,5)</code>	
6	<code>y=randn(2,5)</code>	
7	<code>A=[1:3;4:6]</code>	
8	<code>B=[0:2;6:8]</code>	
9	<code>A+B</code>	
10	<code>A*B</code>	
11	<code>A(2,1)</code>	
12	<code>B(:,2)</code>	
13	<code>A(:)</code>	
14	<code>magic(3)</code>	
15	<code>A'</code>	

## **PRAKTIKUM 4B**

1. Bukalah M-File baru dengan menu *File-New-M-File*
2. Ketikkan program sederhana ini dan simpan dengan nama “praktikum4B” !
3. Jalankan program dengan menu *Tools-Run*, jika disimpan di folder pribadi ubah directory dengan menu *File-Set Path-Browse* pada jendela kerja (*command window*).
4. Amati hasil program, jika ada pesan kesalahan lihat pesan di jendela kerja dan perbaiki program di *M-File*, jangan lupa simpan dan jalankan lagi.

```
% program membuat matriks 5 x 5
% By : JUHARI (nama 'JUHARI' bisa diganti)

'diketahui matriks A'
A1=1:5;A2=6:10;A3=11:15;A4=16:20;A5=21:25
Aij=[A1 A2 A3 A4 A5]
'tekan ENTER untuk melihat matriks Bij'
pause
Bij=Aij'
'tekan ENTER untuk melihat matriks Cij=Aij+Bij'
pause
Cij=Aij+Bij
```

```
% program membuat matriks beraturan
% By : JUHARI (nama 'JUHARI' bisa diganti)

A1=[1:15]';
A2=[1:2:30]';
A3=[30:-2:1]';
A4=[A2+A3];
A5=[ A2-A3];
A6=[A2.*A3];
A7=[ A2.*A2];
A8=[ A3.*A3];
A=[A1 A2 A3 A4 A5 A6 A7 A8]
```



## PRAKTIKUM 4C

1. Buatlah satu M-file program utama untuk menghitung nilai dari kedua fungsi tersebut dari data yang diberikan !

$$z = \sqrt{\frac{x^2}{y^2} + \sin x} \quad \text{bila } x \in [0,5] \text{ dan } y \in [5,10]$$

$$A = \begin{bmatrix} 0 & 12 & -42 & -143 \\ 111 & 10 & 22 & 6 \\ 21 & 75 & 22 & 23 \\ 24 & 13 & 22 & -10 \end{bmatrix} \quad B = \begin{bmatrix} 12 & 22 & 24 & -163 \\ 4 & -10 & 24 & 13 \\ 5 & 11 & 2 & -4 \\ 8 & 40 & 32 & -43 \end{bmatrix} \quad b = \begin{bmatrix} 12 \\ -2 \\ 9 \\ 11 \end{bmatrix}$$

Tentukan nilai dari  $2\mathbf{A}\mathbf{B}$ ,  $\mathbf{A}+\mathbf{B}$ ,  $\mathbf{A}b$ , dan  $x$  bila  $\mathbf{A}x=b$

2. Simpan M-file program anda dengan nama “praktikum2C” !
3. Jalankan program anda !

```
%latihan program computer Matlab praktikum 4C
```

[illegible]

## MODUL 5

### CONTROL FLOW

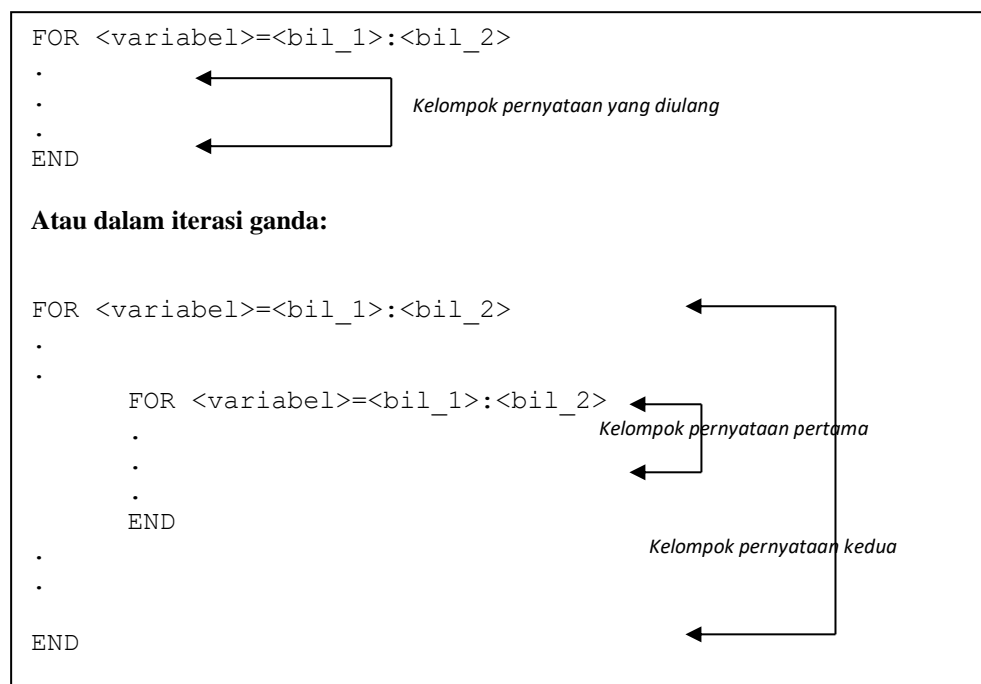
**Tujuan Khusus Praktikum :** Mahasiswa dapat menggunakan pernyataan iteratif untuk membuat program dalam Matlab

#### 5.1 Pernyataan Iteratif

Pernyataan iteratif digunakan untuk menghitung kalkulasi berulang suatu permasalahan. Beberapa permasalahan kadangkala muncul dengan tingkat akurasi tertentu. Untuk mencapai tingkat akurasi ini satu kali kalkulasi dipandang belum cukup akurat sehingga dibutuhkan banyak kalkulasi. Dalam hal ini akan dibahas beberapa pernyataan iterasi yang sering digunakan dalam Matlab.

##### 5.1.1 Iterasi FOR ... END

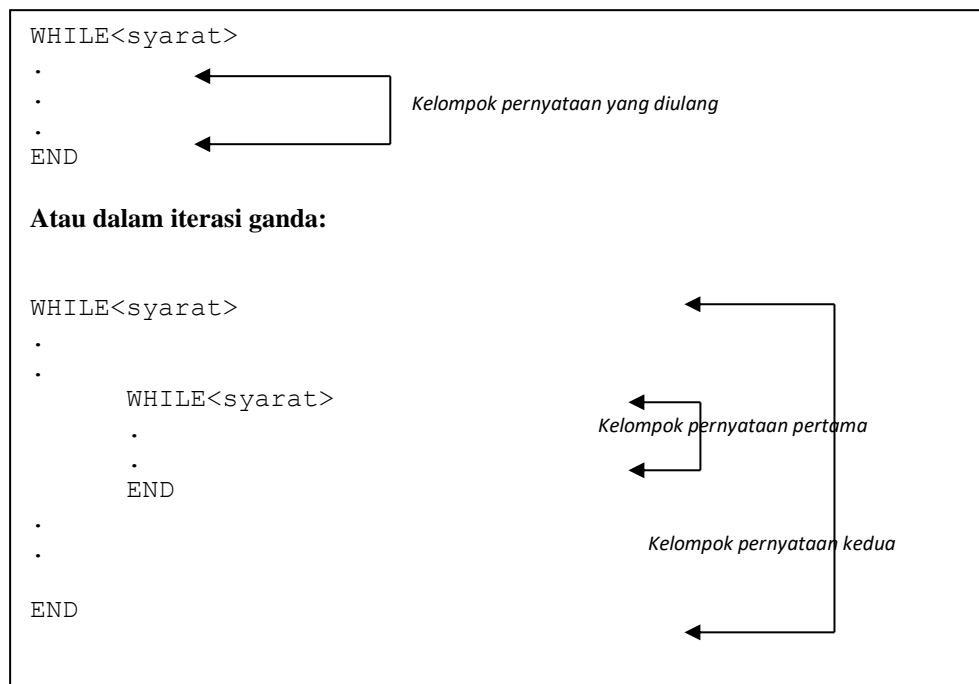
Bentuk umum pernyataan ini dapat ditulis dalam iterasi tunggal atau iterasi ganda. Artinya untuk iterasi tunggal pernyataan yang diulang hanya terdiri dari satu kelompok sedangkan iterasi ganda pernyataan yang diulang terdiri dari beberapa kelompok.



Gambar 5.1 Bentuk umum pernyataan FOR...END

### 5.1.2 Iterasi WHILE ... END

Sebagaimana pernyataan FOR...END bentuk umum pernyataan ini juga dapat ditulis dalam iterasi tunggal dan ganda. Kelebihan penggunaan pernyataan ini adalah adanya suatu prasyarat khusus terhadap jalannya program. Dengan demikian program akan mengalami “self process” dengan merujuk kedalam prasyarat yang telah ditetapkan. Kapan proses berulang itu berhenti dan kapan pula telah mencapai hasil yang optimal, prasyarat akan menjadi signifikan keberadaannya.



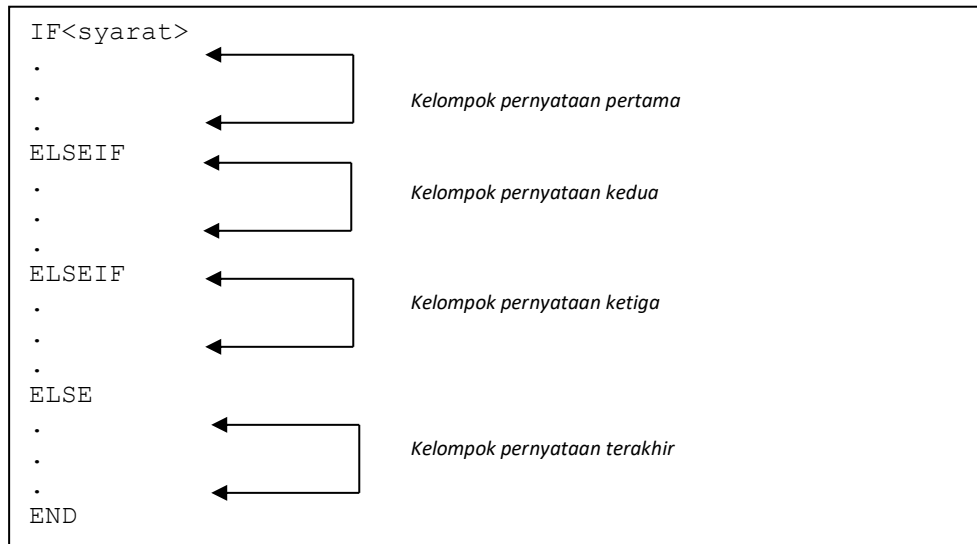
Gambar 5.1 Bentuk umum pernyataan WHILE...END

### 5.1.3 Iterasi IF ... ELSEIF ... ELSE ... END

Berbeda dengan kedua iterasi diatas, iterasi ini digunakan untuk memberikan syarat-syarat khusus terhadap sekelompok pernyataan. Syarat-syarat khusus itu dikemukakan dengan dasar bahwa ada sesuatu yang berlaku pada sekelompok pernyataan tertentu namun tidak berlaku untuk sekelompok pernyataan lainnya, ada sesuatu yang dapat digunakan oleh sekelompok pernyataan tertentu namun tidak untuk sekelompok pernyataan lainnya. Disinilah programmer harus membatasi dengan beberapa syarat.

Syarat-syarat yang ditulis dapat berupa ekspresi *string*, misal “IF BUDI==LULUS” atau ekspresi numeris, misal “IF A=<B” kemudian diikuti dengan

sekelompok pernyataan yang harus menggunakan informasi pertama. Kalau syarat itu dipenuhi maka sekelompok pernyataan itu harus dilalui, namun bila tidak pernyataan itu harus dilewati untuk mengerjakan pernyataan berikutnya. Bentuk umum iterasi ini digambarkan dalam Gambar 5.3.



Gambar 5.1 Bentuk umum pernyataan IF...ELSEIF...ELSE...END

## **PRAKTIKUM 5A**

1. Bukalah M-File baru dengan menu *File-New-M-File*
2. Ketikkan program sederhana ini dan simpan dengan nama “praktikum1B” !
3. Jalankan program dengan menu *Tools-Run*, jika disimpan di folder pribadi ubah directory dengan menu *File-Set Path-Browse* pada jendela kerja (*command window*).
4. Amati hasil program, jika ada pesan kesalahan lihat pesan di jendela kerja dan perbaiki program di *M-File*, jangan lupa simpan dan jalankan lagi.

No.	Program	Keterangan
1	<pre>Clc for i=1:10     i     x=sin(i*pi/180) end x</pre>	
2	<pre>Clc for i=1:10     i     x(i)=sin(i*pi/180) end x</pre>	
3	<pre>Clc for i=1:10 x=sin(i*pi/180)</pre>	
4	<pre>Clc for i=1:10     for j=1:5         z(I,j)=i^2+j^2     end end z</pre>	
5	<pre>Clc A=input('A='); B=rem(A,2); switch B case 1     disp('ganjil') case 2     disp('genap') end</pre>	



No.	Program	Keterangan
6	<pre> Clc A=input('A='); B=rem(A,2); if B==1     disp('ganjil') else     disp('genap') end </pre>	
7	<pre> hitungan=0; x=1:     while (1+x)&gt;1         x=x/2         hitungan=hitungan+1     end </pre>	

## **PRAKTIKUM 5B**

5. Bukalah M-File baru dengan menu *File-New-M-File*
6. Ketikkan program sederhana ini dan simpan dengan nama “praktikum1B” !
7. Jalankan program dengan menu *Tools-Run*, jika disimpan di folder pribadi ubah directory dengan menu *File-Set Path-Browse* pada jendela kerja (*command window*).
8. Amati hasil program, jika ada pesan kesalahan lihat pesan di jendela kerja dan perbaiki program di *M-File*, jangan lupa simpan dan jalankan lagi.

```
% Menghitung penjumlahan matriks
clear all
clc

A=[3 8 5; 6 4 7]; % inisialisasi matrik A
C=[9 5 3; 7 2 1]; % inisialisasi matrik B

% ---proses penjumlahan matrik---
for i=1:2
    for j=1:3
        D(i,j)=A(i,j)+C(i,j);
    end
end

% ---menampilkan matrik A, C dan D---
A
C
D
```

```
% menentukan nilai terkecil dari N bila B=A^N

A=input('Masukkan bilangan asli= ')
N=1;
B=0;
While B<2150050500
    B=A;
    Hit=1;
    While hit<N
        B=B*A;
        Hit=hit*1;
    End
    N=N+1;
End

fprintf('untuk harga a = %2.0f\n',A)
fprintf('harga n adalah = %2.0f\n',N-1)
fprintf('harga B adalah = %2.0f\n',B)
```

## PRAKTIKUM 5C

1. Buatlah satu *M-file* program untuk menghitung dan menampilkan nilai fungsi dari masukan input  $x$  untuk setiap fungsi yang didefinisikan sebagai berikut:
  - a.  $f(x) = \sin 0.5\pi x + 3 \cos \pi x$
  - b.  $f(x) = \sin^2 \pi x - \cos \pi x^2$
  - c.  $h(x) = \frac{6}{x^2} + \sqrt{3x} - \sqrt[5]{x}$
  - d.  $f(x) = \begin{cases} |4x|, & x < 0 \\ 4x, & x > 0 \end{cases}$
  - e.  $f(x) = \begin{cases} 2x, & x < -3 \\ x + 2, & -3 \leq x \leq 3 \\ x^2, & x \geq 3 \end{cases}$
2. Simpan M-file program anda dengan nama “praktikum 5C” !
3. Jalankan program anda !

```
%latihan program computer Matlab praktikum 5C
```

This image shows a full page of white paper with horizontal dotted lines. The lines are evenly spaced and run across the width of the page, providing a guide for handwriting practice. There are no margins, text, or other markings on the page.

## MODUL 6

### MENG GAMBAR GRAFIK

**Tujuan Khusus Praktikum :** Mahasiswa dapat menjalankan fungsi-fungsi Matlab untuk menyelesaikan masalah visualisasi

#### 6.1 Perintah Dasar Menggambar Grafik

Program Matlab mempunyai fasilitas menggambar hasil perhitungan komputasi secara grafis. Elemen dasar yang dibutuhkan untuk menggambar grafik adalah data dalam bentuk *array*. Misal Matlab diberi perintah *array* dalam bentuk sudut theta dari 0 sampai 2 sebanyak 30 data. Dalam Matlab, *array* tersebut dinyatakan dengan variabel *Theta* untuk menentukan nilai  $x = \sin \theta$ , persoalan ini dapat dibuat dalam Matlab dengan *listing* program:

```
theta=linspace(0,2*pi,30)
x=sin(theta)
```

program ini dapat dilengkapi dengan *syntax* untuk menggambar grafik fungsi sinus. Instruksi yang boleh digunakan adalah:

- `Plot(Theta)` : menggambar grafik  $\theta$  terhadap indeksinya
- `Plot(Theta,x)` : menggambar grafik  $x$  terhadap  $\theta$  dengan syarat panjang *array* data  $\theta$  dan  $x$  sama.
- `Plot(Theta,x,string)` : menggambar grafik  $x$  terhadap  $\theta$  dengan karakter berupa *string*. Karakter *string* boleh berupa: **warna, tipe titik data, tipe garis penghubung titik data** dari grafik yang di *plot*.

#### 6.2 Atribut Grafik : title, label, legend, grid and scale

Grafik dapat diberi judul (*titles*). Perintah untuk member judul dan label adalah:

- `title('text','fontsize','size','fontname','name')`
- `xlabel('text','fontsize','size','fontname','name')`
- `ylabel('text','fontsize','size','fontname','name')`
- `zlabel('text','fontsize','size','fontname','name')`
- `grid on` (garis-garis skala)
- `axis` (skala sumbu x)

keterangan :

'text' : nama judul dan label,

'fontsize' : perintah pengaturan huruf,

'size' : ukuran huruf

'fontname' : jenis huruf

'name' : nama huruf

### 6.3 Tipe Garis, Warna dan Penandaan Kurva

Berikut ini adalah tabel simbol-simbol untuk tipe garis dan warna kurva:

Tabel 1.1 Karakter *string* pada grafik

Specifier	Line Color	Specifier	Marker
r	red	.	point
g	green	o	circle
b	blue	x	cross
c	cyan	+	plus
m	magenta	*	asterix
y	yellow	s	square
k	black	d	diamond
w	white	v	triangle down
Specifier	Line Style	^	triangle up
-	solid	<	triangle left
--	dashed	>	triangle right
:	dotted	p	pentagram
..	dash-dot	h	hexagram

### 6.4 Grafik Dimensi Dua

Ada beberapa fasilitas penggambaran grafik 2 dimensi, diantaranya:

Perintah	Fungsi
loglog	Berfungsi sama dengan plot, tetapi skala yang digunakan untuk kedua sumbu adalah skala logaritmis
semilogx	Berfungsi sama dengan plot, tetapi skala yang digunakan untuk sumbu x adalah skala logaritmis dan sumbu y adalah skala linier



semilogy	Berfungsi sama dengan plot, tetapi skala yang digunakan untuk sumbu x adalah skala linier dan sumbu y adalah skala logaritmis
area	Berfungsi sama dengan plot, tetapi daerah antara 0 dan y diisi, basis nilai y dapat ditentukan tetapi defaultnya adalah nol
pie	Membuat grafik lingkaran dengan belahan terlepas
bar	Membuat grafik batang dalam 2 dimensi
pareto	Membuat grafik batang dengan urutan menurun terhadap akumulasi nilai grafik
stairs	Membuat grafik tangga
hist	Membuat grafik histogram
histfit	Membuat grafik histogram dengan perbandingan kurva distribusi normal
stem	Membuat grafik garis menggantung
polar	Membuat grafik koordinat polar
compass	Membuat grafik data kompleks yang menampilkan sudut dan besarnya elemen-elemen kompleks sebagai anak panah yang berasal dari pusat koordinat
feather	Membuat grafik data kompleks yang menampilkan sudut dan besarnya elemen-elemen kompleks sebagai anak panah yang berasal dari titik kesamaan tempat pada garis mendatar
rose	Membuat grafik histogram polar 20 bagian menggunakan sudut-sudut dalam vector
ginput	Memilih titik-titik dari grafik aktif dengan bantuan mouse
fplot	Membuat grafik secara otomatis dari suatu fungsi dengan satu variabel diantara limit yang ditentukan tanpa harus mendefinisikan data untuk variabel tersebut

## 6.5 Grafik Dimensi Tiga

Matlab menyediakan berbagai fungsi untuk menampilkan data secara tiga dimensi. Beberapa fungsi menggambar garis dalam tiga dimensi, sedang yang lain menggambar permukaan dan menempatkan bingkai. Warna dapat dipergunakan untuk menghadirkan dimensi keempat.

Anda dapat membuat grafik atau plot dalam tiga dimensi dengan menggunakan sintak perintah:

Perintah	Fungsi
Plot3	Membuat grafik secara tiga dimensi dari data tiga satuan.
Meshgrid	Menggambarkan poin-poin data teratur dibidang x-y
Mesh	Menggambarkan data nilai fungsi dengan dua variabel dari meshgrid dalam bidang xyz dengan permukaan seperti jala
Meshc	Sama dengan perintah mesh tetapi dengan menambahkan countour dibawahnya
Meshz	Sama dengan perintah mesh tetapi dengan menambahkan grafik tirai atau bidang referensi
Waterfall	Sama dengan perintah mesh kecuali garis-garis jala hanya tampak dari arah sumbu x
Surf	Mengisi ruang pada permukaan grafik jala tiga dimensi dengan aturan yang sama dengan perintah mesh
Surfc	Sama dengan perintah surf tetapi dengan menambahkan countour dibawahnya
Surfl	Sama dengan perintah surf tetapi dengan menambahkan pengaturan tampilan permukaan berdasarkan suatu sumber cahaya
contour3	Menggambarkan grafik kontur garis-garis yang menunjukkan ketinggian yang tetap dalam 3D

Pcolor	Memetakan ketinggian dengan sudut himpunan warna dan memberikan informasi yang sama dengan yang dihasilkan oleh kontur dengan skala yang sama.
Ribbon	Sama dengan perintah plot, hanya saja kolom y digambarkan dengan tinta terpisah oleh 3D
fill3	Versi tiga dimensi dari perintah fill, yaitu menggambar polygon tiga dimensi dalam ruang 3D

## **PRAKTIKUM 6A**

1. Bukalah M-File baru dengan menu *File-New-M-File*
2. Ketikkan program sederhana ini dan simpan dengan nama “praktikum6A” !
3. Jalankan program dengan menu *Tools-Run*, jika disimpan di folder pribadi ubah directory dengan menu *File-Set Path-Browse* pada jendela kerja (*command window*).
4. Amati hasil program, jika ada pesan kesalahan lihat pesan di jendela kerja dan perbaiki program di *M-File*, jangan lupa simpan dan jalankan lagi.

No.	Program	Output
1	<pre> t=[0:360]; x=cos(t) y=cos(t*pi/180); z= sin(t*pi/180);  figure(1) plot(t,x,t,y,t,z) legend('x','y','z')  figure(2) plot(t,y,'c',t,z,'m') grid on title('grafik fungsi') xlabel('sudut t') ylabel('nilai fungsi') legend('cos(t)','sin(t)')  figure(3) plot(t,y,'-',t,z,'-') grid on title('grafik fungsi') xlabel('sudut t') ylabel('nilai fungsi') legend('cos(t)','sin(t)') </pre>	
2	<pre> A=peaks(25) B=[.5 1 1.6 1.2 .8 2.1] figure(1); subplot(2,2,1); plot(A) title('grafik peaks') subplot(2,2,2); pie(B,B==max(B)) title('grafik pie') subplot(2,2,3); pareto(B) title('grafik parito') subplot(2,2,4); pareto(B) title('grafik parito') </pre>	

	<pre>figure(2); subplot(2,2,1); bar(exp(-A.*A)) title('grafik bar') subplot(2,2,2); stairs(exp(-A.*A)) title('grafik tangga') subplot(2,2,3); barh(exp(-A.*A)) title('bar mendatar') subplot(2,2,4); hist(exp(-A.*A)) title('histogram')</pre>	
3	<pre>figure(1) x=rand(50,1) stem(x, '.'); title('Grafik stem dari data acak')  figure(2) t=linspace(0.2*pi) r=sin(2*t).*cos(2*t) polar(t,r) title('Grafik polar sin(2t)cos(2t)')  figure(3) y=eig(randn(20,20)) compass(y) title('Grafik compass dari nilai eigen matriks acak')  figure(4) fplot('sin(x)./x, [-20 20 -.4 1.2]') title('Grafik Fungsi grid on')</pre>	
4	<pre>figure [x,y]=meshgrid(-2:.2:2, -1:.15:1); z=x.*exp(-x.^2-y.^2); subplot(2,2,1) surf(z) xlabel('x') ylabel('y') zlabel('z') colormap('bone')  subplot(2,2,2) surf(z)</pre>	

	<pre> xlabel('x') ylabel('y') zlabel('z')  subplot(2,2,3) contour(z) xlabel('x') ylabel('y') zlabel('z')  subplot(2,2,4) surf(z) xlabel('x') ylabel('y') zlabel('z') shading('flat') </pre>	
5	<pre> [x,y,z]=peaks figure(1); contour(x,y,z,20) title('Grafik kontur dari fungsi peaks')  figure(2); contour(x,y,z,20) axis([-3 3 -3 3 -6 8]) title('Grafik kontur 3D dari fungsi peaks')  figure(3); pcolor(x,y,z) title('Grafik pseudocolor dari fungsi peaks')  figure(4); pcolor(x,y,z) shading interp hold on contour(x,y,z,20,'k') hold off title('Grafik pseudocolor dengan contour fungsi peaks')  figure(4); surf(x,y,z,atan(x,y)) colormap(hsv) shading flat axis([-3 3 -3 3 -6.5 8.1]) axis off title('surf one color') </pre>	

### **PRAKTIKUM 6B**

1. Buatlah satu *M-file* program untuk menampilkan grafik fungsi berikut:

$$f(x) = \sin^2 x + 3 \cos 4x$$

$$g(x) = \sin x \cos x - \cos x^2$$

Dengan interval  $[-a\pi, a\pi]$  dengan  $a$  dari masukan input dalam satu figure yang memuat empat gambar, yaitu:

- a. Grafik f dengan garis putus-putus
  - b. Grafik g dengan warna hijau
  - c. Grafik f garis titik-titik dan g garis lurus
  - d. Grafik  $f+g$
2. Berikan atribut pelengkap (grid, title, label, dan legend) untuk masing-masing gambar
  3. Simpan M-file program anda dengan nama “praktikum6B”
  4. Jalankan program anda !

```
%latihan program computer Matlab praktikum 6B
```

[illegible]

## MODUL 7

### ANALISIS DATA

**Tujuan Khusus Praktikum :** Mahasiswa dapat menjalankan fungsi-fungsi Matlab untuk menyelesaikan masalah yang berkaitan dengan analisis data

#### 7.1 Analisis Statistik

Dalam bab ini, kita akan belajar bagaimana menganalisis dan memanipulasi data menggunakan MATLAB, terutama untuk perhitungan statistik: rentang data, maksimum/minimum, rata-rata, deviasi, jumlah kumulatif, dan sebagainya. Di MATLAB fungsi-fungsi statistik semacam ini telah ada dan bisa digunakan secara fleksibel. Dalam penjelasan bab ini, **x** dan **y** kita misalkan sebagai vector (baris ataupun kolom), dan **A** dan **B** sebagai matriks  $m \times n$ .

Untuk menganalisa data digunakan perintah *mean* yang menghasilkan nilai rata-rata, *std* untuk nilai standart deviasi, *max* untuk nilai maksimum dan *min* untuk nilai minimum. Untuk fungsi-fungsi analisis data lainnya terdapat pada tabel berikut:

Fungsi	Keterangan
<code>corrcoef(x)</code>	Koefisien korelasi
<code>cov(x)</code>	Matriks kovarian
<code>cplxpair</code>	Mensortir vector menjadi pasangan-pasangan kompleks konjugat
<code>cross(x, y)</code>	Vector cross product
<code>cumprod(x)</code>	Cumulative product
<code>cumprod(x, n)</code>	Cumulative product dengan dimensi n
<code>cumsum(x)</code>	Cumulative summation
<code>cumsum(x, n)</code>	Cumulative summation dengan dimensi n
<code>cumtrapz(x, y)</code>	Integrasi trapezoid kumulatif
<code>cumtrapz(x, y, n)</code>	Integrasi trapezoid kumulatif dengan dimensi n
<code>diff(x)</code>	Perbedaan (selisih antara elemen satu dengan elemen sesudahnya)
<code>diff(x, m)</code>	Perbedaan (selisih antara elemen satu dengan elemen sesudahnya) dengan derajat m
<code>dot(x, y)</code>	Vector dot product



<code>gradient (f, dx, dy)</code>	Pendekatan gradient fungsi $f(x,y)$
<code>std(x)</code>	Standart deviasi
<code>randn(x)</code>	Membangkitkan bilangan random distribusi normal
<code>trapz(x, y)</code>	Integrasi trapezoid dari $y=f(x)$
<code>mean(x)</code>	Nilai rata-rata
<code>mean(x, n)</code>	Nilai rata-rata dengan n dimensi
<code>median(x)</code>	Nilai tengah
<code>median(x, n)</code>	Nilai tengah dengan n dimensi

## 7.2 Pencocokan Kurva (Regresi)

Pencocokan kurva secara kuadrat terkecil merupakan regresi yang akan meminimalkan jumlah kesalahan kuadrat pada setiap point data dari kurva yang digunakan. Dalam Matlab, fungsi *polyfit* menyelesaikan masalah pencocokan kurva atau regresi dengan metode kuadrat terkecil. Perintah ini memerlukan tiga input, yaitu variabel bebas (x), variabel tak bebas (y), dan derajat atau order polynomial (n). Perintah ini akan menghasilkan vector baris koefisien-koefisien untuk polynomial.

## **PRAKTIKUM 7A**

1. Bukalah M-File baru dengan menu *File-New-M-File*
2. Ketikkan program sederhana ini dan simpan dengan nama “praktikum6A” !
3. Jalankan program dengan menu *Tools-Run*, jika disimpan di folder pribadi ubah directory dengan menu *File-Set Path-Browse* pada jendela kerja (*command window*).
4. Amati hasil program, jika ada pesan kesalahan lihat pesan di jendela kerja dan perbaiki program di *M-File*, jangan lupa simpan dan jalankan lagi.
5. Tuliskan fungsi dari setiap sintak program pada kolom yang disediakan pada tabel praktikum !

No	Program	Output
1	<pre>clc;clear; x=[0:.1:1]; y=[-.447 1.976 3.25 6.16 7.05 5.26 7.66 9.56 9.48 9.30 11.2] p1=polyfit(x,y,1) p2= polyfit(x,y,2) x1=linspace(0,1,100) f1=polyval(p1,x1) f2=polyval(p2,x1) figure(1) plot(x,y,'-o',x1,f1,'.') title('interpolasi linier') grid on  figure(2) plot(x,y,'-o',x1,f2,'.') title('interpolasi kuadrat') grid on  figure(3)</pre>	

	<pre> x1=linspace(0.2*pi,60) x2=linspace(0.2*pi,6) plot(x1,sin(x1),x2,sin(x2),'---') title('interpolasi linier') grid on disp('interpolasi linier untuk sin(pi):') interp1(x2,sin(x2),pi,'linier') </pre>	
2	<pre> clc;clear; suhu=randn(31,3); d=1:31; figure(1) plot(d,suhu) xlabel('hari dalam bulan') title('suhu tertinggi di tiga kota') rata2=mean(suhu) rerata2=mean(rata2) ratakolom=mean(suhu,1) ratabaris=mean(suhu,2) </pre>	

## PRAKTIKUM 7B

1. Buatlah satu *M-file* program untuk menampilkan analisis statistic dengan data acak normal berukuran 50 observasi:
  - a. Nilai rata-rata
  - b. Nilai tengah
  - c. Nilai maksimum
  - d. Nilai minimum
  - e. Standart deviasi
  - f. Grafik histogram
  - g. Grafik plot scatter (plot titik sebaran)
  - h. Grafik plot garis
2. Berikan atribut pelengkap (grid, title, label, dan legend) untuk masing-masing gambar
3. Simpan M-file program anda dengan nama “praktikum6B”
4. Jalankan program anda !

```
%latihan program computer Matlab praktikum 7B
```

This image shows a full page of white paper with horizontal dotted lines. The lines are evenly spaced and run across the width of the page, providing a guide for handwriting practice. There are no margins, text, or other markings on the page.

## MODUL 8

### GUIDE MATLAB

**Tujuan Khusus Praktikum :** *Mahasiswa dapat membangun aplikasi GUI untuk menyelesaikan perhitungan numerik*

#### 8.1 Pendahuluan

GUIDE atau GUI builder merupakan sebuah *graphical user interface* (GUI) yang dibangun dengan obyek grafik seperti tombol (*button*), kotak teks, slider, menu dan lain-lain. Aplikasi yang menggunakan GUI umumnya lebih mudah dipelajari dan digunakan karena orang yang menjalankannya tidak perlu mengetahui perintah yang ada dan bagaimana kerjanya.

Sampai saat ini, jika kita membicarakan pemrograman berorientasi visual, yang ada di benak kita adalah sederetan bahasa pemrograman, seperti visual basic, Delphi, visual C++, visual Fox Pro, dan lainnya yang memang didesai secara khusus untuk itu. Matlab merintis ke arah pemrograman yang menggunakan GUI dimulai dari versi 5, yang terus disempurnakan sampai sekarang (Matlab 7).

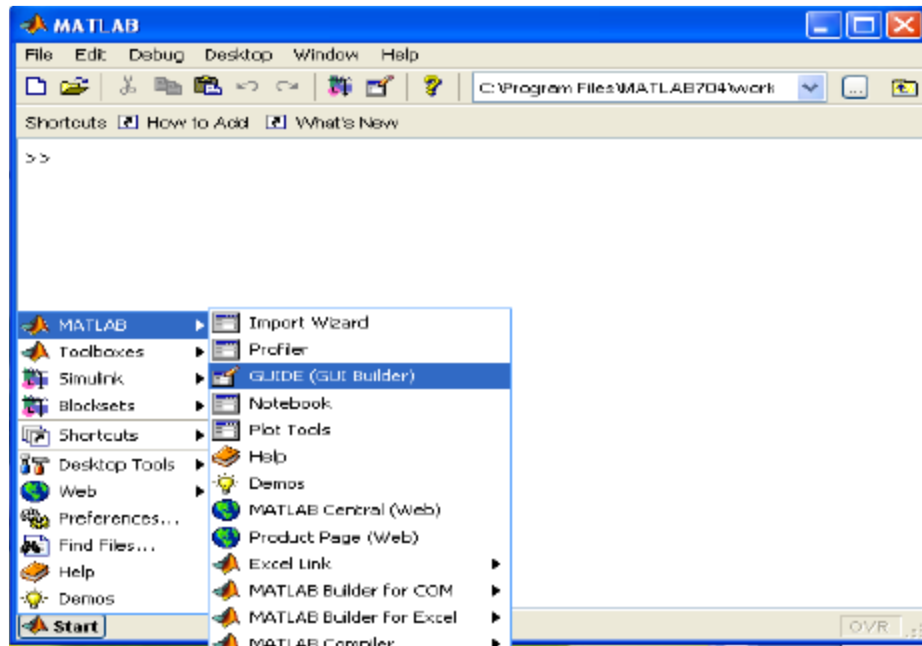
GUIDE Matlab mempunyai kelebihan tersendiri dibandingkan dengan bahasa pemrogram lainnya, diantaranya:

1. GUIDE Matlab banyak digunakan dan cocok untuk aplikasi-aplikasi berorientasi sains, sehingga banyak peneliti dan mahasiswa menggunakan GUIDE Matlab untuk menyelesaikan riset atau tugas akhirnya.
2. GUIDE Matlab mempunyai fungsi built-in yang siap digunakan dan pemakai tidak perlu repot membuatnya sendiri.
3. Ukuran file, baik FIG-file maupun M-file, yang dihasilkan relatif kecil.
4. Kemampuan grafisnya cukup andal dan tidak kalah dibandingkan dengan bahasa pemrograman lainnya.

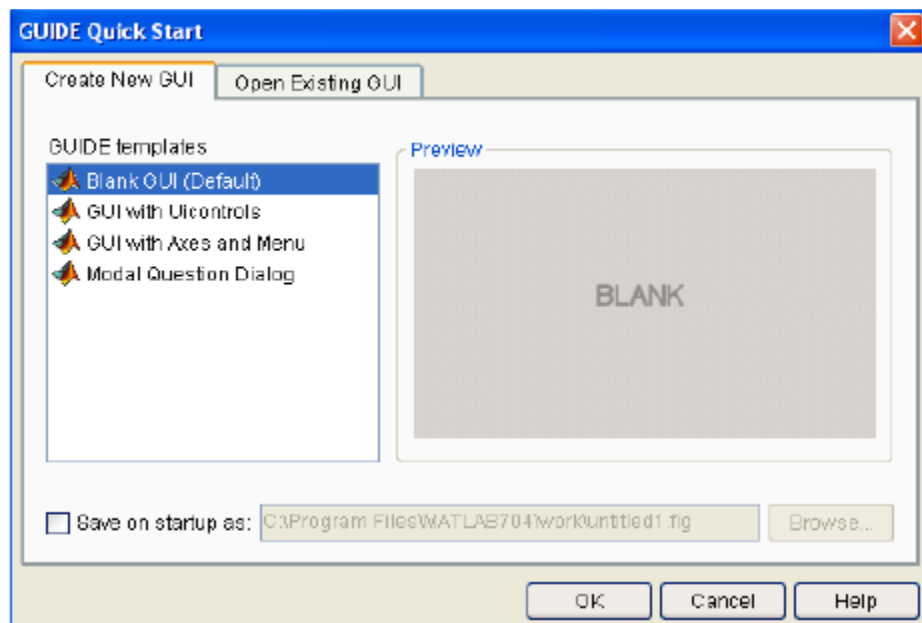
#### 8.2 Memulai GUIDE MATLAB

Memulai GUIDE Matlab dapat dilakukan dengan dua cara, yaitu:

- 1) Melalui command matlab dengan mengetikkan: >> guide
- 2) Klik tombol Start Matlab dan pilihlah MATLAB, lalu pilih GUIDE (GUI Bulder)



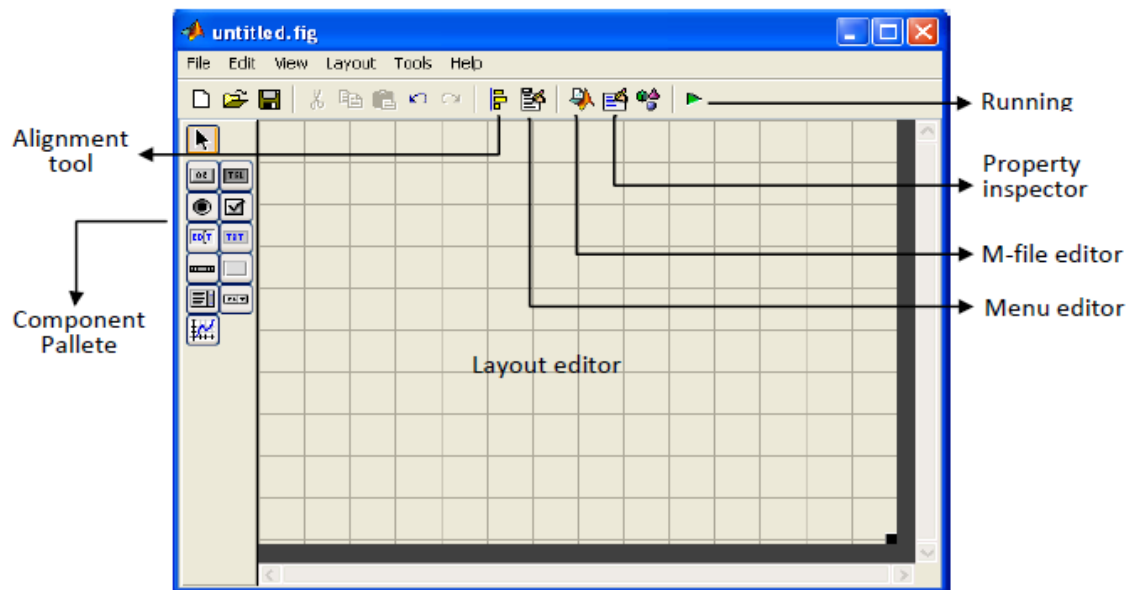
Gambar 8.1 Memulai GUIDE MATLAB



Gambar 8.2 GUIDE Quick Start

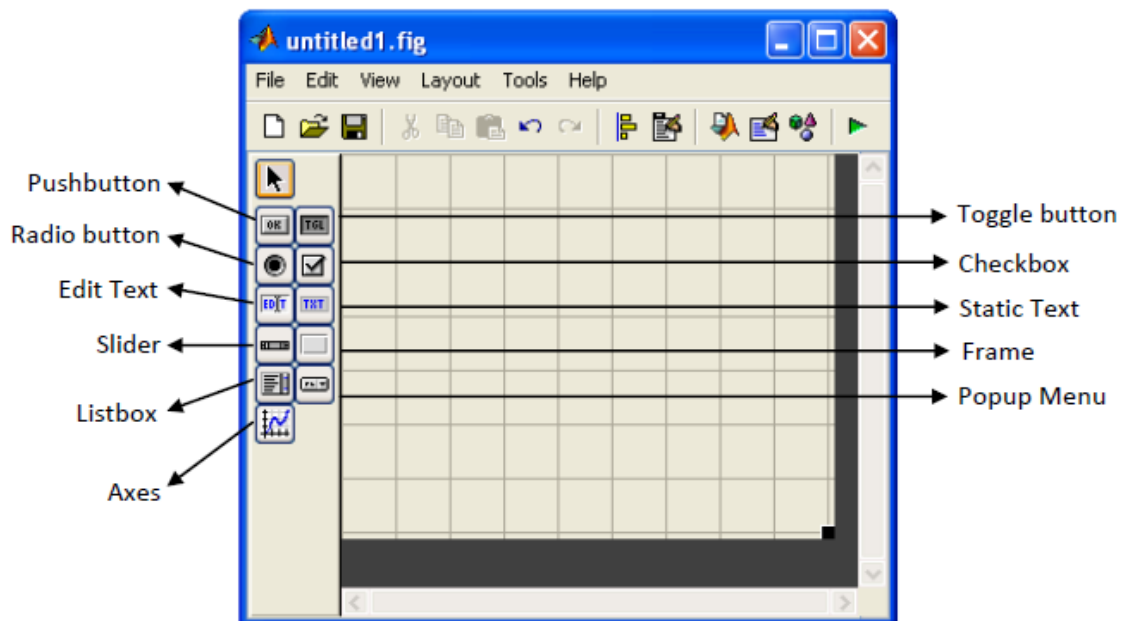
Pada pilihan Create New GUI terdapat menu GUIDE templates yang memiliki beberapa tipe dasar dari GUI, sehingga kita dapat melakukan modifikasi pada template agar menjadi GUI seperti yang kita harapkan. Sebagai pemula, kita gunakan Blank GUI (Default) yang merupakan sebuah GUI dengan figure kosong dan merupakan kondisi default dari GUIDE dan dipilih jika kita memang akan membuat sebuah aplikasi dengan

komponen yang layout-nya tidak terdapat pada GUI template yang lain. Setelah kita memilih Blank GUI templates, maka akan muncul tampilan Menu Utama GUIDE.



Gambar 8.3 Tampilan GUIDE

Komponen palet pada GUIDE Matlab terdiri dari beberapa uicontrol (kontrol user interface), seperti pada bahasa pemrograman visual lainnya, yaitu: pushbutton, togglebutton, radiobutton, checkboxes, edit text, static text, slider, frames, listboxes, popup menu, dan axes. Kita dapat meletakkan semua kontrol pada layout editor dan selanjutnya hanya tinggal mengaturnya melalui property inspector.



Gambar 8.4 Komponen GUIDE

## **PRAKTIKUM 8A**

1. Buat GUIDE untuk praktikum dalam membangkitkan sinyal !