

## MATERI 12

### FUNGSI DAN INTEGRAL

Beberapa fungsi '**Fungsi**' dalam MATLAB disajikan dalam tabel berikut:

Category	Function	Description
Plotting	fplot	Untuk membuat plot fungsi
Optimization and zero finding	fminbnd	Mencari nilai minimum fungsi satu variabel.
	fminsearch	Mencari nilai minimum fungsi beberapa variabel
	fzero	Mencari nilai nol dari fungsi satu variabel
Numerical Integration	quad	Menghitung integral numerik, adaptive Simpson quadrature.
	quadl	Menghitung integral numerik, adaptive Lobatto quadrature.
	dblquad	Menghitung integral ganda

#### PENYAJIAN FUNGSI (FUNCTIONS) DALAM MATLAB

MATLAB dapat menyajikan fungsi matematika dengan menggunakan **inline** dan **M-files**.

##### a. Sebagai obyek inline

Sebagai **obyek inline**, sudah kita lakukan beberapa waktu yang lalu.

Contoh:

$$f(x) = \frac{1}{(x-0.3)^2 + 0.01} + \frac{1}{(x-0.9)^2 + 0.04} - 6$$

Dengan menggunakan inline, fungsi tersebut didefinisikan sbb:

```
f = inline('1./((x-0.3).^2 + 0.01) + 1./((x-0.9).^2 + 0.04)-6');
```

Mengevaluasi f di 2.0

```
>>f(2.0)
```

```
ans =
```

```
-4.8552
```

Dengan inline, anda juga dapat membuat fungsi yang lebih dari 1 argumen dengan menpesifikasikan nama-nama argumennya. Contoh, fungsi berikut mempunyai 2 buah argument input, yaitu x dan y:

```
>> f = inline('y*sin(x)+x*cos(y)', 'x', 'y')
```

```
>> f(pi, 2*pi)
```

```
ans =
```

```
3.1416
```

##### b. Sebagai Fungsi MATLAB

Fungsi ini dibuat melalui menu **File → New → Script**.

Kemudian tulis fungsi di bawah ini di layer MATLAB editor.

```
function y = fsi(x) %nama fungsinya fsi
y = 1./((x-0.3).^2 + 0.01) + 1./((x-0.9).^2 + 0.04) - 6;
```

Simpan dengan nama **fsi.m**, caranya pilih menu **File → save As → beri nama fsi.m → save**

#### Catatan penting:

- ♪ Nama file harus sama dengan nama fungsi
- ♪ Jangan menggunakan nama fungsi dengan nama-nama fungsi bawaan yang sudah ada dalam MATLAB, misal **sin**, **log**, **exp**, **polyfit**, **fzero**, **fmin**, dll.

Hal tersebut akan merusak fungsi bawaan MATLAB, dan akan merugikan anda sendiri! (pada saat dipanggil, akan muncul pesan 'Attempt to execute SCRIPT as a function')

- ♪ Gunakan operasi elemen (menggunakan titik untuk perkalian dan pembagian, pangkat)
- ♪ Sebaiknya akhiri definisi rumus dengan tanda ; (titik koma) agar hasilnya tidak ditampilkan.

Mengevaluasi fungsi **fsi.m** pada 2.0

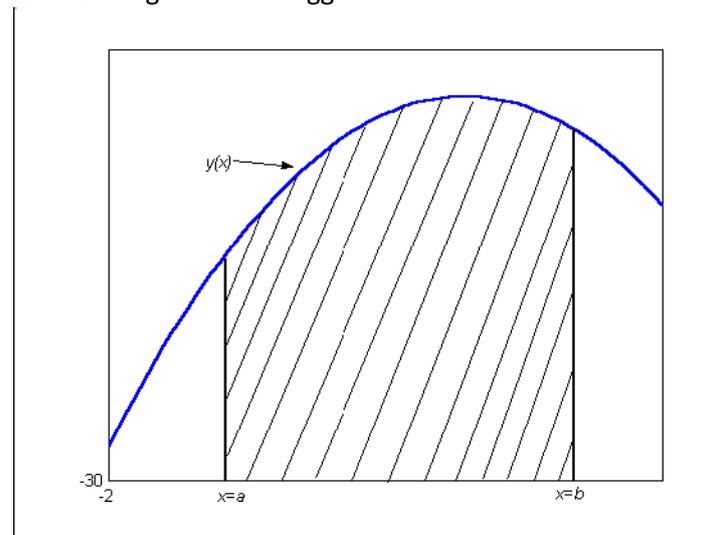
Cara 1	Cara 2
<pre>fh = @fsi; feval(fh,2.0) ans = -4.8552</pre>	<pre>feval('fsi',2.0) ans = -4.8552</pre>

Catatan:

untuk menggunakan fungsi yang didefinisikan dengan menggunakan M-File, maka nama fungsi harus diapit dengan tanda petik ' '. Lihat cara 2 di atas. Sedangkan yang didefinisikan dengan inline, langsung dipanggil nama fungsinya tanpa diapit tanda ' '.

## INTEGRASI NUMERIK

Integral fungsi  $f(x)$  dalam ranah  $a \leq x \leq b$  dapat diinterpretasikan sebagai luas daerah dibawah kurva  $f(x)$  yang membentang dari  $x=a$  hingga  $x=b$ .



Dalam hal ini  $f(x)$  disebut integrand,  $x=a$  disebut batas bawah integral dan  $x=b$  disebut batas atas integral sedangkan  $x$  adalah variabel integral.

## MENGHITUNG INTEGRAL DENGAN METODE NUMERIK

Integral terbatas bisa diselesaikan secara numerik dengan MATLAB, yaitu:

$$q = \int_a^b f(x) dx$$

Terdapat sejumlah metode perhitungan integral secara numerik, misalkan: trapezoid, kuadratur, dll.

<b>trapz(x,y)</b>	menghitung integral dari <b>y</b> sebagai fungsi dari <b>x</b> . Vektor <b>x</b> dan <b>y</b> panjangnya harus sama. Nilai elemen dalam <b>x</b> sebaiknya disortir
<b>trapz(x,A)</b>	menghitung integral dari setiap kolom di <b>A</b> sebagai fungsi dari <b>x</b> ; hasilnya berupa vektor baris berisi hasil integrasi. Jumlah kolom <b>A</b> harus sama dengan panjang <b>x</b> .
<b>quad('fcn',a,b)</b>	menghitung aproksimasi dari integral fungsi <b>fcn</b> pada interval $a \leq x \leq b$ . Fungsi <b>fcn</b> harus didefinisikan terlebih dahulu dalam M-file.
<b>quad('fcn',a,b,tol)</b>	menghitung aproksimasi integral dari <b>fcn</b> dengan toleransi kesalahan sebesar <b>tol</b> .
<b>quad('fcn',a,b,tol,trace,pic)</b>	menghitung aproksimasi integral dari <b>fcn</b> dengan toleransi <b>tol</b> . Jika <b>trace</b> tidak nol, maka grafik yang mengilustrasikan integral akan diplot. Hasil integrasi dievaluasi pada <b>pic</b> . bisa diberi nilai nol pada <b>tol</b> dan <b>trace</b> dengan matriks kosong [ ].
<b>quadl( ... )</b>	sama dengan <i>command</i> <b>quad</b> , tetapi menghitung dengan akurasi yang lebih tinggi.

Sebagai contoh, kita hitung integral berikut ini dengan metode numerik:

$$\int_0^2 e^{-x^3} dx$$

```
>> x = linspace(0,2,50); % definisikan vektor x
>> y = exp(-x.^3); % hitung nilai y
>> integral = trapz(x,y) % integralkan !
    integral =
    0.8821
```

Dengan *command* **quad**,

#### CARA 1 :

terlebih dahulu harus mendefinisikan fungsi dalam M-file:

```
function y = myfun(x)
y = exp(-x.^3);
```

Kita hitung integral tersebut dengan toleransi yang berbeda:

```
>> format long; % format bilangan "long"
>> int_1 = quad('myfun',0,2,0.001), ...
    int_2 = quad('myfun',0,2,0.00001)
    int_1 =
    0.89309707589214
    int_2 =
    0.89295225387894
```

Kita bandingkan akurasi dengan **quadl**:

```
>> int_3 = quadl('myfun',0,2)
    int_3 =
    0.89295351461757
>> format short; % mengembalikan format ke "short"
```

Ini adalah hasil paling akurat yang bisa diperoleh MATLAB.

Jika tidak menggunakan fungsi dalam M-File, integral diatas dapat diselesaikan dengan cara :

## CARA 2 :

Dengan menuliskan langsung fungsinya diantara tanda petik ('), Tanpa pendefinisian fungsi dalam M-File

```
>> int_y=quad(' exp(-x.^3) ',0,2,00001)
int_y =
    0.8931
```

## CARA 3 :

Dengan pendefinisian fungsi memakai inline

```
>>Y=inline(' exp(-x.^3) ')
>> int_y=quad(Y,0,2,00001)
```

## INTEGRAL LIPAT-2

Kita bisa menghitung integral terbatas lipat-2 dengan menyelesaikan integralnya satu per satu menggunakan *command quad*.

Misalkan kita ingin menghitung integral berikut ini:

$$\int_0^1 \int_0^1 e^{-x^3-y} dy dx$$

Pertama, kita buat M-file untuk fungsi ini:

```
function z = fungsiku(x,y)
z = exp(-x.^3-y);
```

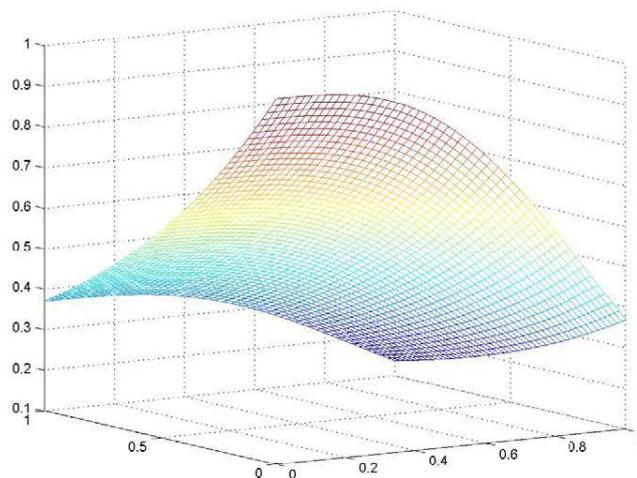
**dblquad('fcn',xmin,xmax,ymin,ymax,tol)** : menghitung integral lipat-2 untuk fungsi dua variabel **fcn** pada area segiempat  $xmin \leq x \leq xmax$ ,  $ymin \leq y \leq ymax$ .

Untuk contoh integral di atas:

```
>> Integral_dobel = dblquad('fungsiku',0,1,0,1)
Integral_dobel =
    0.5104
```

Untuk mendapatkan gambaran dari fungsi tersebut, kita ketikkan:

```
>> [X,Y] = meshgrid(x,x);
>> Z = fungsiku(X,Y);
>> mesh(X,Y,Z)
```



### INTEGRAL LIPAT-3

Integral lipat-3 bisa diselesaikan setahap demi setahap. Misalkan untuk integral berikut ini disimpan dalam M-file:

$$\int_{-2}^2 \int_{-2}^2 \int_{-2}^2 \sqrt{x^2 + y^2 + z^2} dz dy dx$$

Untuk mendapatkan nilai integral dari fungsi diatas, buat fungsi

```
function w = funxyz(x,y,z)
w = sqrt(x.^2 + y.^2 + z.^2);
```

**triplequad('fcn',xmin,xmax,ymin,ymax,zmin,zmax,tol)** , menghitung integral lipat-3 untuk fungsi tiga variabel **fcn** pada area balok  $xmin \leq x \leq xmax$ ,  $ymin \leq y \leq ymax$ ,  $zmin \leq z \leq zmax$ , dengan toleransi kesalahan sebesar **tol**.

Untuk contoh integral di atas, kita hitung dengan toleransi 0,001:

```
>> Integral_tripel = triplequad('funxyz', -2,2,-2,2,-2,2,0.001)
Integral_tripel =
122.9577
```

### MENGHITUNG PANJANG KURVA

Anda dapat menggunakan quad atau quadl untuk menghitung panjang kurva. Misal diketahui kurva parameter sbb:

$$x(t) = \sin(2t), \quad y(t) = \cos(t) \quad z(t) = t$$

where  $t \in [0, 3\pi]$

Grafik plot 3 kurva tsb adalah sbb:

```
t = 0:0.1:3*pi;
plot3(sin(2*t),cos(t),t)
```

Rumus panjang busur menyebutkan bahwa panjang kurva adalah integral dari akar kuadrat turunan persamaan paramaternya:

Fungsi dengan nama **hcurve.m**

```
function f = hcurve(t)
f = sqrt(4*cos(2*t).^2 + sin(t).^2 + 1);
```

Integral fungsi tsb dicari dengan cara:

```
len = quad(@hcurve,0,3*pi)
len =
1.7222e+01
```

Maka panjang kurva tsb adalah 17.2.