



Introduction to Matlab



Matlab의 특징

- ▣ 행렬 데이터가 기본 연산
- ▣ 그래픽기능 강화
- ▣ M-file을 이용한 프로그래밍

1.Format

- Format short-----소수점이하 4자리
- Format e-----지수형식으로 표시
- Format long-----소수점 14자리
- Format compact-----소수점 4자리
- Format rad-----분수로 표시
- Format bank-----소수점 2자리
- Format hex-----16진법으로 표시
- Vpa-----원하는 소수점이하 자리를
구함

1.Format

```
MATLAB Command Window
File Edit Window Help
[Icons]
 format short
 pi
ans =
    3.1416
 format short e
 pi
ans =
    3.1416e+000
 format long
 pi
ans =
    3.14159265358979
 format compact
 pi
ans =
    3.14159265358979
 format bank
 pi
ans =
    3.14
 format hex
 pi
ans =
    400921fb54442d18
 vpa(pi,20)
ans =
    3.1415926535897932385

```

1) 스칼라 계산

```
MATLAB Command Window
File Edit Window Help
[Icons]
[] x=23;
[] y=14;
[] x+y
ans =
    37
[] x-y
ans =
     9
[] x*y
ans =
   322
[] x/y
ans =
   1.6429
[] x^2+y^2
ans =
   725
[] z=x^2+2*x-y
z =
   561
[]
```

2) 행렬 계산

$$X = \begin{bmatrix} 1 & 2 \\ 4 & 5 \end{bmatrix}, \quad Y = \begin{bmatrix} 2 & 3 \\ 5 & 6 \end{bmatrix}$$

```
MATLAB Command Window
File Edit Window Help
x=[1,2;4,5.]
x =
     1     2
     4     5
y=[2 3;5 6 ]
y =
     2     3
     5     6
x+y
ans =
     3     5
     9    11
x-y
ans =
    -1    -1
    -1    -1
x^2
ans =
     9    12
    24    33
```

```
MATLAB Command Window
File Edit Window Help
inv(x)
ans =
   -1.6667    0.6667
    1.3333   -0.3333
x.^2
ans =
     1     4
    16    25
x*y
ans =
    12    15
    33    42
x/y
ans =
    1.3333   -0.3333
    0.3333    0.6667
x.*y
ans =
     2     6
    20    30
|
```

3) 벡터 계산

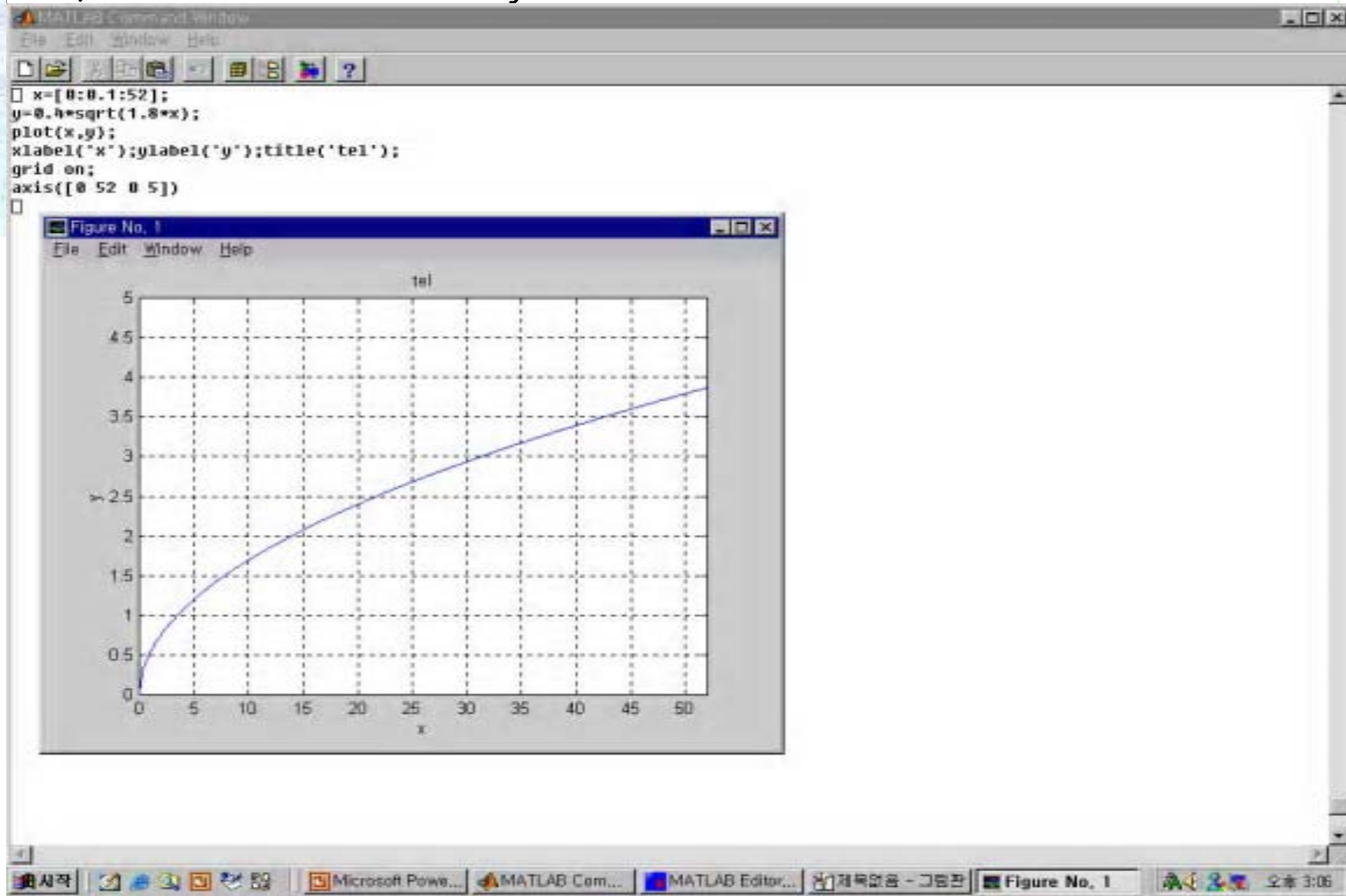
```
MATLAB Command Window
File Edit Window Help
[Icons]
[] a=[1 2 3];b=[4 5 6];
[] dot(a,b)
ans =
    32
[] sum(a.*b)
ans =
    32
[] cross(a,b)
ans =
    -3     6    -3
[] x=0:0.1:1;
[] y=x^3-3*x^2-0.2*x
??? Error using ==> ^
Matrix must be square.
[] x=0:0.1:1;
[] y=x.^3-3*x.^2-0.2*x
y =
Columns 1 through 7
     0    -0.0490   -0.1520   -0.3030   -0.4960   -0.7250   -0.9840
Columns 8 through 11
 -1.2670  -1.5680  -1.8810  -2.2000
[]
```

2.graphic

1)plot, label, title, grid, axis 명령어

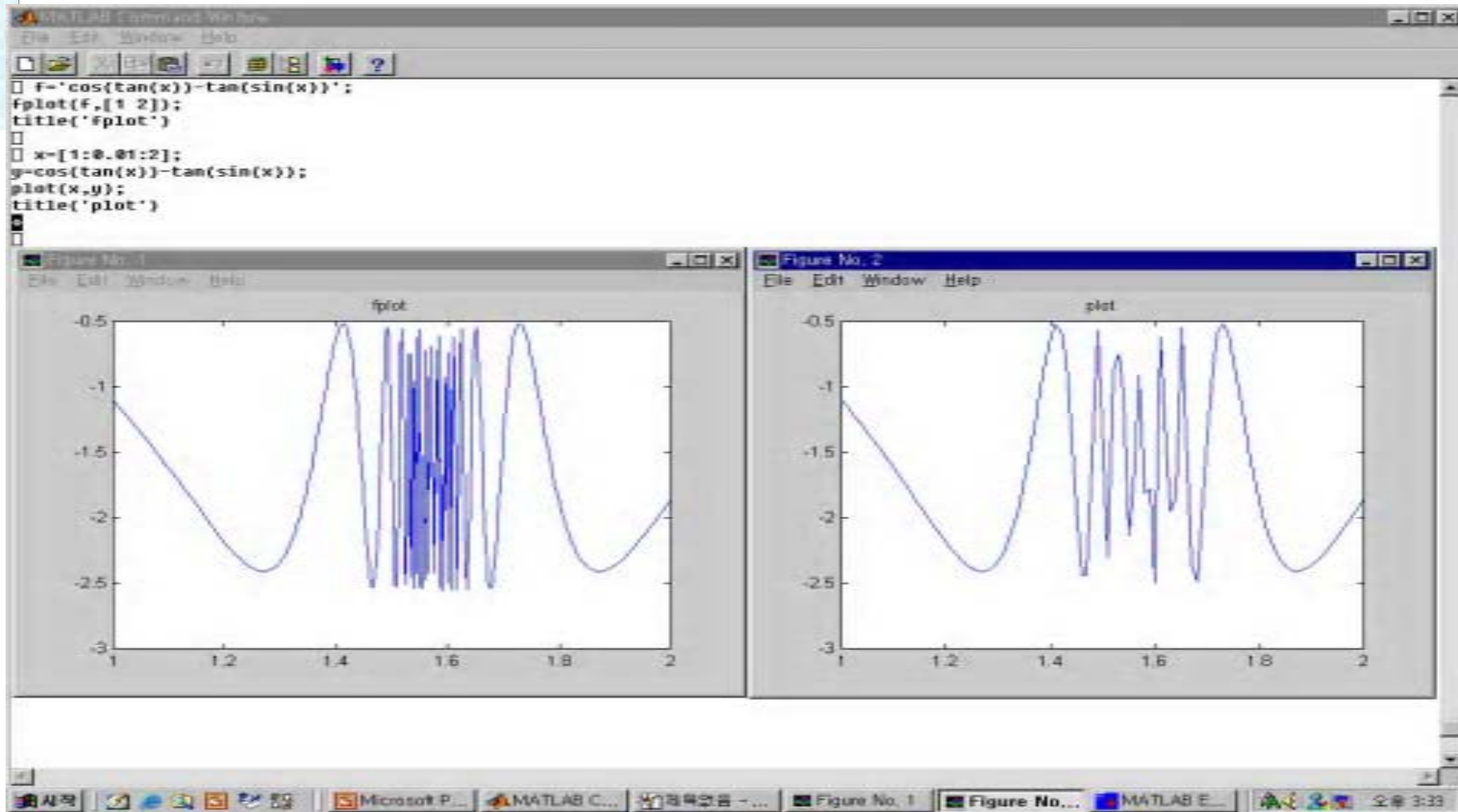
- Plot(x,y)-----y를 x범위에서 그린다.
- Xlabel('x좌표이름')---x좌표이름을 붙인다.
- Ylabel('y좌표이름')---y좌표이름을 붙인다.
- Title('그림이름')-----그림이름을 붙인다.
- Text-----지정된 위치에 text를 붙인다.
- Gtext-----마우스를 사용하여 text를 붙인다.
- Grid on-----grid를 만든다.
- Axis([x1 x2 y1 y2])--- $x_1 \leq x \leq x_2$, $y_1 \leq y \leq y_2$ 의 축을 잡는다.

ex) $0 \leq x \leq 52$ 인 범위의 $y = 0.4\sqrt{1.8x}$ 를 그려라. 단 $\Delta x = 0.1$



2) 함수 그림 명령어 fplot('함수이름',[xmin xmax])

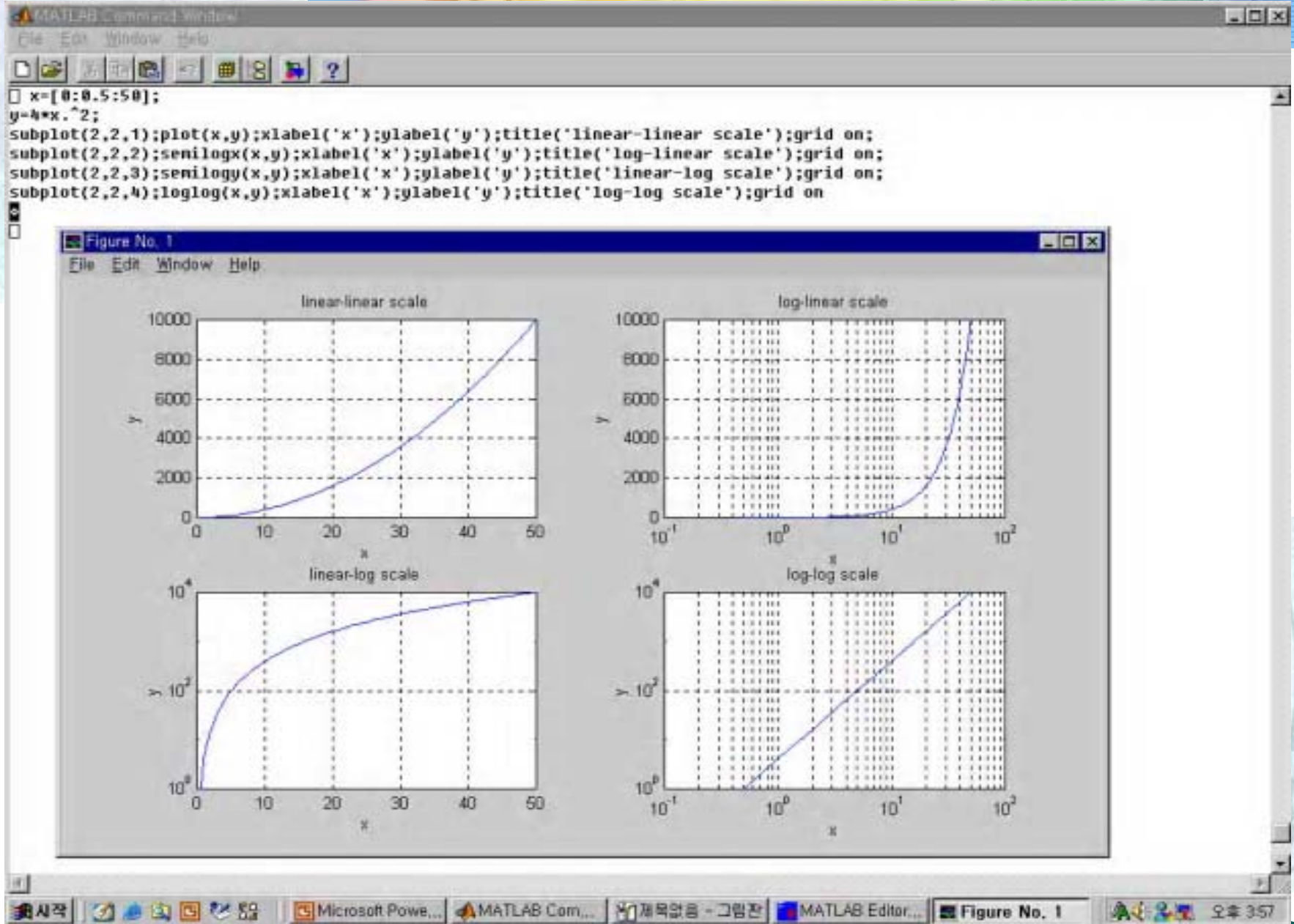
Ex) $f = \cos(\tan x) - \tan(\sin x)$ 의 그림을 범위 $1 \leq x \leq 2$ 에서 fplot와 plot명령어를 사용하여 그려라.



3) subplot와 log명령어

- Subplot(m,n,p)-----m×n개의 작은 방을 만들고 p번째 방에 그림을 그린다.
- Semilogx-----x축만 대수 좌표
- Semilogy-----y축만 대수 좌표
- Loglog-----x,y축 대수 좌표

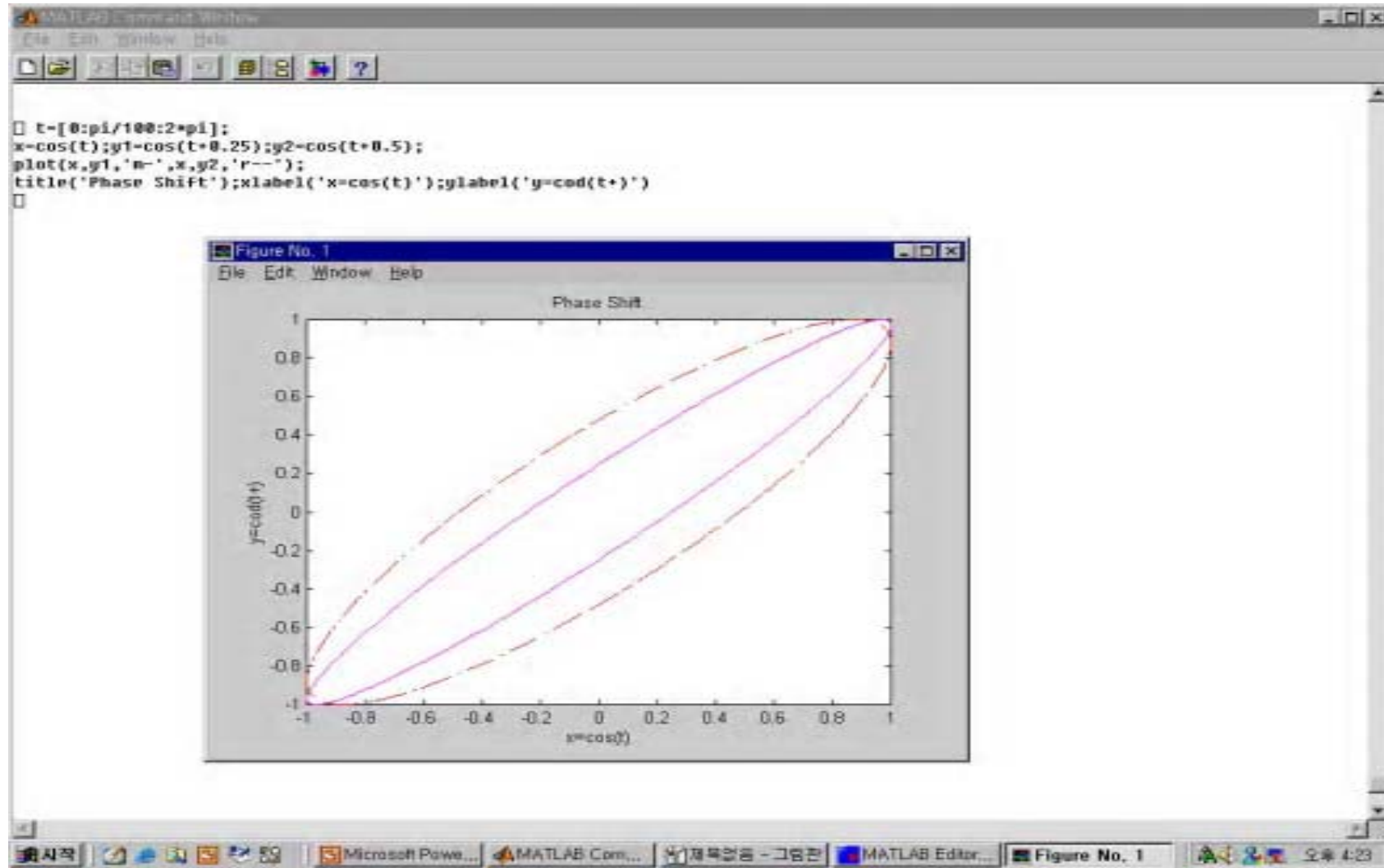
Ex) $y=4x^2$, $0 \leq x \leq 50$, $\Delta x=0.5$ 에 대한 (1)plot(x,y)
(2)semilogx(x,y) (3)semilogy(x,y) (4)loglog(x,y)의 그림
을 subplot를 이용하여 그려라.



4)그림기호, 선 종류 및 색

기호 이름	기호	선 이름	기호	색	기호
Dot	●	Solid	-	황(yellow)	y
Asterisk	*	Dash	--	적자(magenta)	m
Cross	×	Dash Dot	-.	청록(cyan)	c
Circle	○	Dotted	:	적(red)	r
Plus	+			녹(green)	g
Square	□			청(blue)	b
Diamond	◇			백(white)	w
Star	☆			흑(black)	b

Ex) $x = \cos t$, $y_1 = \cos(t+0.25)$, $y_2 = \cos(t+0.5)$ 의 그림을 색을 넣어서 그려라. 단 $0 \leq t \leq 2\pi$, $\Delta t = \pi/1000$ 이고 x좌표는 $x = \cos(t)$, y좌표는 $y = \cos(t+)$ 이고 제목은 Phase Shift이다.



3. 프로그램

Escape 문자

Escape 문자	설 명
\n	New Line
\t	Horizontal tab
\b	Backspace
\r	Carriage return
\f	Form feed
\\	\ 표시
\" or \"	“ 표시
%%	% 표시

변환기호

변환 기호	설 명
%f	실수(예 : 31.5 ⇒ %f4.1)
%c	문자
%s	문자열
%e	지수(예: 1.23e10)
%E	지수(예: 1.23E10)

3) M-file 작성법

M-file은 Matlab언어로 쓰여진 file을 의미하며 다른 언어의 subroutine이나 function과 같은 것을 의미.

M-file 1)function mode—입력 값을 받아서 어떤 연산을 수행하여 출력 값으로 보내주는 함수로써의 기능이 주 목적

2)script mode---명령어 또는 함수들의 일괄처리가 주 목적

Matlab을 실행한 후에 첫번째 Icon “New File”을 클릭하면 “Matlab editor/debugger”text editor가 뜬다. 여기에 M-file을 작성한다.

1)Function mode

형식:

Function[output_1,output_2,,]=f/n_함수(input_1,input_2,,)

2)Script mode



```
r=3;  
vol=nam(r)  
r=3.000 vol=113.0973355  
  
vol =  
  
    113.0973  
  
nn  
  
vol =  
  
    113.0973  
  
[
```

```
function [vol]=nam(r)  
vol=(4/3)*pi*r^3  
fprintf('r=%5.3f vol=%10.7f\n', r,vol)  
|
```

Ready Line 4 8:11 PM

```
r=3;  
vol=(4/3)*pi*r^3
```

Ready Line 1 8:12 PM

1)택일문

1.if 문

If문은 일반 프로그램 언어와 마찬가지로 단순 if(if~end)문 및 블록 if(if~else~end)문이 있다.

2.switch 문

If문의 대안으로 만들어진 것으로서 if문보다 훨씬 읽기와 사용하기가 쉽다.

If 수식1

문장1;

Elseif 수식2

문장2;

Else

문장3;

End

switch 입력변수(스칼라 또는 문자열)

case 값1

문장1

case 값2

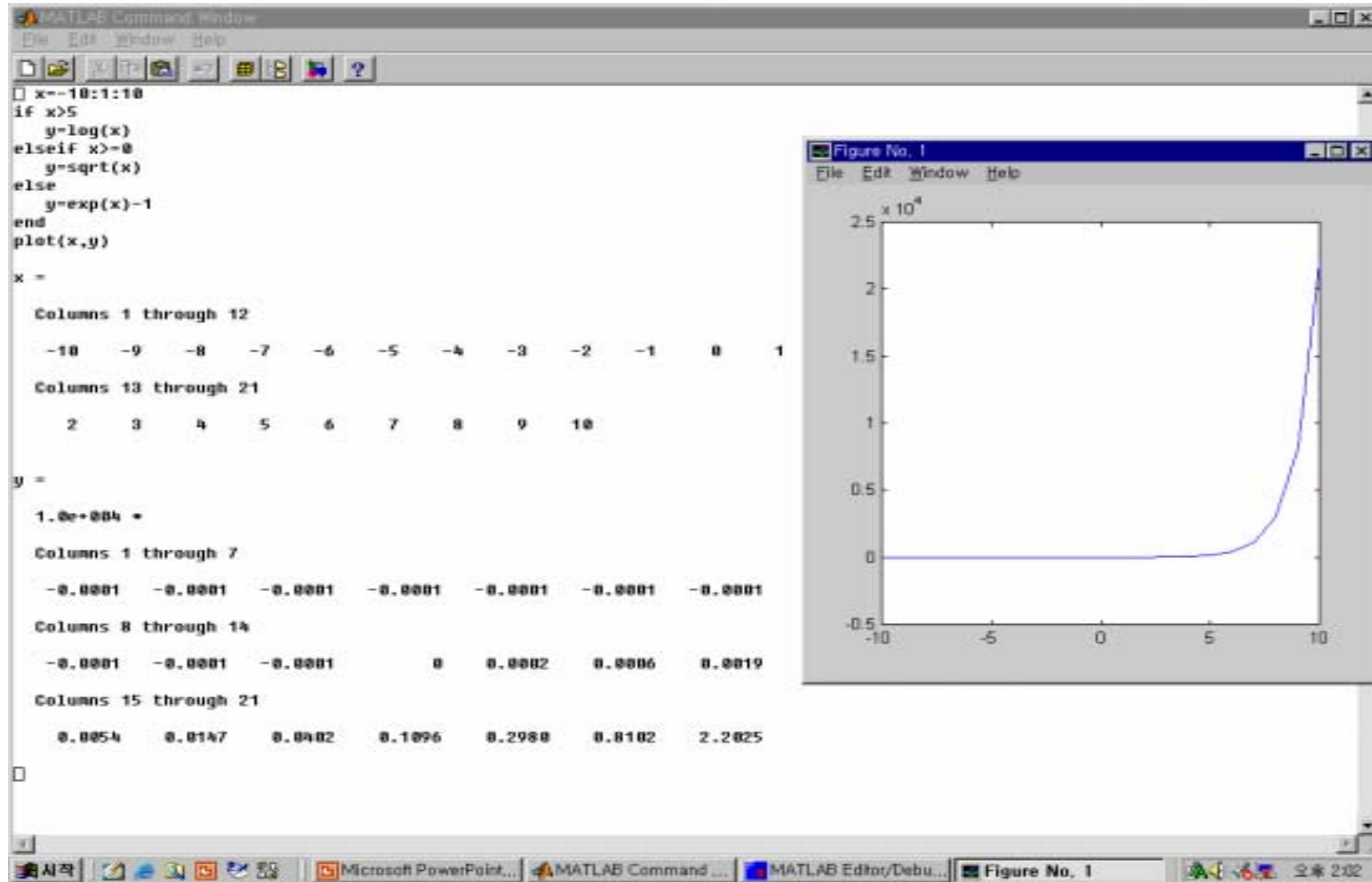
...

otherwise

문장n

end

Ex)-10≤x≤10 범위에서 x>5이면 y=ln(x) , 0≤x≤5이면 y=√(x), x<0이면 y=e^x-1를 계산하여라



Ex) x값이 2이면 $y=x+2$, x값이 3이면 $y=2x$ 값, 나머지는 0의 값을 출력하는 프로그램을 만들어라.

```
MATLAB Command Window
File Edit Window Help
[Icons]
□ x=2;
switch x
case 2,
    y=x+2
case 3,
    y=2*x
otherwise,
    y=0
end

y =

     4

□ x=3;
switch x
case 2,
    y=x+2
case 3,
    y=2*x
otherwise,
    y=0
end

y =

     6

□ x=5;
switch x
case 2,
    y=x+2
case 3,
    y=2*x
otherwise,
    y=0
end

y =
```

```
MATLAB Command Window
File Edit Window Help
[Icons]
0
□ x=1;
switch x
case 2,
    y=x+2
case 3,
    y=2*x
otherwise,
    y=0
end

y =

     0

□ |
```

2)반복문(Loop_)

1.For loop

```
For 반복변수=초기값:증분값:최종값  
    문장  
End
```

2.While 문

While반복문은 특별한 조건을 만족시키면 반복과정을 끝내거나,진행방향에 패스의 번호를 모르는 경우에 사용한다.

Ex) $vol = (4/3) \times \pi r^3$ 의 값을 (1) $r=2$, (2) $r=1:5$ for문 사용,
 (3) $0 \leq r \leq 5$ while문 사용하여 구하여라.

```

MATLAB Command Window
File Edit Window Help
r=2;
vol=(4/3)*pi*r^3

vol =
    33.5103

for r=1:5
    vol=(4/3)*pi*r^3
    disp([r,vol])
end

vol =
    4.1888
    1.0000    4.1888

vol =
    33.5103
    2.0000    33.5103

vol =
    113.0973
    3.0000    113.0973

vol =
    268.0826
    4.0000    268.0826
    
```

```

MATLAB Command Window
File Edit Window Help
vol =
    523.5988

r=-1;
while r<5
    r=r+1
    vol=(4/3)*pi*r^3;
    disp([r,vol]);
end

r =
    0
    0    0

r =
    1
    1.0000    4.1888

r =
    2
    2.0000    33.5103

r =
    3
    3.0000    113.0973
    
```

```

MATLAB Command Window
File Edit Window Help
r =
    4
    4.0000    268.0826

r =
    5
    5.0000    523.5988
    
```

```
Ex) % -----  
    % Generate a temperature conversion table, followed by a graph  
    % Instructor: Nam Sun Wang  
    % -----
```

```
clear all      % Start fresh  
C = [0:100]';  % Generate a column vector of x from 0 to 100  
F = 32 + 1.8 * C; % Convert from Celsius to Fahrenheit  
[C F]         % Display in tabular format  
pause        % Pause for non-Windows operating environment (e.g.,  
DOS)  
plot(C,F)     % Display in graphical format; C in x-axis; F in y-axis  
save 'temp1.dat' C F -ascii % Save data in a file; C before F in 1-  
column format  
combine=[C F]; save 'temp2.dat' combine -ascii % Save data in a file  
in 2-column format
```



```

MATLAB Command Window
File Edit Window Help
C=[0:100];
F=32+1.8*C;
[C,F]
plot(C,F);
title('Tem conversion'),xlabel('C'),ylabel('F')

ans =

Columns 1 through 7
    0    1.0000    2.0000    3.0000    4.0000    5.0000    6.0000

Columns 8 through 14
    7.0000    8.0000    9.0000   10.0000   11.0000   12.0000   13.0000

Columns 15 through 21
   14.0000   15.0000   16.0000   17.0000   18.0000   19.0000   20.0000

Columns 22 through 28
   21.0000   22.0000   23.0000   24.0000   25.0000   26.0000   27.0000

Columns 29 through 35
   28.0000   29.0000   30.0000   31.0000   32.0000   33.0000   34.0000

Columns 36 through 42
   35.0000   36.0000   37.0000   38.0000   39.0000   40.0000   41.0000

Columns 43 through 49
   42.0000   43.0000   44.0000   45.0000   46.0000   47.0000   48.0000

Columns 50 through 56
   49.0000   50.0000   51.0000   52.0000   53.0000   54.0000   55.0000

```

```

MATLAB Command Window
File Edit Window Help

Columns 57 through 63
   56.0000   57.0000   58.0000   59.0000   60.0000   61.0000   62.0000

Columns 64 through 70
   63.0000   64.0000   65.0000   66.0000   67.0000   68.0000   69.0000

Columns 71 through 77
   70.0000   71.0000   72.0000   73.0000   74.0000   75.0000   76.0000

Columns 78 through 84
   77.0000   78.0000   79.0000   80.0000   81.0000   82.0000   83.0000

Columns 85 through 91
   84.0000   85.0000   86.0000   87.0000   88.0000   89.0000   90.0000

Columns 92 through 98
   91.0000   92.0000   93.0000   94.0000   95.0000   96.0000   97.0000

Columns 99 through 105
   98.0000   99.0000  100.0000  102.0000  103.8000  105.6000  107.4000

Columns 106 through 112
  109.2000  111.0000  112.8000  114.6000  116.4000  118.2000  120.0000

Columns 113 through 119
  121.8000  123.6000  125.4000  127.2000  129.0000  130.8000  132.6000

Columns 120 through 126
  134.4000  136.2000  138.0000  139.8000  141.6000  143.4000  145.2000

```



```

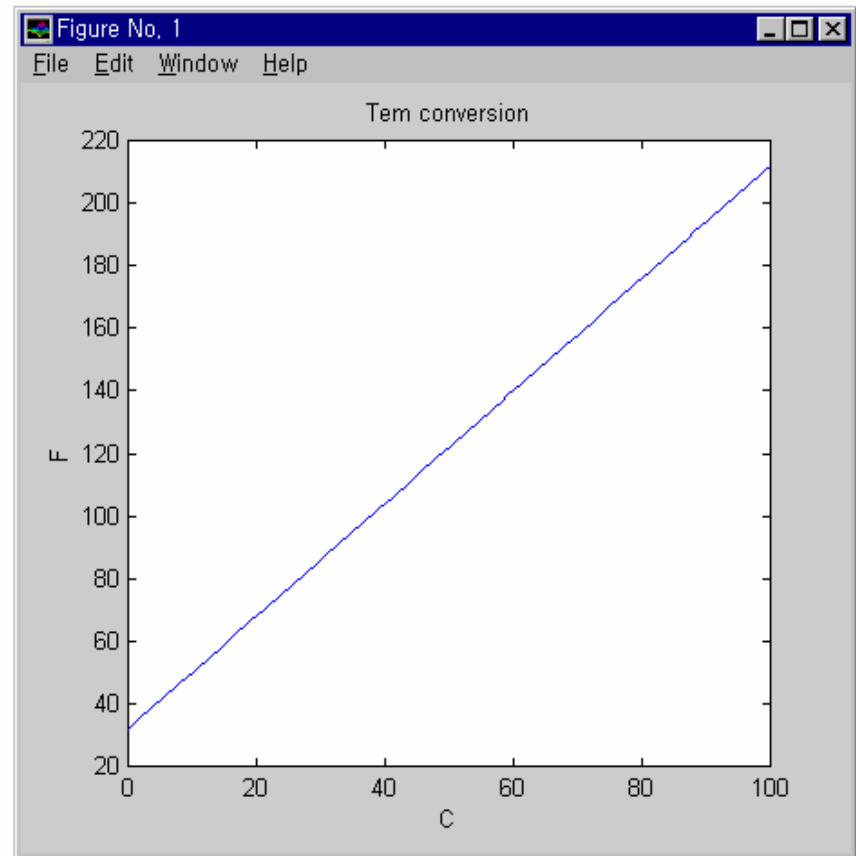
MATLAB Command Window
File Edit Window Help
Columns 127 through 133
77.0000 78.8000 80.6000 82.4000 84.2000 86.0000 87.8000
Columns 134 through 140
89.6000 91.4000 93.2000 95.0000 96.8000 98.6000 100.4000
Columns 141 through 147
102.2000 104.0000 105.8000 107.6000 109.4000 111.2000 113.0000
Columns 148 through 154
114.8000 116.6000 118.4000 120.2000 122.0000 123.8000 125.6000
Columns 155 through 161
127.4000 129.2000 131.0000 132.8000 134.6000 136.4000 138.2000
Columns 162 through 168
140.0000 141.8000 143.6000 145.4000 147.2000 149.0000 150.8000
Columns 169 through 175
152.6000 154.4000 156.2000 158.0000 159.8000 161.6000 163.4000
Columns 176 through 182
165.2000 167.0000 168.8000 170.6000 172.4000 174.2000 176.0000
Columns 183 through 189
177.8000 179.6000 181.4000 183.2000 185.0000 186.8000 188.6000
Columns 190 through 196
190.4000 192.2000 194.0000 195.8000 197.6000 199.4000 201.2000

```

```

MATLAB Command Window
File Edit Window Help
Columns 197 through 202
203.0000 204.8000 206.6000 208.4000 210.2000 212.0000

```



```

Ex)%-----
% Use "inv", " (pre-division), or "^-1", to solve a set of linear algebraic
  equations in MATLAB
% Ax=b
% A minimum hard-coded example; not recommended.
% Instructor: Nam Sun Wang
%-----
% Start fresh -----
  clear all
% Input matrix/vector coefficients from the problem discussed in class
% 3*x1 - x2 + 2*x3 = 12
% x1 + 2*x2 + 3*x3 = 11
% 2*x1 - 2*x2 - x3 = 2
% The coefficients for matrix A are (in row-wise order):
  A = [ 3 -1 2 ; 1 2 3 ; 2 -2 -1 ];
% The coefficients for column vector b are:
  b = [ 12; 11; 2 ];

```

% Solve the linear set of equations by calling the "inv" function -----

```
x = inv(A)*b;
```

% Print results -----

```
disp('The solution with the "inv" function is ...')
```

```
disp(x)
```

% An alternative way by pre-division -----

```
x = A\b;
```

% Print results -----

```
disp('The solution with pre-division is ...')
```

```
disp(x)
```

% Yet another way by raising to -1 power -----

```
x = A^(-1)*b;
```

% Print results -----

```
disp('The solution with ^-1 is ...')
```

```
disp(x)
```

MATLAB Command Window

File Edit Window Help



```
□ a=[3 -1 2;1 2 3;2 -2 -1];  
b=[12;11;2];  
x=inv(a)*b;  
disp('The solution with the "inv" function is...')  
disp(x)
```

```
The solution with the "inv" function is...  
3.0000  
1.0000  
2.0000
```

```
□ x=a\b;  
disp('The solution with pre-division is ...')  
disp(x)
```

```
The solution with pre-division is ...  
3.0000  
1.0000  
2.0000
```

```
□ x=a^(-1)*b;  
disp('The solution with ^-1 is...')  
disp(x)
```

```
The solution with ^-1 is...  
3.0000  
1.0000  
2.0000
```

```
□
```

```
function fact=fact(n)
```

```
% -----  
% Calculate n!  
% n ... the given integer (input)  
% fact ... product of a sequence of numbers (i.e., factorial) (output)  
% -----  
% This function works for scalar input only, not matrices.  
% This is valid only for n<22 (because n! can fill up a real variable fast--In  
% Matlab, numbers are real numbers, not integers).  
% 21!= 51090942171709440000  
% 22!=1124000727777607680000 ...true value  
% 22!=1124000727777607700000 ...value returned by fprintf('%25.0f',  
% fact(22))  
% :  
% For 21<n, the error becomes increasingly severe.  
% Matlab provides the following two functions: factorial(n), prod(1:n)  
% -----  
    fact=1;  
    for i=1: n  
        fact = fact*i;
```

MATLAB Command Window

File Edit Window Help



```
n=4;  
fact=1;  
for i=1:n  
    fact=fact*i  
end
```

fact =

1

fact =

2

fact =


6

fact =

24

MATLAB Editor/Debugger - [C:\MATLAB\bin\fact.m]

File Edit View Debug Window Help



Stack:

```
function y=fact(n)  
fact=1;  
for i=1:n  
    fact=fact*i  
end
```

Ready Line 4 11:42 AM