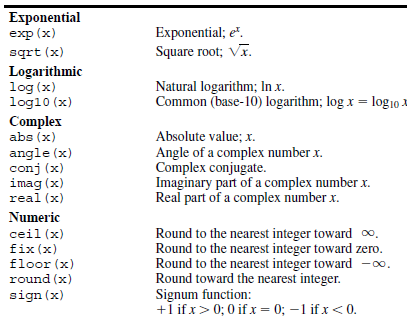
%======================================================== sec. 4.1 some common used build-in functions

% help --> MATLAB --> mathematics --> elementary function

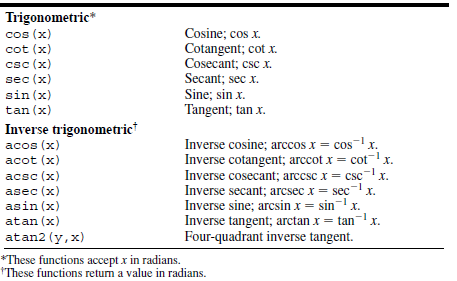
%========================================================Some commonly used elementary function

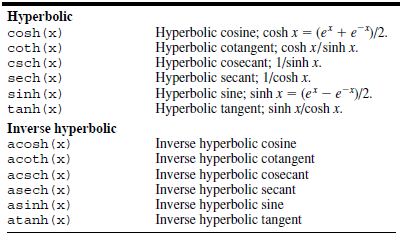


x = -1:.1:1;

figure, plot(x,abs(x),'o')

Trigonometric functions





%% Complex number :

1. Definition of a complex number:
2. Representation with Cartesian coordinate and polar coordinate
3. Translation between these two representation.

% atan2(Y,X) is the four quadrant arctangent of the elements

% of X and Y. -pi <= atan2(Y,X) <= pi.

clear all;

x = [1 -1 -1 1];

y = [1 1 -1 -1];

z=x+1i.\*y;

z2=x(2)+1i\*y(2);

mag\_z2=abs(z2);

ang\_z2=atan2(y(2),x(2)) \* 180/pi;

x2=mag\_z2.\*cosd(ang\_z2);

y2=mag\_z2.\*sind(ang\_z2);

% in radian

x2=abs(z).\*cos(atan2(y,x));

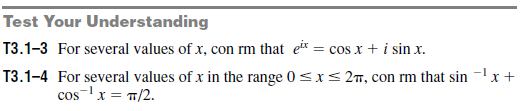
y2=abs(z).\*sin(atan2(y,x));

%% in radians

x2=abs(z).\*cos(atan2(y,x));

y2=abs(z).\*sin(atan2(y,x));

%% Exercise



Use (a) z1=x1+1i\*y1 ; with x1=1 and y1=-2 to translate between Cartesian coordinate and polar coordinate.

(b) add and multiple two complex numbers Z1 and Z2 in polar coordinate

Where z2=x2+1i\*y2 ; with x2=1 and y2=-1

t = clock; % given a variable of current time

fprintf( ' %02.0f:%02.0f:%02.0f\n', t(4), t(5), t(6) );

x=1:2:7;

cumsum(x')

cumsum(1:4)

date

realmax % largest positive floating number on your computer

realmin

rem(19, 5)

% The following statements convert 40 inches this way by using fix and rem commands:

feet = fix(40/12)

inches = rem(40, 12)

fprintf( ' %d feet %d inches \n', feet,inches );

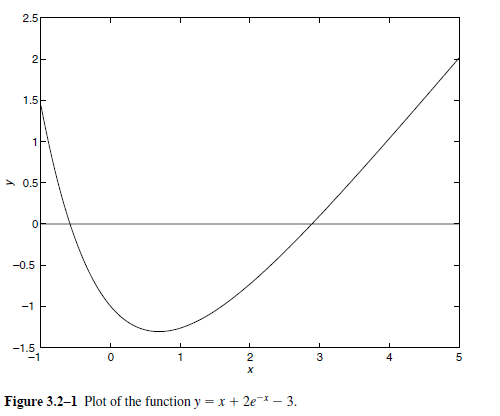
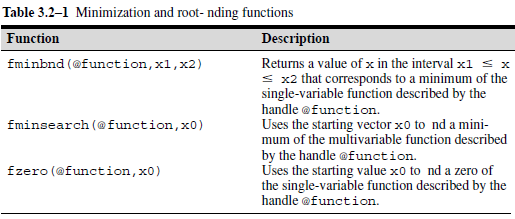
% Exercise 4.6

%% Some useful matlab functions for the functional evalution

% % polynomial function & its integration: appendix A

See appendix A for the

%% minimum values and roots of the function



%% function handle

hy =@(x) x + 2\*exp(-x) - 3;

x=-2:0.2:4;

% plot function

figure;plot(x,hy(x),'--\*')

% minimum & root

x\_min= fminbnd(hy, 0, 4);

x\_root=fzero(hy, 0);

%% the polynomial

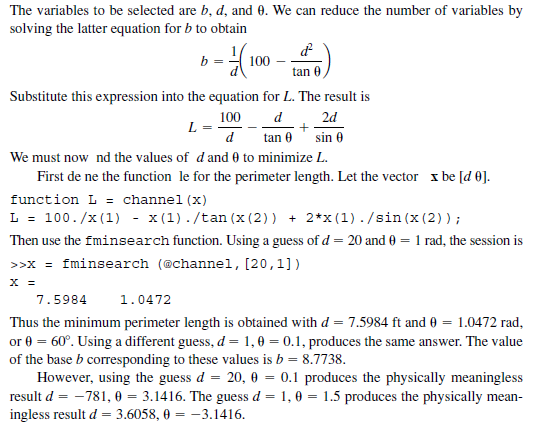
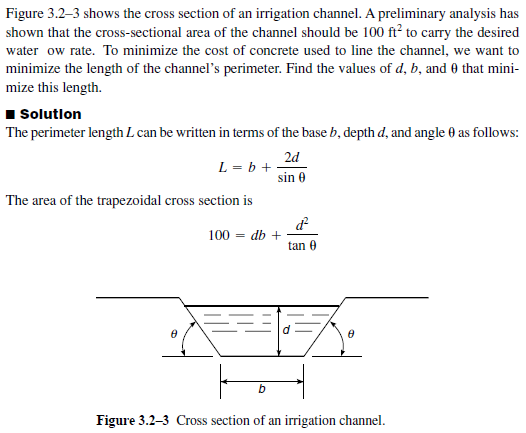
hy2=@(x) (0.025.\*x.^5-0.0625.\*x.^4-0.333.\*x.^3+x.^2);

x=-4:0.2:4;

figure;plot(x,hy2(x),'--\*')

x\_min= fminbnd(hy2, -1, 4);

x2=fzero(hy2,[-4 -1]);



Lh=@(x) 100./x(1) - x(1)./tan(x(2)) + 2\*x(1)./sin(x(2));

x0=[20,1];

x1 = fminsearch (Lh,x0,options);

% notes that x1 is a local minimum