**EGR 111 Quiz 2 MATLAB sheet**

**Make a vector:**

Var = start:increment:end

X = 10:-1:1

Declare: var = [matrix]

B = [2 4 6; 8 10 12]

replace: B(pos­X,posY) = replacenumber

**Plots:**

X = var

Y = var

plot(X,Y, ‘+’)

xlabel(‘x’)

ylabel(‘y’)

title(‘insert title here’)

**Compare 2 functions**

figure

X = linspace(0,2\*pi);

Y = cos(x);

Z = sin(x);

plot(x,y,x,z)

xlabel(‘x’)

ylabel(‘cos & sin’)

title(‘insert title’)

legend(‘sin’, ’cos’)

**Using subplots (separates)**

figure

X= linspace(0,2\*pi);

Y=

Z=

subplot(2,1,1)

plot(x,y)

xlabel(‘x’)

ylabel(‘y1’)

title(‘insert title’)

subplot(2,1,2)

plot(x,z)

xlabel(‘x’)

ylabel(‘y2’)

title(‘insert title’)

**Copy Data from Excel:**

[data, txt, raw] = xlsread(‘file.xlsx’);

hours = data(:,1);

power=data(:,2);

**Audio**

[y,fs]=wavread(file.wav);

sound(y,fs)

**Generate backwards**

X=10:10:100

xcopy = x(1:end)

xreverse=x(end:1:1)

**Make tones**

fs = 44100; %Sampling freq (Hz)

Ts = 1/fs; %Sampling interval(s)

t= 0:Ts:0.5; %Sampling every Ts

E=0.5\*cos(2\*pi\*amp\*t);%tone1

D=0.5\*cos(2\*pi\*amp\*t);%tone2

C=0.5\*cos(2\*pi\*amp\*t);%tone3

s=zeros(1,fs\*0.1); %Silence

x= 0.5\*cos(2\*pi\*500\*t+pi/2);

plot(t,x) %Plots Sinusoid

sound(x,fs)

y=[tone1,tone2,tone1]%concat notes

soundsc(y,fs)

Getting User Input

x = input(‘Enter a number: ’)

disp(‘The value of x is: ’); disp(x)

**Conditional statements**

if x==a

disp

elseif x ~=a

else

end

**Relational Operators**

< less than

> greater than

<= less than or equal to

>= greater than or equal to

== equal to

~= not equal to

& And

| Or

~ Not

**Logical functions**

With A as a vector

any(A) – returns 1 is any of A is true

all(A) – returns 1 if all of A is true

find(A) – returns indices of A that are true

sum(A) – sums the values of the vector

**Inputting data from xls files**

[data,txt,raw] = xlsread(‘name of file.xlsx’)

varName = data(:,1) % copies 1st column

**Making a function**

function d = rad2deg(r)

% function d = rad2deg(r)

% converts radians to degrees

% r – input angle in radians

% d- output angle in degrees

d = r \* 180/pi

function [a,m]= rect2polar(x,y)

% function [a,m]= rect2polar(x,y)

% Converts the real and imaginary parts

% of a complex number to the magnitude

% and angle

% x -real part (input)

% y -imaginary part (input)

% a -angle in radians (output)

% m -magnitude (output)

z = x+j\*y;

a = angle(z);

m = abs(z);

**Complex Numbers**

[a,b]= pol2cart(pi/4)

real(z) – finds real part of z

imag(z) – finds imaginary part of z

abs(z) – finds magnitude of z

angle(z)- finds the angle of z in radians

[theta, r] = cart2pol(a,b)- Cartesian >polar

[a,b]=pol2cart(theta,r)