

MATLAB Assignment 1

Introduction: The purpose of this lab was to become familiar with basic Matlab commands and to learn how to manipulate signals as well as write basic functions.

Problem 1:

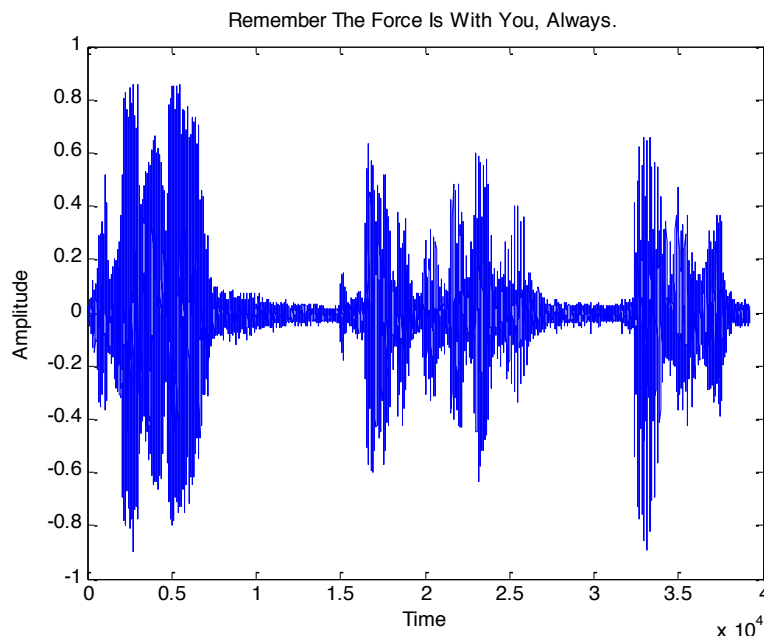
Results:

In this problem, I loaded a sound file, played it, and plotted and labeled the result.

M-File:

```
%Problem 1: Input data from sound file "Message1"
%Play sound file at a rate of 11025 and plot the results.
sound (Message1, 11025);
plot (Message1);
%Label the different parts of the plot
title ('Remember The Force Is With You, Always. ');
xlabel ('Time');
ylabel ('Amplitude');
```

Plot:



Problem 2:

Results:

(Answers to questions in problem 2):

You should use array multiplication to define matrix Z because you are multiplying each of the elements together.

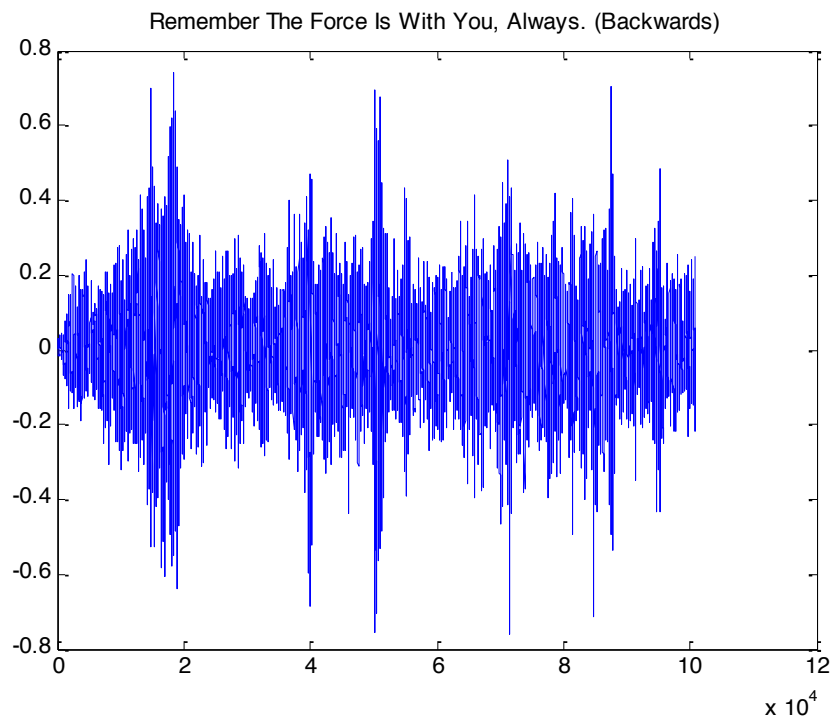
The “flipud” command is the correct one to use. This flips the content of the message from “up to down” which reverses the signal. Flipping it the other way would not produce the desired result because the message is only a single column vector. Therefore flipping it from “left to right” would not make sense.

M-File:

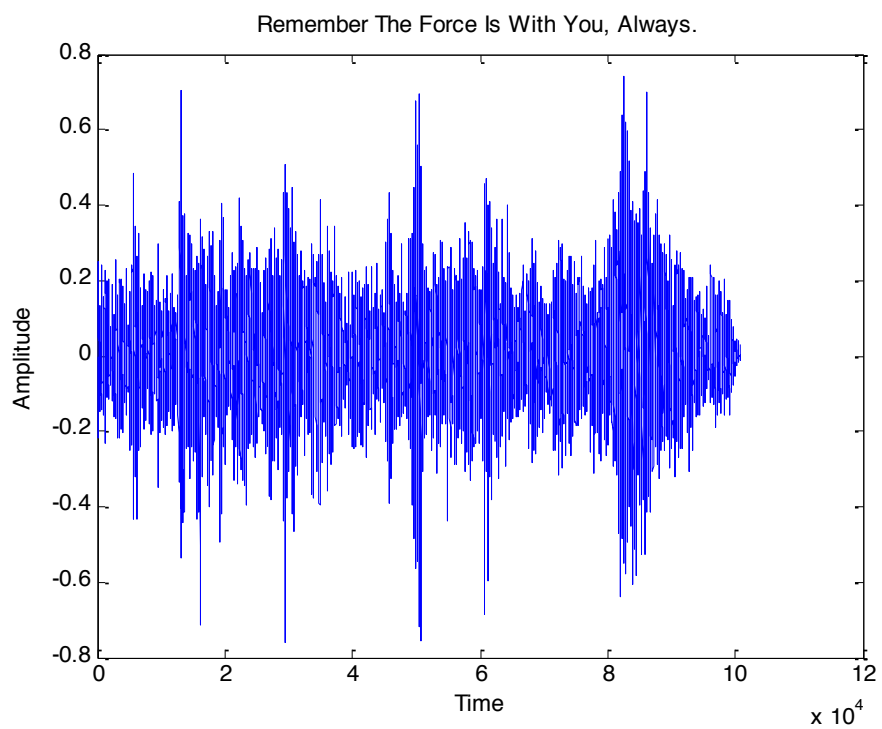
```
%Problem2
%First, import the data in the binary MAT-file and load the variable X14
%N is the number of rows in X14
N=100880;
%n is the sequence 1, 2, 3,...N
n=1:N;
%Use matrix n and given operations to create matrix C
C=10*cos(n*pi/6);
%Assign M to the magnitudes of the samples of X14
M=abs(X14);
%Assign K to the imaginary components of X14\
K=imag(X14);
%Need to transpose matrix C in order for dimensions to match
Z=(M-50).*(K+C');
%Eliminate extraneous imaginary components of the samples of X14
real(Z);
%Play the decoded message at a rate of 11025
sound(Z, 11025);
%Plot Z and label the graph
figure;
plot(Z);
title('Remember, The Force Is With You, Always. (Backwards)');
xlabel('Time');
ylabel('Amplitude');
%Create the desired signal Y by flipping Z
Y=flipud(Z);
%Play, plot, and label the signal Y
sound(Y, 11025);
figure;
plot(Y);
title('Remember The Force Is With You, Always. ');
xlabel('Time');
ylabel('Amplitude');
```

Plots:

(Plot of Z):



(Plot of Y):



Problem 3:

Results:

>> Problem3

ans =

0.0886 + 0.0574i

ans =

0.5748

>> Problem3

ans =

0.0886 + 0.0574i

a =

0.5748

b =

0.1056

M-File:

```
%Problem 3
%Compute and express in both rectangular and polar coordinates
%Result in rectangular coordinates
((6 + i*2)*(1 + i)) / ((4 + i*9)*(7 - 5*i))
%Result in polar coordinates
[a,b]=cart2pol(real(((6 + i*2)*(1 + i)) / ((4 + i*9)*(7 - 5*i))), imag(((6 + i*2)*(1 + i)) / ((4 + i*9)*(7 - 5*i))))
```

Problem 4:

>>save('sol2b.mat')

>>save('sol2b.mat', 'Y', 'M', 'C')

>>save('sol2b_C.txt', 'C', '-ascii')

type('sol2b_C.txt')

Problem 5:

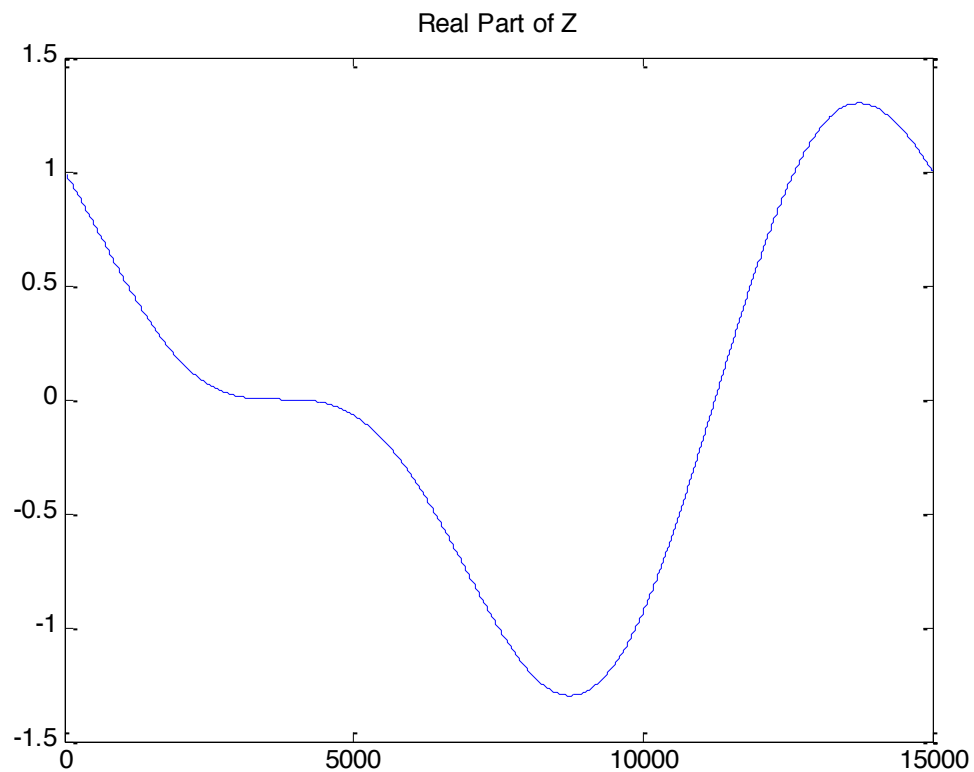
M-File:

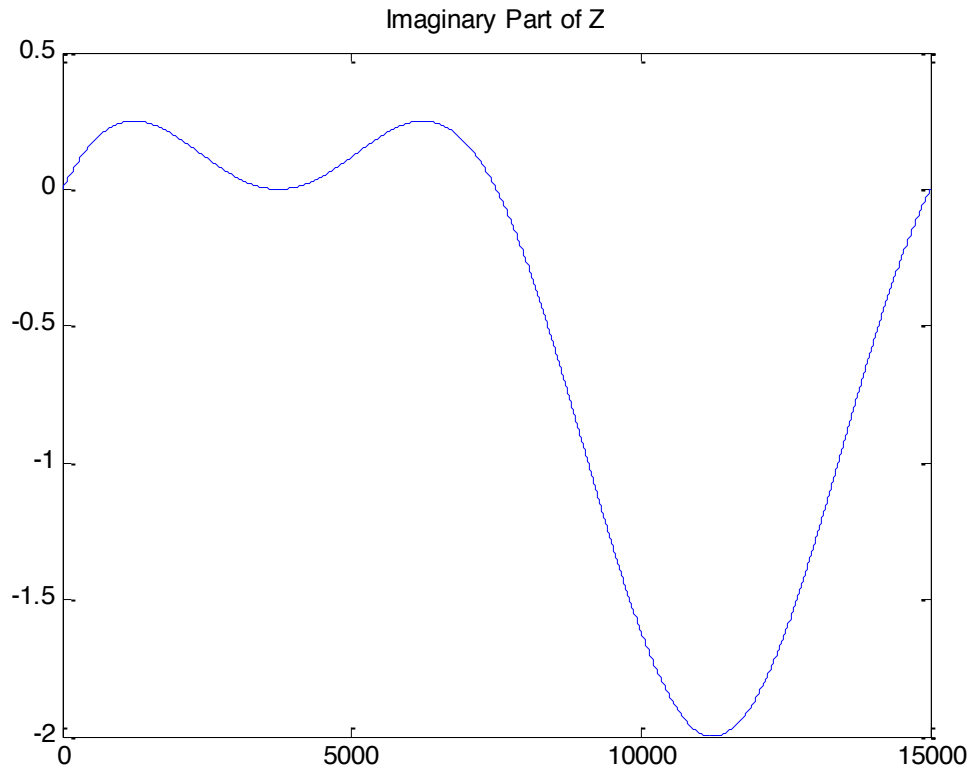
```

%Problem5
%create a function z by first defining r and n
r=1-sin(n);
%n varies from 0 to 2pi (using hints)
n=0:14999;
n=n*2*pi/15000;
%function z (composed of r and n)
z=r.*exp(1j*n);
%Plot of the real part of z
figure(1);
plot(real(z));
%title
title('Real Part of Z');
%Plot of imaginary part of Z
figure(2);
plot(imag(z));
%title
title('Imaginary Part of Z');

```

Plots:





Problem 6:

M-File:

```
%Problem6
%Function 'half' removes every other vector from an arbitrary length vector
%creating a shorter vector made up of only the elements with odd index
%numbers in the original vector.
%Funciton 'half'

function output=half(input)
output=input(1:2:end);

% function 'twice' that creates a longer vector by adding an additional
%element
%between neighboring elements in the original vector.
```