Robert Chance
A13088857

ECE 101 Lab 1

Introduction

The purpose of this lab was to become acquainted with Matlab by using the software to solve several math and engineering problems.

Results

Problem 1 - I loaded "Message1" into Matlab as an ASCII file, and proceeded to play the file at 11025 Hz. I then produced a graph of the audio samples and used Matlab functions to label the graph, as well as the x and y axes. This audio file was a short clip from Star Wars.

Problem 2 - I loaded "Problem2.mat" into Matlab, which contained a variable "X16", a sound file encoded as a set of complex numbers. I used the given equation to decode the file, making use of several other variables, functions, and matrix operations. I played this file at 11025 Hz and discovered it was time reversed, which I corrected by using the "flipud" function. This flipped the vector across the horizontal axis, reversing the direction. I played the sound file again, and discovered it also to be an audio clip from Star Wars. From here, I plotted and labeled the graphs for both the decoded audio clip, and the time reversed audio clip.

Problem 3 - This problem involved simplifying a complex equation and converting the answer from rectangular coordinates to polar coordinates. I first used Matlab to simplify the equation into rectangular coordinates, giving a value of .3669 + .3195j. I then converted that answer to polar coordinates by calling the "abs" and "angle" functions with my rectangular expression as a parameter. This gave values of R=.4865 and θ=-.7169 radians. I then produced those same values by calculating R and θ mathematically.

Problem 4 - For this problem, I was asked to save variables into files of different types. I first called the function save('prob4.mat') to save all of my workspace variables to a file named 'prob4.mat'. In order to save specific variables, I called the same function, but with parameters as follows: save('prob4.mat', 'Y', 'M', 'C'). This replaced the contents of 'prob4.mat' with only the variables specified in the arguments of the function call. Finally, to save the variable 'C' as a list of floating point numbers in an ASCII file, I used the following function call: save('prob2_C.txt', 'C', '-ascii', '-double'). This produced the intended result.

Problem 5 - I began this problem by creating a vector, n, with discrete values [0,14999], and then scaling these values to encompass one period of the function $z(\theta) = sin2\theta \cdot e^{j\theta}$. I then called 'plot(z)' to plot this function to the complex plane. I then plotted the real and imaginary parts of the function and compared them with the plots of $Real\{z(\theta)\} = sin2\theta \cdot cos\theta$, and $Imag\{z(\theta)\} = sin2\theta \cdot sin\theta$, respectively. These graphs coincided.

Problem 6 - I wrote a function called 'half' which took a vector as a parameter, and returned a vector of half the length, containing every other value from the original. I did this by creating a new vector with the colon operand, and populating it with every by vector manipulation, only taking every 2nd value.

To create the function 'twice', I created a new vector of zeros as placeholders, with twice the length of the first, minus one. I then replaced every other zero with the values from the first vector, and replaced the remaining zeros by averaging the adjacent values.

I used these functions to manipulate simple vectors, confirming that my procedure was correct. I discovered that playing the audio files through the 'half' function caused the audio speed to be doubled, and likewise, playing them through 'twice' caused the audio file speed to be cut in half.

I also discovered that calling 'half(twice(X))' will return exactly the same vector, but calling 'twice(half(X))' will only produce the same vector if an even number of terms are removed with half(X). There is information loss otherwise.