Lab 3

Properties of Sinusoids

1 Introduction

This document will serve as the specification ("specs") for the program you are being asked to write. Be glad that none of you are programming for a living, since the specs you usually get are not nearly this nice!

Once you understand what needs to be done, your task is to create a MATLAB function which *implements* the specs you have been given.

Datasets and other files that you need for this assignment can be downloaded from:
http://bigiron.atistar.net/~stepp/coding/

2 The Problem

Very often in our research, we must deal with sinusoidal time series, generally the output of some kind of oscillator (be it mechanical or human).

In order to analyze these time series appropriately, we must know certain things about them, e.g. frequency, period, amplitude.

Most properties of cyclic data, however, require knowledge of the cycles. Specifically, at what times each cycle hits its respective peak or valley.

2.1 Part 1 - Peak Detection

Problem: Given a cyclic time series, find the times of each peak (or peaks and valleys).

2.1.1 Program Specification

Write a function named `find_peaks` with *minimally* the following function declaration:

```
function peaks = find_peaks( timeseries )
```

—OR—

```
function [peaks valleys] = find_peaks( timeseries )
```

Depending on whether or not you are calculating both peaks and valleys.

This function takes a single parameter which is the MATLAB vector containing the cyclic time series.

Note that I say *minimally* above, your implementation may require additional parameters.

The output parameter, `peaks`, is a vector containing the times of each peak in `timeseries`. The output parameter, `valleys`, is the same, but containing valley times.

2.1.2 Implementation

Your function should be saved in a file named `find_peaks.m`.

The way your implementation solves the problem is up to you, but if you need to get started just send me an email.

Once you have run your function, and are happy with your output, send your `find_peaks.m` to me. I'll give you constructive comments and hints on ways to make it better.

2.2 Part 2 - Using What You've Done

Now that you have a function for finding peak times, you have an essential tool needed for finding out more interesting things.

Problem 1: Find the frequency of a given time series. The frequency may change over time, so you must return the frequency cycle by cycle (a vector of frequencies, not just a single number).

Assume a sampling frequency of 100 Hz.

Problem 2: Find the amplitude of a given time series. The amplitude may change over time, so you must return the amplitude cycle by cycle (a vector of amplitudes, not just a single number).

Problem 3: In order to remove transients and other problems at the endpoints, we only want some cycles from the middle of our time series. Extract some number of cycles, after skipping some number at the beginning.

2.2.1 Program Specification

Write the following functions:

```
function freqs = find_freq( timeseries )
function amps = find_amp( timeseries )
function cycles = extract_cycles( timeseries, num_skip, num_cycles )
```

Your implementations may require additional parameters.

The input parameter `timeseries` contains the cyclic data. The last function has additional input parameters, `num_skip` and `num_cycles`. These are both single integers, indicating how many cycles to skip, and how many cycles after that should be extracted.

You will need to use your `find_peaks` function in all three of these.

2.2.2 Implementation

Your functions should be saved in files `find_freq.m`, `find_amp.m`, `extract_cycles.m`.

The way your implementations solve the problem is up to you, but if you need to get started just send me an email.