Discrete-time filters are often implemented in hardware using fixed-point arithmetic. Quantization to a finite number of bits can significantly affect filter performance. In this project you will explore some of the issues associated with implementing IIR filters in fixed-point arithmetic. To complete this project, you will need to do some additional reading on your own. You have been practicing your technical reading skills throughout the semester (by preparing for the RATs). With this project, you’ll have the opportunity to put those reading skills to good use.

You will work in the same teams as you did for Matlab Project IV. Each team will turn in a single report. The report must include all of the analytical (i.e., pencil/paper) work, Matlab plots and code, and relevant explanations for each part of the project. A list of guidelines for preparing the report of this project are given below.

- The report coverpage should be signed by all members of the team. The coverpage should also include a brief description of each person’s contribution to the team project.

- The report for must be typed and all pages must be numbered.

- All plots must be neatly annotated with x-axis and y-axis labels and a title. Any graph not labeled will be considered not handed in.

- I will not spend time trying to figure out which graphs are for which problems. When referring to plots in the text, I recommend doing at least one of the following:
  - use figure numbers, e.g., “Figure 1 is a plot of the signal \( x[n] \).”
  - cite the page number they are on, e.g., “The figure at the top of page 4 is a plot of \( x[n] \).”

- All Matlab code must be well-documented and should be included in an appendix at the end of the report.

- Please do not include answers to questions or other descriptions within your Matlab code. You must write a report separate from the Matlab code itself. You are encouraged to include comments in your Matlab code, but I will not consider the comments part of your official report.

1 Quantization Effects in Discrete-Time Filter Structures

1.1 Reading assignment

Read Sections 6.6 and 6.7 of *Discrete-Time Signal Processing* by Oppenheim and Schafer with Buck. Section 6.6 gives an overview of finite precision effects, and Section 6.7 discusses coefficient quantization issues explicitly. Your team should discuss the reading assignment. If you have any questions about the reading, I’ll be happy to meet with each team individually or answer questions via email.
1.2 Book problems

Do the Basic and Intermediate Problems in Section 10.3 (parts a-l) of *Computer Explorations in Signals and Systems* by Buck, Daniel, and Singer. Your solutions should include answers to all of the questions in the book, the required plots, and your Matlab code.

Important notes:

- Part b: you only need to follow the instruction in the first sentence (about zooming in on the frequency response plot). You did the rest of this question in class when we discussed IIR filter design. You do not need to include the solution for the in-class problem in your report.

- The book says to use dpzplot program to do the pole-zero plots. Please do not use this program. I think it is better to use the built-in Matlab program zplane instead. The zplane function takes the same input parameters as dpzplot, e.g., zplane(b,a). Note that b and a must be row vectors. Type help zplane for additional information.

1.3 Analysis of Elliptic Filter from Project IV

In this part, you will repeat parts d-g of the Basic Problems from the book using the elliptic filter designed for Project IV. After implementing the pole-zero plot, frequency response, and impulse response calculations, please answer the following questions for both the 16-bit and 12-bit cases:

1. Does the quantized filter still meet specifications?

2. Is the quantized filter both causal and stable? Why or why not?

Since some groups were unable to design an elliptic filter to meet specifications in Project IV, I am providing the code I used to design the elliptic filter on the next page. You may copy this code to design the filter for this part of the problem.

1.4 Summary Memo

Write a one to two page memo summarizing what you learned from this project.
Code for elliptic filter from Project IV

```matlab
%---------------------------------------------
%%% Design of elliptic filter (from Matlab Project IV, fall 2005)

fs=22050; % sampling frequency (Hz)
fpass=3000; % desired passband edge (Hz)
fstop=3700; % desired stopband edge (Hz)
delta1=.02; % desired passband ripple
delta2=10^(-50/20); % desired stopband ripple

%%% Compute the DT filter pass/stopband edges
wp=2*pi*fpass/fs; % rad/sec
ws=2*pi*fstop/fs; % rad/sec
deltaw=ws-wp; % desired DT transition bandwidth
wc=(ws+wp)/2; % desired cutoff for window designs

%%% Compute the desired ripples in dB for IIR designs
%%% Note: have to scale the specs first since Matlab’s analog designs
%%% have a max gain of 1. Use Problem 7.3 in O/S/B as a guide.
deltahat1=2*delta1/(1+delta1);
deltahat2=delta2/(1+delta1);
sc=1+delta1; % factor to rescale b coefficients after design

Rp=-20*log10(1-delta1);
Rs=-20*log10(delta2);
Rphat=-20*log10(1-deltahat1);
Rshat=-20*log10(deltahat2);

%%% Elliptic design
[ne,wne]=ellipord(wp/pi,ws/pi,Rphat,Rshat); % Divide wp and ws by pi
[be,ae]=ellip(ne,Rphat,Rshat,wne);
be=be*sc; % Rescale
%---------------------------------------------
```