1 a) $f_d \approx 2V\frac{f_t}{c}$

$V = 300\ m/s$

$f_t = 1 \times 10^9$

$c = 3 \times 10^8\ m/s$

$600\ \frac{1\times10^9}{3\times10^8} = 600\ \frac{1}{3} \times 10 = 200 \times 10$

$= 2000\ Hz$

b) $f_t = 1$

$f_t + f_d$ ~~$f_t - f_t + f_d = f_d$~~

~~$f = \frac{1}{dt}$~~ $\Delta f = f_d$

f_d is the frequency shift $0.0005\ s$

$f_d = \frac{1}{T}$ T = duration of shift

c) $f_t = 2 \times 10^9$

$600\ \frac{2\times10^9}{3\times10^8} = \frac{1200}{3} \times 10 = 400 \times 10$

$= 4000\ Hz$

$\frac{1}{4000} = 2.5 \times 10^{-4}\ s$ 4000 samples

yes

2 a)

$$R = \frac{c}{2k}\Delta f = \frac{c}{2\frac{\Delta f}{\Delta t}}\Delta f = \frac{c}{2}\Delta t$$

R = distance meters
c = speed of light m/s
$k = \frac{\Delta f}{\Delta t}$

Δf = frequency change

$V = \frac{2R}{\Delta t}$ 2R is total distance traveled as returned signal is an echo

$V = c$ v = c as v is the speed of propergation

$c = \frac{2R}{\Delta t}$

$\frac{c}{2}\Delta t = R$ $\frac{R}{\Delta t} = \Delta f$

$\frac{c}{2k}\Delta f = R$

b) $\frac{300}{2 \cdot 1}$ 25

$= 3750\ m = 3.75\ Rm$

```
clear;
close;
home;
pkg load signal


data = dlmread("doppler_shift.dat");
seconds = data(:,1);
volts = data(:,2);

T = max(seconds);
dt = (max(seconds)-min(seconds))/length(seconds);
fs  = 1/dt;

%lowering these values makes the spectrograph give sharper results buts breaks the indexing done later on, the scales are messed up and my hardcoded methods don't suit themselves well to that
n1 = 2048;
n2 = 16;
[sdata,info] = stft(volts,n1,n2,n1,"hamming");
sdata = sdata(1:end/2,:);
[n_freq,n_time] = size(sdata);
fbins = [0 fs/2];
tbins = [0 T];
sdata = abs(sdata);
figure(1)
image(tbins,fbins,sdata)
xlabel('Time (seconds)')
ylabel('Frequency (Hz)')
h = colorbar();
colormap('viridis');
set(gca(),'fontsize',18)
set(h,'fontsize',18)
set(gca(),'YDir','normal');
%print -dpng "m3question3.png"


%time at which slice is taken
st = 20.0e-5;
% a single column slice of spectrograph represents a line of the frequency at the time
frequencyData = sdata(round((st/tbins(2))*size(sdata)(1)),:);
```

```
%filter
%window size of 100 seems to give the best results not merging the initial singal and first echo
w = 100;
filter = ones(w)/(w);
filter = filter(1,:);
fFrequencyData = conv(frequencyData,filter);

%min Peak of 5 seems to catch the second more faint echo
[fpks fidx] = findpeaks(fFrequencyData,"MinPeakHeight",5);

f1 = (fidx(1)/size(sdata)(2))*fbins(2);
f2 = (fidx(2)/size(sdata)(2))*fbins(2);
f3 = (fidx(3)/size(sdata)(2))*fbins(2);
%3.5882e+06 hz or 3.5 MHz
df1 = f2 - f1
%1.7030e+07 hz or 17 MHz
df2 = f3 - f1


sf = 16e6;
timeData = sdata(:,round((sf/fbins(2))*size(sdata)(2)));

w = 50;
filter = ones(w)/(w);
filter = filter(1,:);
fTimeData = conv(timeData,filter);
figure(2);
plot(fTimeData);

[tpks tidx] = findpeaks(fTimeData,"MinPeakDistance",100,"MinPeakHeight",5);

ti = (tidx(3)/size(sdata)(1))*tbins(2);
t1 = (tidx(2)/size(sdata)(1))*tbins(2);
t2 = (tidx(1)/size(sdata)(1))*tbins(2);
dt1 = ti-t1;
dt2 = ti-t2;

%speed constant
c = 3e8;

%distance to first aircraft in meters

d1 = (c/2)*dt1
%4.3506km
```

```
d1/1000;
%distance to second aircraft in meters

d2 = (c/2)*dt2
%26.697km
d2/1000;

%playing audio for bonus
%nbits = 16;
%fs = 1e3;
%player1 = audioplayer(fFrequencyData,fs,nbits);
%play(player1);
```

```
clear;
close;
home;
pkg load signal

data = imread("aircraft.jpg");
%data = data(:,:,1);
data = data(1:500,1:500);
w = 3;
k = ones(w)/(w^2);

%averaging filter to reduce noise
data = filter2(k,data);

data = uint8(data);
k = ones(3)./-8;
k(2,2) = 1;
$edge enhancement
data = filter2(k,data);
imshow(data);


%number is 054
%jet is a su-57
```