Table of Contents
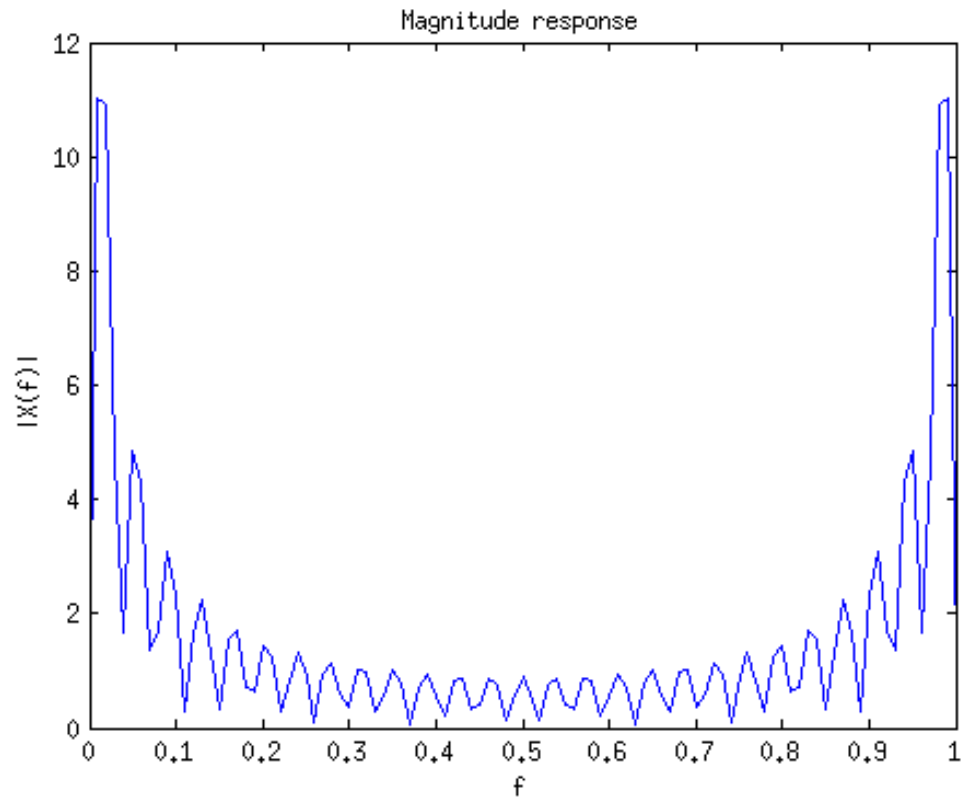
Problem 1

a)

```
f=0:0.01:1;
Nx=28;
numerator=0.9.*exp(-j*2*pi*f).*(1-0.9*exp(-(Nx-1)*j*2*pi*f));
denominator=1-0.9.*exp(-j*2*pi*f);
X=numerator./denominator;

plot(f, abs(X));
title('Magnitude response');
xlabel('f');
ylabel('|X(f)|');
```


b)

```
n=0:1:Nx-1;
```

```matlab
x=0.9.^n;
N1=Nx/4;
N2=Nx/2;
N3=Nx;
N4=2*Nx;

X1=fft(x, N1);
X2=fft(x, N2);
X3=fft(x, N3);
X4=fft(x, N4);
```

d)

```matlab
f1=linspace(0, 1, length(X1));
f2=linspace(0, 1, length(X2));
f3=linspace(0, 1, length(X3));
f4=linspace(0, 1, length(X4));

figure(1)
subplot(2,2,1);
stem(f1, abs(X1), 'r');
hold on;
plot(f, abs(X), 'b');
title('NX/4')
xlabel('f');
ylabel('|X(f)|');


subplot(2,2,2);
stem(f2, abs(X2), 'r');
hold on;
plot(f, abs(X), 'b');
title('NX/2')
xlabel('f');
ylabel('|X(f)|');

subplot(2,2,3);
stem(f3, abs(X3), 'r');
hold on;
plot(f, abs(X), 'b');
title('NX')
xlabel('f');
ylabel('|X(f)|');

subplot(2,2,4);
stem(f4, abs(X4), 'r');
hold on;
plot(f, abs(X), 'b');
title('2NX')
xlabel('f');
ylabel('|X(f)|');


% e)
```
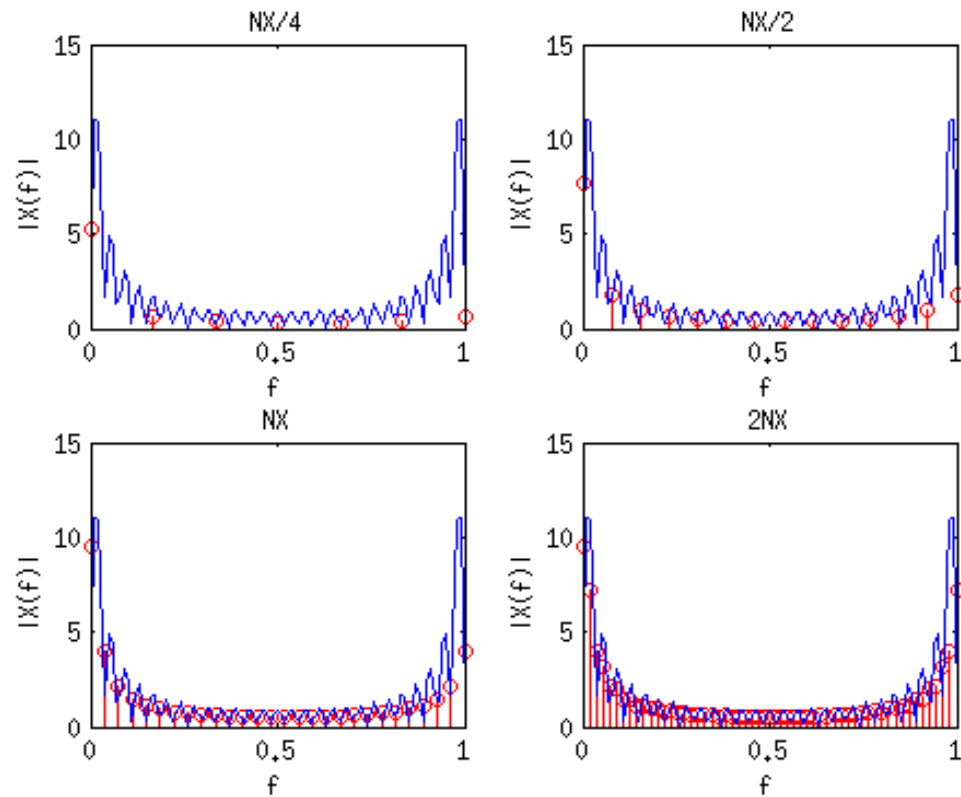
```matlab
% The values are nomalized to a period within the range of [0, 0.5], so that
% the signal can be repeated as long as the values within that range are
% known
```
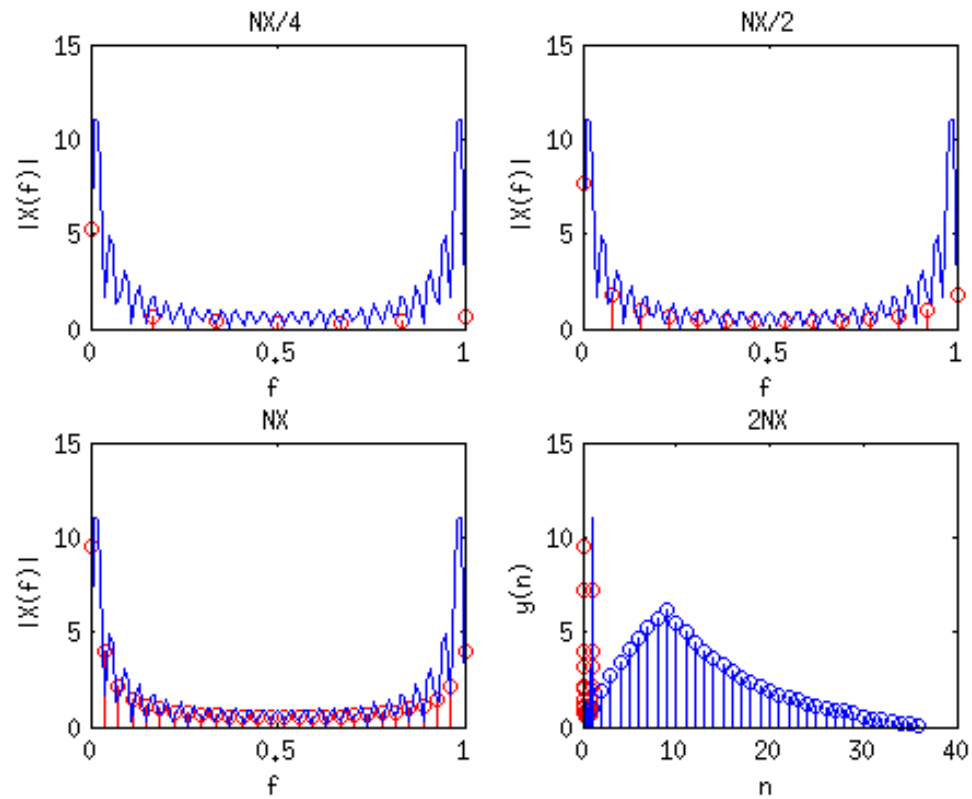

Problem 2

a)

```matlab
Nx=28;
n1=0:1:Nx-1;
Nh=9;
n2=0:1:Nh-1;
h=ones(1, length(n2));
x=0.9.^n1;


y=conv(x, h);

stem(y);
xlabel('n');
ylabel('y(n)');
```

Ny=Nx+Nh-1=36

```
length(y);
```

Since they are the same, the theory checks out

b)

```
N=Nx+Nh-1;
N1=N/4;
N2=N/2;
N3=2*N;

H=fft(h, N); X=fft(x, N); Y=H.*X;
H1=fft(h, N1); X1=fft(x, N1); Y1=H1.*X1;
H2=fft(h, N2); X2=fft(x, N2); Y2=H2.*X2;
H3=fft(h, N3); X3=fft(x, N3); Y3=H3.*X3;

y=ifft(Y); y1=ifft(Y1); y2=ifft(Y2); y3=ifft(Y3);
figure(2)
subplot(2,2,1);
stem(y);
title('Ny');

subplot(2,2,2);
stem(y1);
title('Ny/4');
```
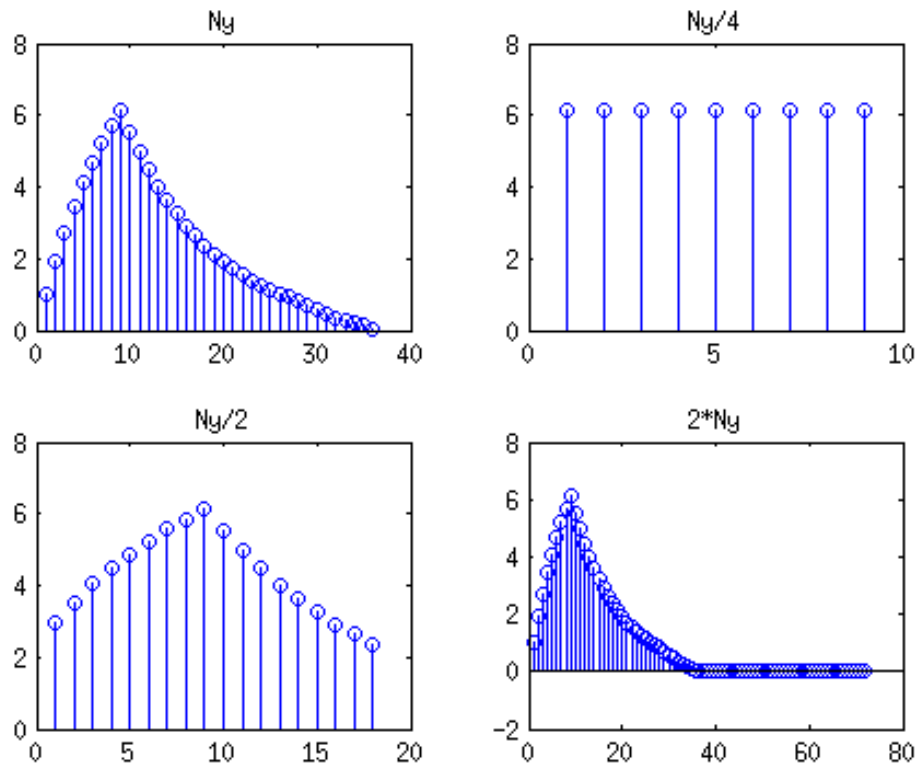
```
subplot(2,2,3);
stem(y2);
title('Ny/2');

subplot(2,2,4);
stem(y3);
title('2*Ny');
```


Problem 3

```
% b)
N1=1024;

f1=7/40;
f2=9/40;
n1=0:99;
n2=0:999;
n3=0:29;
n4=0:9;

x1=sin(2*pi*f1*n1)+sin(2*pi*f2*n1);
x2=sin(2*pi*f1*n2)+sin(2*pi*f2*n2);
x3=sin(2*pi*f1*n3)+sin(2*pi*f2*n3);
x4=sin(2*pi*f1*n4)+sin(2*pi*f2*n4);
```

```
Y1=fft(x1, N1);
Y2=fft(x2, N1);
Y3=fft(x3, N1);
Y4=fft(x4, N1);

f1=linspace(0, 1, length(Y1));
f2=linspace(0, 1, length(Y2));
f3=linspace(0, 1, length(Y3));
f4=linspace(0, 1, length(Y4));

figure(3)
subplot(2, 2, 1);
plot(f1, abs(Y1));
title('length=1024');
xlabel('f');
ylabel('|X(f)|');

subplot(2,2,2);
plot(f2, abs(Y2));
title('length=1000');
xlabel('f');
ylabel('|X(f)|');

subplot(2, 2, 3);
plot(f3, abs(Y3));
title('length=30');
xlabel('f');
ylabel('|X(f)|');

subplot(2,2,4);
plot(f4, abs(Y4));
title('length=10');
xlabel('f');
ylabel('|X(f)|');
```
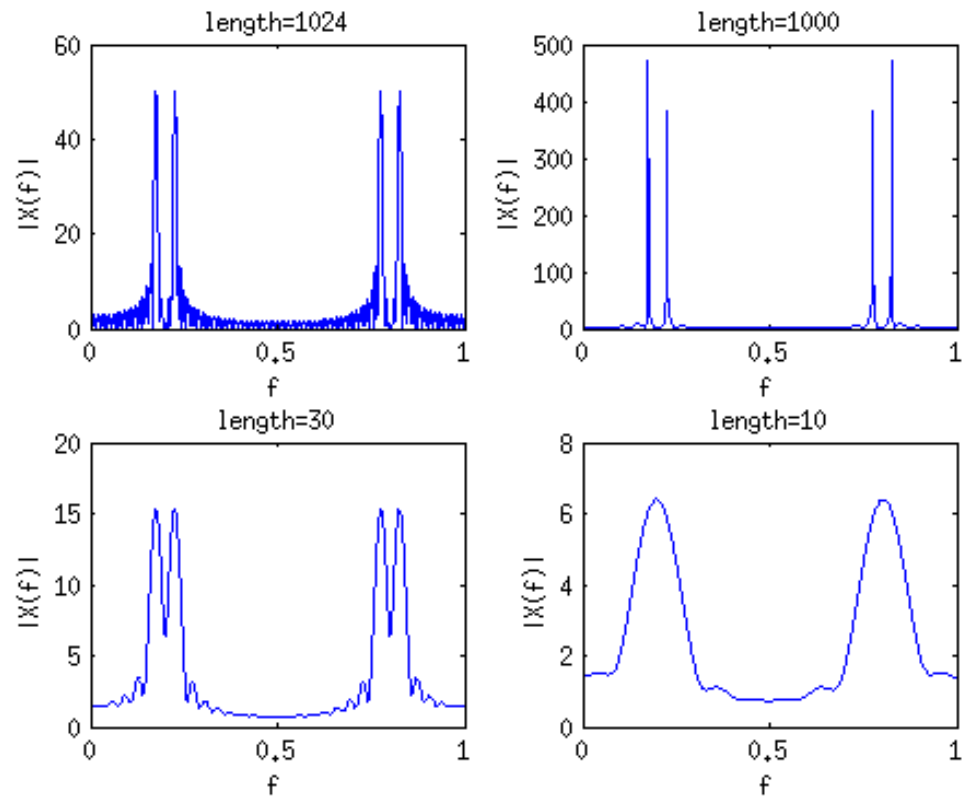
The smaller length between samples, the more accurate/narrow dft/fft

```matlab
% c)
n=0:99;

f1=7/40;
f2=9/40;

N1=256;
N2=128;

x=sin(2*pi*f1*n)+sin(2*pi*f2*n);

Y1=fft(x, N1);
Y2=fft(x, N2);
Y3=fft(x, N3);
Y4=fft(x, N4);

f1=linspace(0, 1, length(Y1));
f2=linspace(0, 1, length(Y2));

figure(4)
subplot(2, 1, 1);
plot(f1, abs(Y1));
title('DFT length=256');
xlabel('f');
```
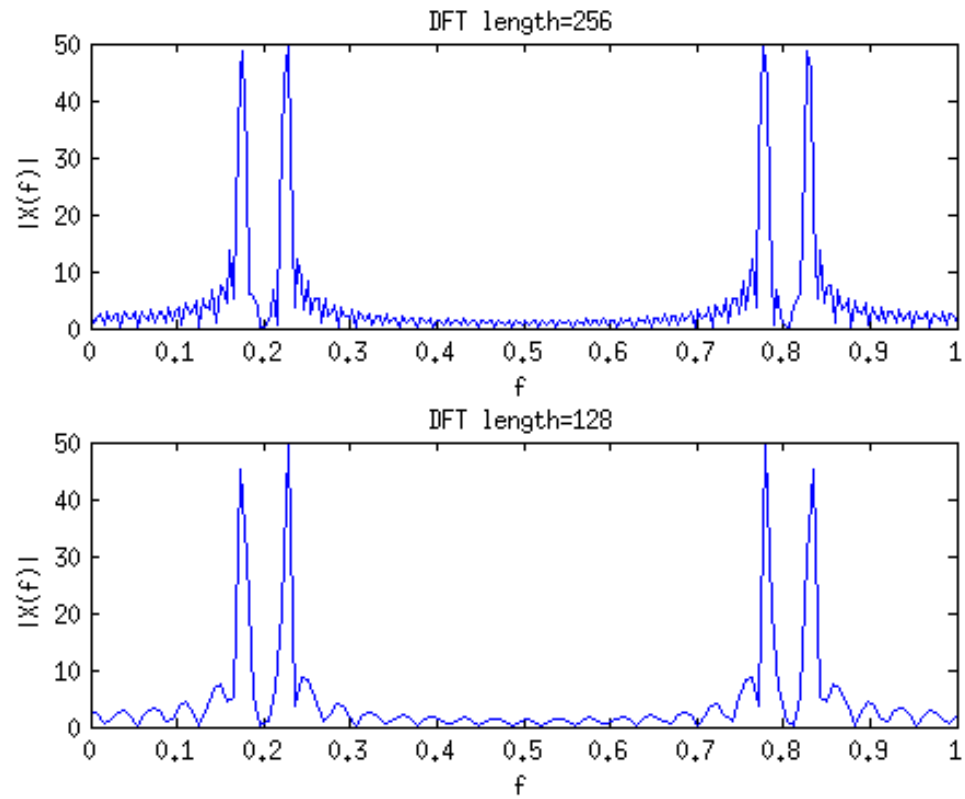
```
ylabel('|X(f)|');

subplot(2,1,2);
plot(f2, abs(Y2));
title('DFT length=128');
xlabel('f');
ylabel('|X(f)|');
```


The higher the DFT length, the narrower amplitude response. I assume 128 and 256 are used as DFT lenghts because the fft is most effective with multiples of 2.

Published with MATLAB® R2013a