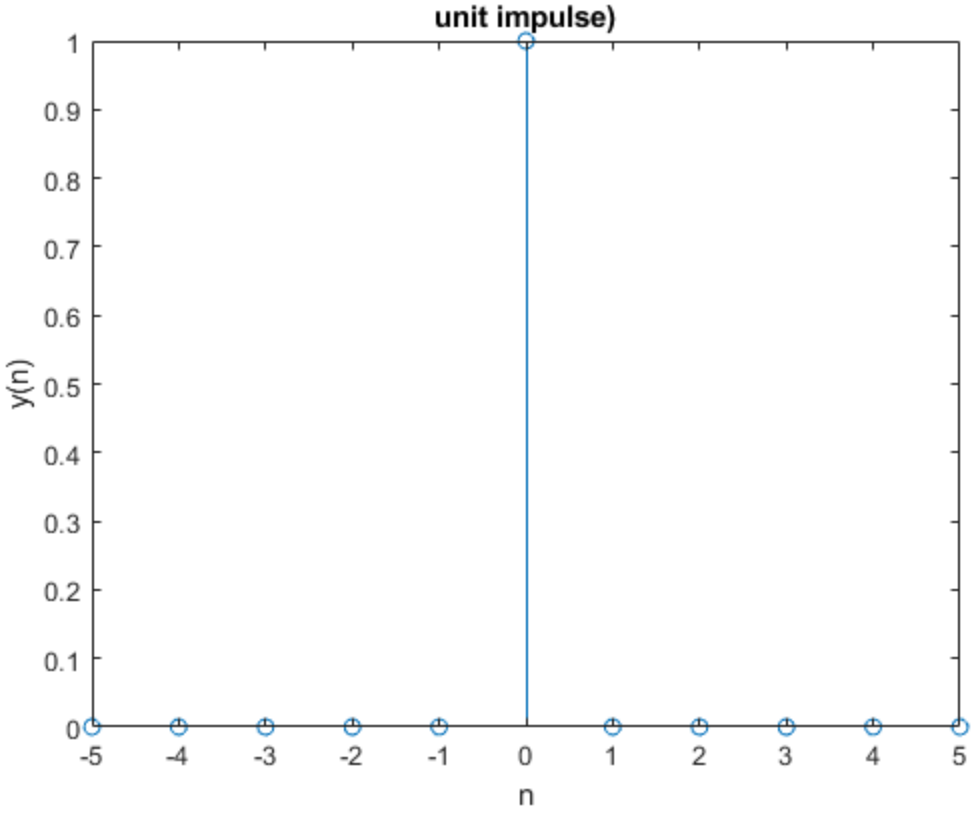
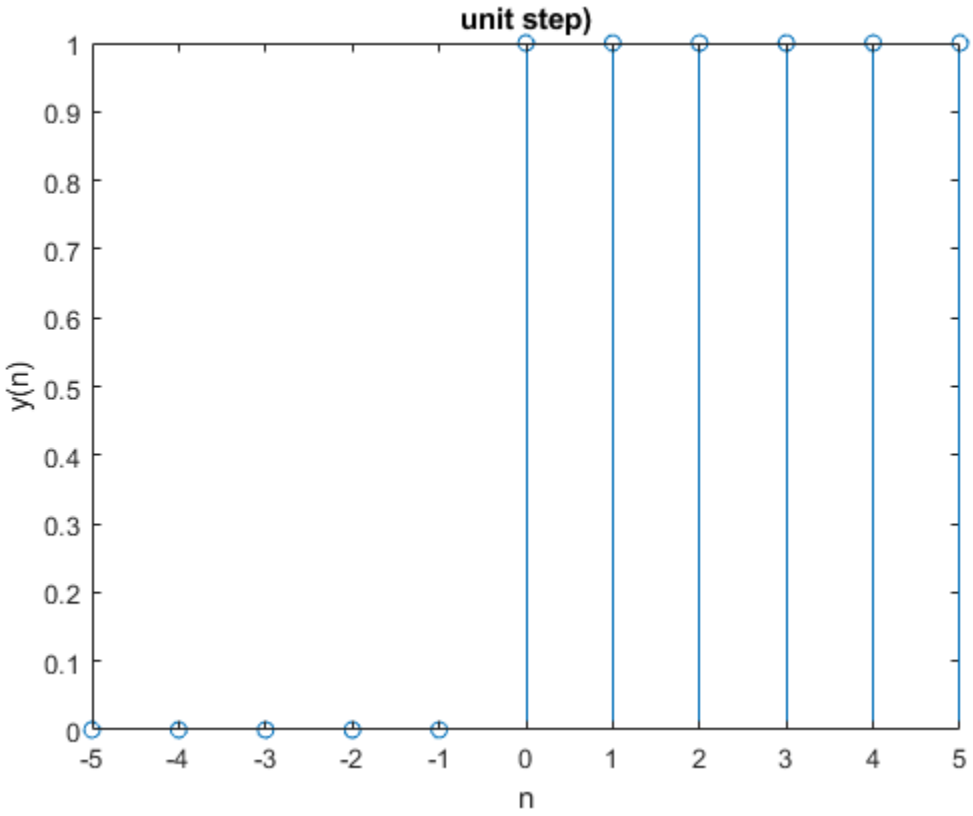
Table of Contents

Problem 1

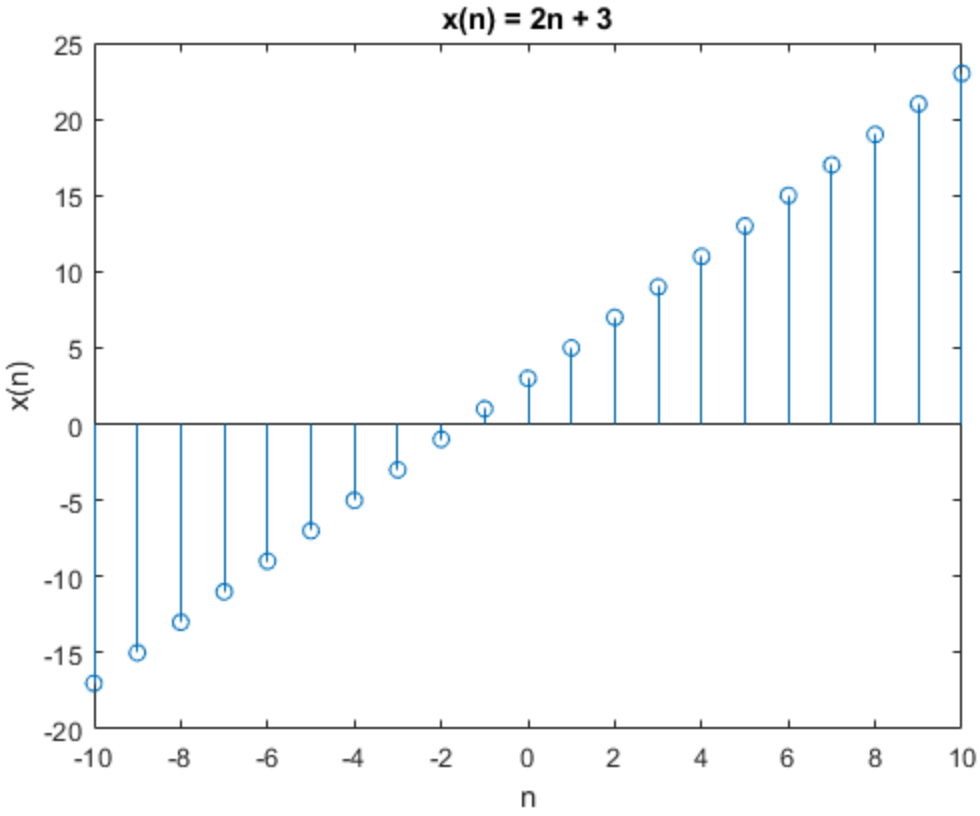
```
n_p = 0;
n1 = -5;
n2 = 5;

[y,n] = unit_impulse(n_p, n1, n2);

figure(1)
stem(n,y),title('unit impulse)'), xlabel('n'), ylabel('y(n)')

% Unit step
n_s = 0;
n1 = -5;
n2 = 5;

[y,n] = unit_step(n_s, n1, n2);

figure(2)
stem(n,y), title('unit step)'), xlabel('n'), ylabel('y(n)')
```

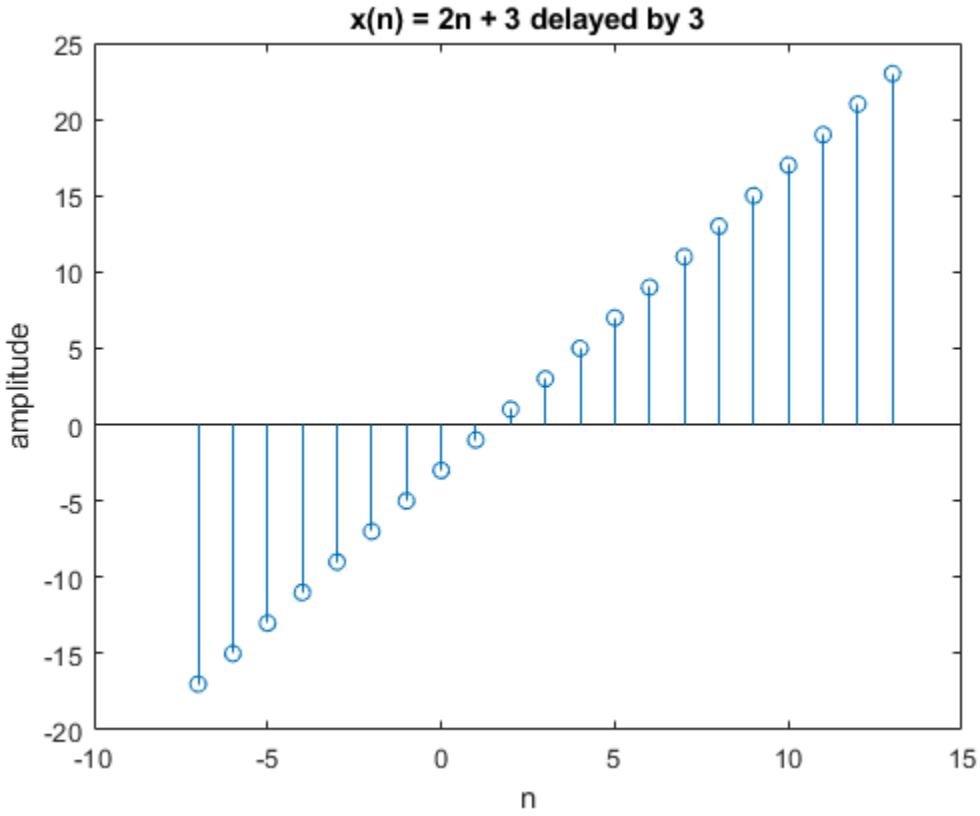

unit impulse)
y(n)
n
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
-5
-4
-3
-2
-1
0
1
2
3
4
5

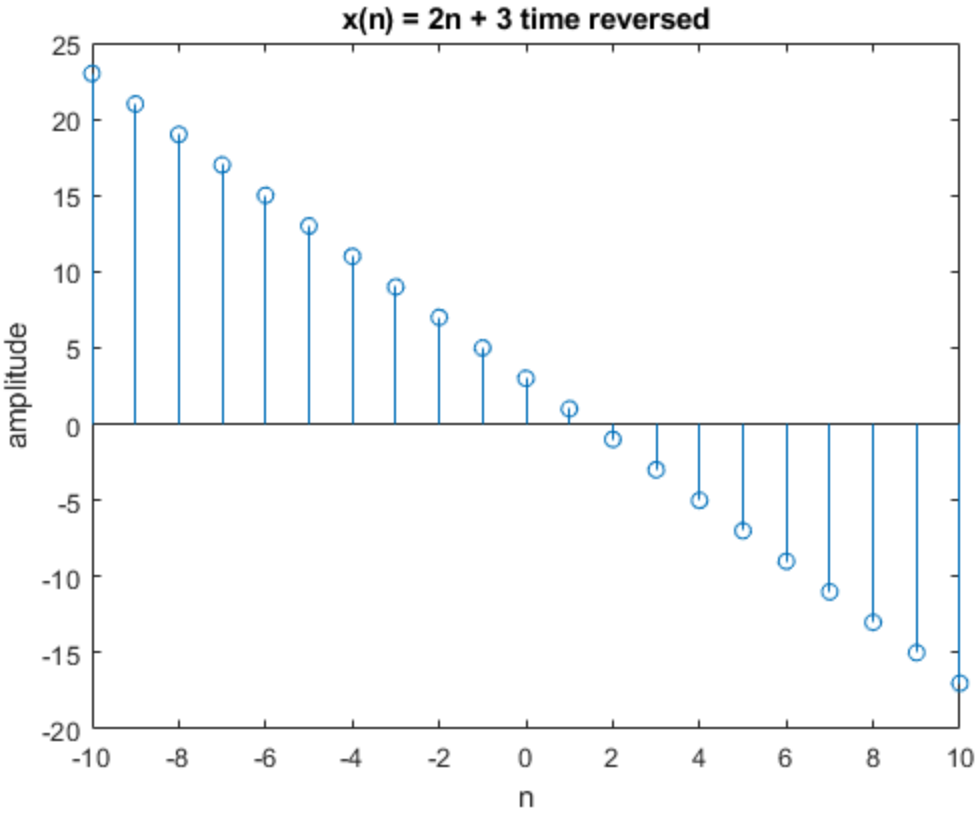

unit step)
y(n)
n
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
-5
-4
-3
-2
-1
0
1
2
3
4
5

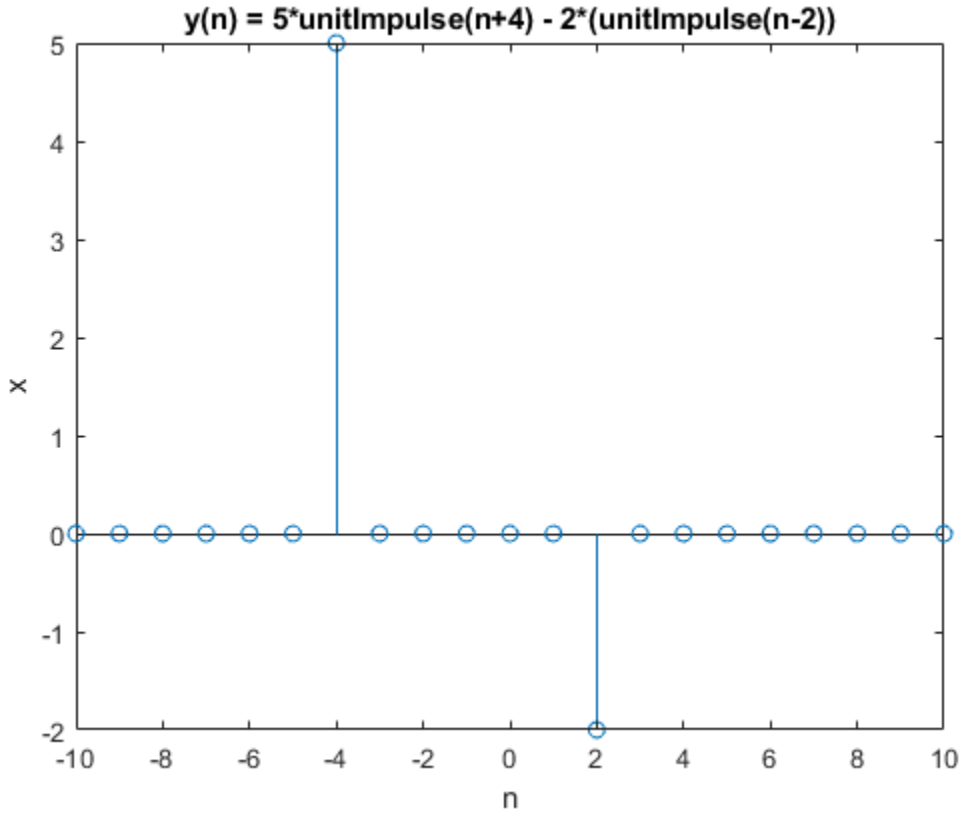
Problem 2

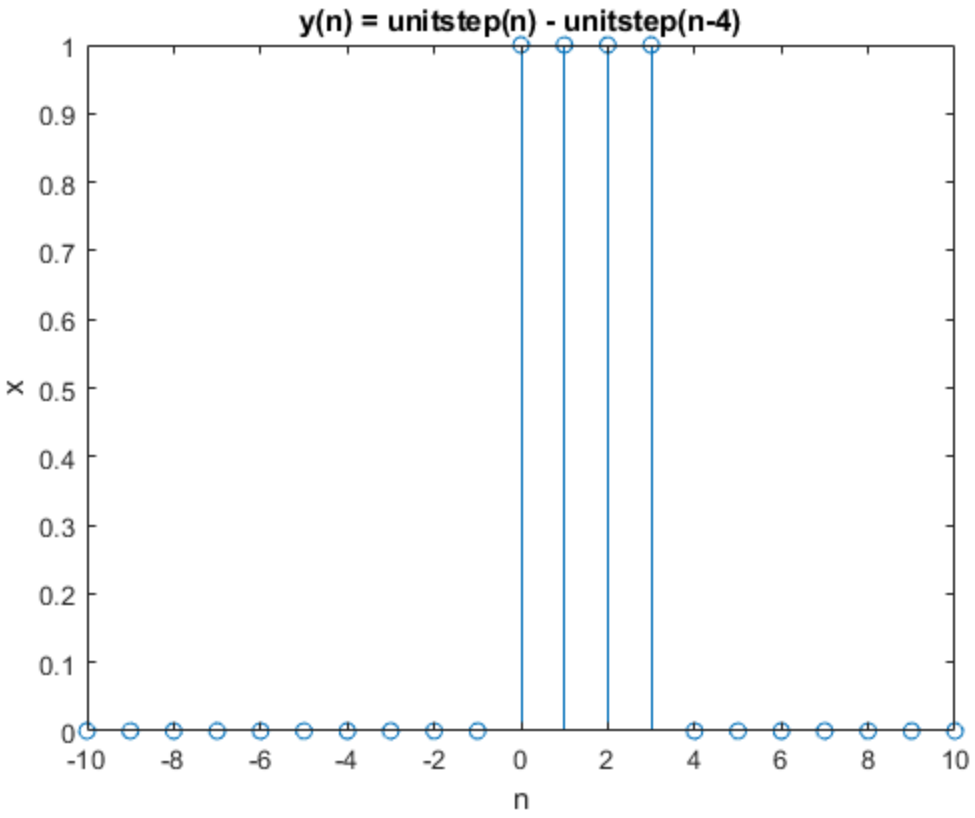
```
%2.2
n = -10:10;
x = 2*n + 3;

%a
figure(3)
stem(n,x),xlabel('n'), ylabel('x(n)'), title('x(n) = 2n + 3')

%b
[y,m] = timeShift(x,n,3);
figure(4)
stem(m,y), title('x(n) = 2n + 3 delayed by 3'), ylabel('amplitude'),
 xlabel('n')

%c
[y,m] = timeReverse(x,n);
figure(5)
stem(m,y), title('x(n) = 2n + 3 time reversed'), ylabel('amplitude'),
 xlabel('n')

%2.3
[y1,m] = unit_impulse(-4, -10, 10);
[y2,m] = unit_impulse(2, -10, 10);

y = 5*y1 - 2*y2;
figure(6)
stem(n,y),xlabel('n'), ylabel('x'), title( 'y(n) = 5*unitImpulse(n+4) -
 2*(unitImpulse(n-2))')

%2.4
z = unit_step(0,-10,10) - unit_step(4,-10,10);

figure(7)
stem(n,z),xlabel('n'), ylabel('x'), title( 'y(n) = unitstep(n) -
 unitstep(n-4)')
```


x(n) = 2n + 3
x(n)
n
25
20
15
10
5
0
-5
-10
-15
-20
-10
-8
-6
-4
-2
0
2
4
6
8
10


x(n) = 2n + 3 delayed by 3
amplitude
n
25
20
15
10
5
0
-5
-10
-15
-20
-10
-5
0
5
10
15

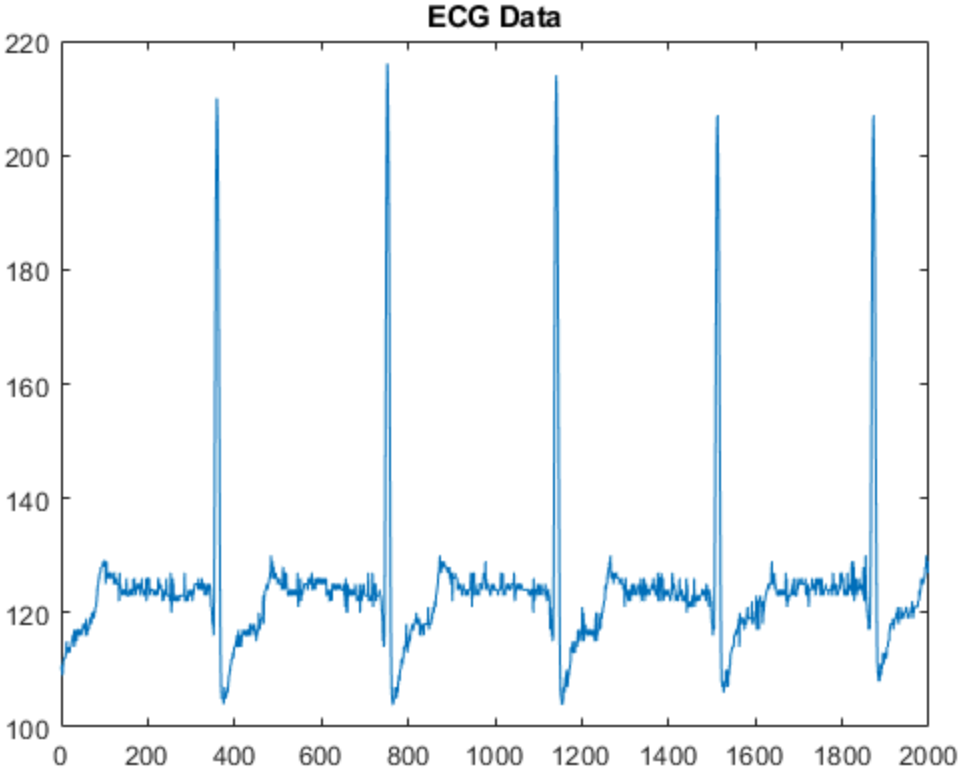

x(n) = 2n + 3 time reversed
amplitude
n
25
20
15
10
5
0
-5
-10
-15
-20
-10
-8
-6
-4
-2
0
2
4
6
8
10

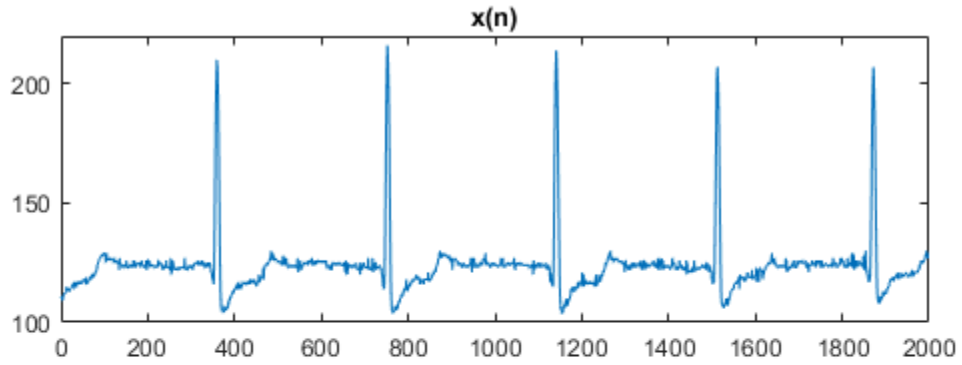

y(n) = 5*unitImpulse(n+4) - 2*(unitImpulse(n-2))
x
n
5
4
3
2
1
0
-1
-2
-10
-8
-6
-4
-2
0
2
4
6
8
10

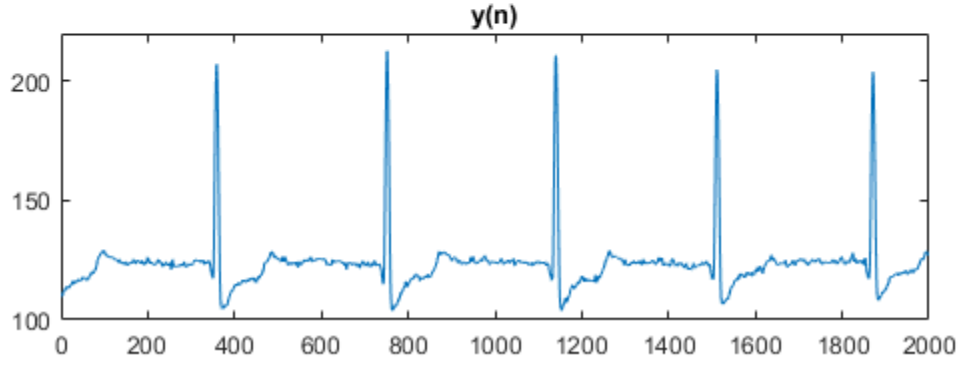


Problem 3

```
load("SAMPLE_ECG.mat")

x = ECG_Data;
figure(8)
plot(x),title('ECG Data')

for i = 1:length(x)-2
    y(i) = (x(i)+x(i+1) + x(i + 2))/3;
end

figure(9)
subplot(2,1,1)
plot(x),title('x(n)'),axis([0 2000 100 220]);
subplot(2,1,2)
plot(y),title('y(n)'),axis([0 2000 100 220]);

% Yes y(n) looks smoother there is less noise before every spike. For
% example, looking between 200 and 400 on the x axis you see less movement
% from the line.
```


ECG Data
220
200
180
160
140
120
100
0
200
400
600
800
1000
1200
1400
1600
1800
2000


x(n)
200
150
100
0
200
400
600
800
1000
1200
1400
1600
1800
2000


y(n)
200
150
100
0
200
400
600
800
1000
1200
1400
1600
1800
2000

Unit Step Function

```
function [y,n] = unit_step(n_s,n1,n2)
n = n1:n2;
y = [(n - n_s) >= 0];   %y = 1 when n - n_s >= 0
end
```

Unit Impulse Function

```
function [y,n] = unit_impulse(n_p,n1, n2)
n = n1:n2;
y = [(n-n_p) == 0];
end
```

Time Reverse Function

```
function [y,m] = timeReverse(x,n)
y = fliplr(x);
m = n;
end
```

Time Shift Function

```
function [y,m] = timeShift(x,n, n_d)
y = x;
m = n_d + n;
end
```

Published with MATLAB® R2022b