Assignment 1

SYSC4405

Jesse Levine 101185127

1 Question 1

The following answers are based on the systems described by the D-E below:

$$y[n] = 5x[n] - 3x[n-2] + x[n-4] \tag{1}$$

1.1 (a)

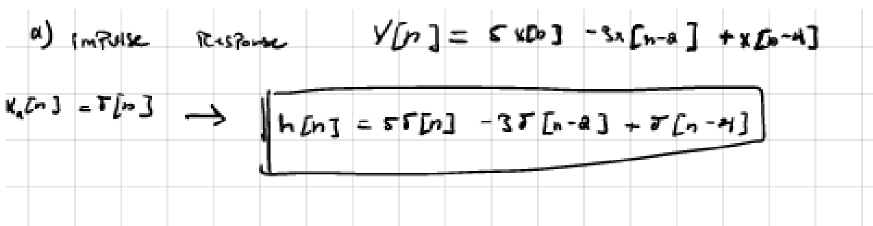

Figure 1: Impulse Response of the System

1.2 (b)

$x_b[n] = \delta[n] + 3\delta[n-1] - 4\delta[n-3]$

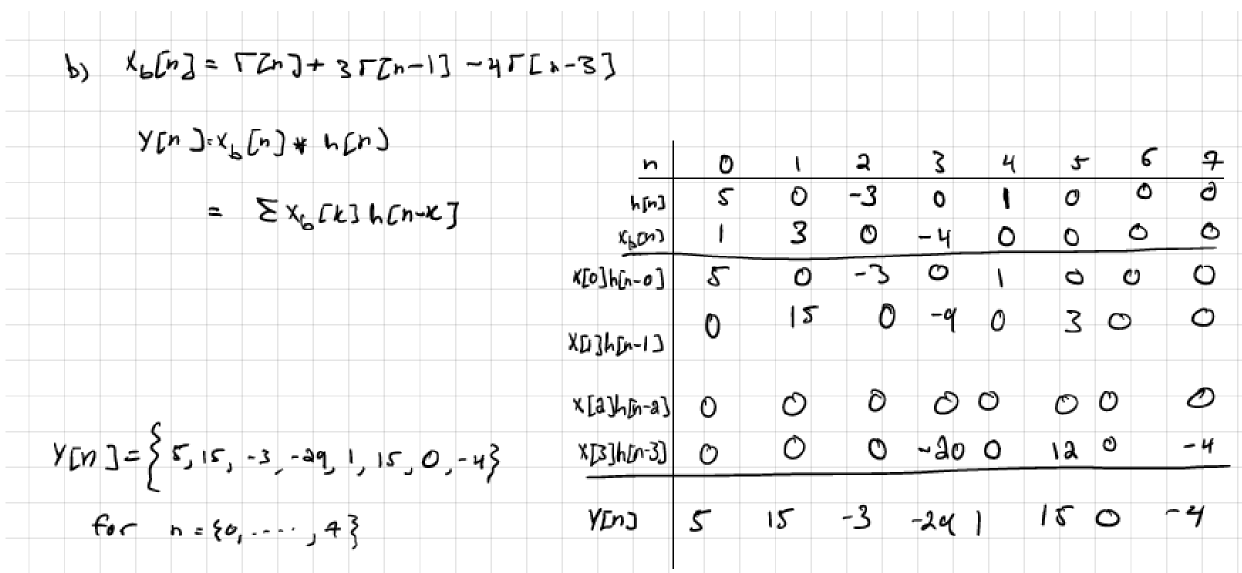

Figure 2: Response of System to Input $x_b[n]$

1.3 (c)

```
%1_c
b = [5, 0, -3, 0, 1];
a = 1;
n = 0:7;

x = zeros(1, length(n));

x(n == 0) = 1;
x(n == 1) = 3;
x(n == 3) = -4;

y = filter(b, a, x);
```

Figure 3: Code for part c

```
y =

  Columns 1 through 4

     5    15    -3   -29

  Columns 5 through 8

     1    15     0    -4
```

Figure 4: Response of System from Matlab

The Matlab code simulating the system matches the handwritten analysis in part b.

1.4 (d)

```
%1_d
h = [5, 0, -3, 0, 1];
n_xb = 0:7;
x_b = zeros(1, length(n_xb));

x_b(n_xb == 0) = 1;
x_b(n_xb == 1) = 3;
x_b(n_xb == 3) = -4;

y_1 = conv(h, x_b);
```

Figure 5: Code for Part D

```
y_1 =

  Columns 1 through 6

     5    15    -3   -29     1    15

  Columns 7 through 12

     0    -4     0     0     0     0
```

Figure 6: Response of System using conv() function

The output response of the system using the conv() function matches the output with the filter() function in part c and the hand written analysis in part b.

1.5 (f)

f) unit step response   impulse => $h[n] = 5\delta[n] - 3\delta[n-2] + \delta[n-4]$

$x_f[n] = u[n]$

is 1 $n \geq 0$, 0 elsewhere

$y[n] = x_f[n] * h[n]$

$= \sum h[k]\, x_f[n-k]$

n	0	1	2	3	4	5	6	7
$h[n]$	5	0	-3	0	1	0	0	0
$x_f[n]$	1	1	1	1	1	1	1	1
$h[0]x[n-0]$	5	5	5	5	5	5	5	5
$h[1]x[n-1]$	0	0	0	0	0	0	0	0
$h[2]x[n-2]$	0	0	-3	-3	-3	-3	-3	-3
$h[3]x[n-3]$	0	0	0	0	0	0	0	0
$h[4]x[n-4]$	0	0	0	0	1	1	1	1
$y[n]$	5	5	2	2	3	3	3	3

Figure 7: System's Unit Step Response Handwritten Analysis

```
%1_f
n_xf = 0:7;
x_f = ones(1,length(n_xf));
y_3 = filter(b,a,x_f);
```

Figure 8: Matlab code for Unit Step Response

```
y_3 =

  Columns 1 through 6

     5     5     2     2     3     3

  Columns 7 through 8

     3     3
```

Figure 9: System's Unit Step Response through Matlab

1.6 (g)

Input to system: $x_g[n] = 3\delta[n-2] - u[n-1]$

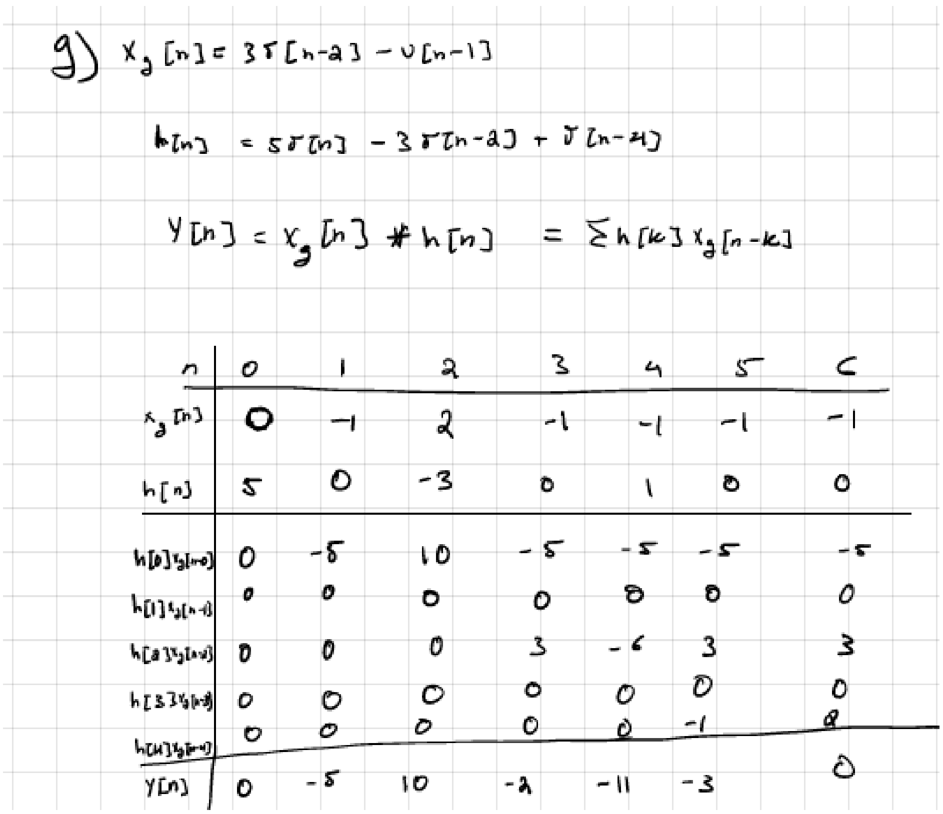

Figure 10: Hand Written Analysis for the System's Response to Input $x_g[n]$

1.7 (h)

```
%1_h
n_xg = 0:7;
x_g = zeros(1,length(n_xg));
x_g(n==2) = 3;
x_g(n >=1) = x_g( n>= 1) -1;

y_34 = filter(b,a,x_g);
```

Figure 11: Code to Simulate System's Response to input $x_g[n]$

```
y_34 =

  Columns 1 through 6

     0    -5    10    -2   -11    -3

  Columns 7 through 8

     0    -3
```

Figure 12: Output

The results are exactly as predicted in the handwritten analysis in part g.

1.8 (i)

The system is stable. The D-E in equation 1 describes an FIR system; FIR systems are always stable.

1.9 (j)

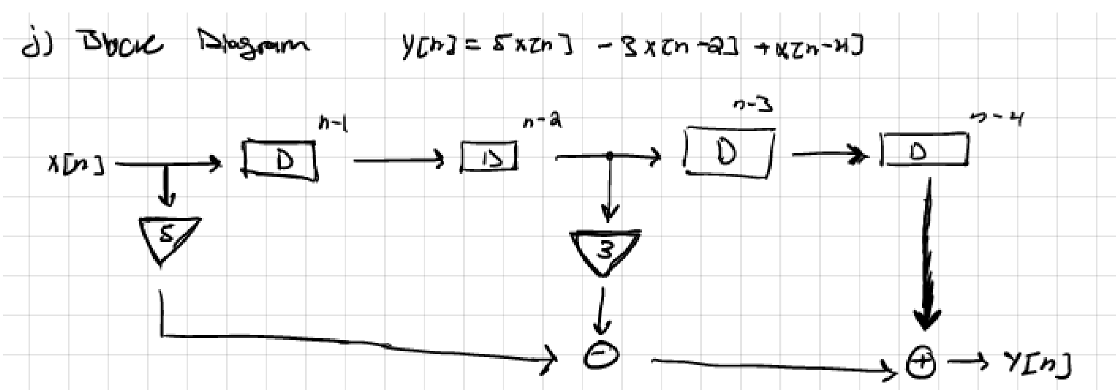

Figure 13: Enter Caption

2 Question 2

The following answers are based on the system described by the block diagram below:

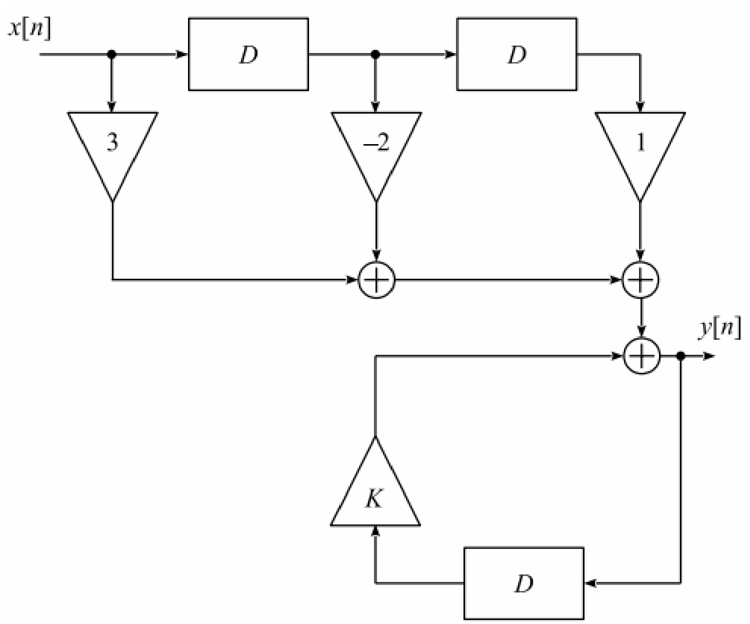

Figure 14: Block Diagram of System

2.1 (a)

The D-E that describes the block diagram is:

$$y[n] = 3x[n] - 2x[n-1] + x[n-2] + ky[n-1] \tag{2}$$

2.2 (b)

Given the feedback loop/recursive nature of the system, this is an IIR system.

2.3 (c)

c) unit impulse response $x[n] = \delta[n]$

for $n = \{0, \dots, 4\}$

$$h[n] = \sum_{k=1}^{N} a_k h[n-k] + \sum_{k=0}^{M} b_k \delta[n-k]$$

let $n=0$

$$y[0] = h[0] = 3\delta[0] - 2\delta[-1] + \delta[-2] + k\,y[-1]$$

$$\boxed{= 3}$$

let $n=1$

$$y[1] = h[1] = 3\delta[1] - 2\delta[0] + \delta[-1] + k\,y[0]$$

$$= 0 - 2 + 0 + 3k$$

$$= \boxed{-2+3k}$$

let $n=2$

$$y[2] = h[2] = 3\delta[2] - 2\delta[1] + \delta[0] + k\,y[1]$$

$$= 1 + k(-2+3k)$$

$$\boxed{= 1 - 2k + 3k^2}$$

$$y[n] = 3x[n] - 2x[n-1] + x[n-2] + k\,y[n-1]$$

let $n=3$

$$y[3] = h[3] = 3\delta[3] - 2\delta[2] + \delta[1] + k\,y[2]$$

$$= k(1 - 2k + 3k^2)$$

$$\boxed{= k - 2k^2 + 3k^3}$$

let $n=4$

$$y[4] = h[4] = 3\delta[4] - 2\delta[3] + \delta[2] + k\,y[3]$$

$$= k(k - 2k^2 + 3k^3)$$

$$= \boxed{k^2 - 2k^3 + 3k^4}$$

Figure 15: Enter Caption

2.4 (d)

```
%2_d
B = [3, -2, 1];
A = [1, -0.75];
N = 0:20;

h_d = impz(B,A,N);
```

Figure 16: Code to compute h[n] using impz() function with k = 0.75

```
h_d =

    3.0000
    0.2500
    1.1875
    0.8906
    0.6680
    0.5010
    0.3757
    0.2818
    0.2113
    0.1585
    0.1189
    0.0892
    0.0669
    0.0502
    0.0376
    0.0282
    0.0212
    0.0159
    0.0119
    0.0089
    0.0067
```

Figure 17: Output for h[n] using impz() function with k = 0.75

2.5 (e)

```
%2_e
x = [1 zeros(1,20)];
y_e = filter(B,A,x);
stem(N,y_e); %plotting Response
```

Figure 18: Code to compute response of system with filter() function

```
y_e =

  Columns 1 through 4

    3.0000    0.2500    1.1875    0.8906

  Columns 5 through 8

    0.6680    0.5010    0.3757    0.2818

  Columns 9 through 12

    0.2113    0.1585    0.1189    0.0892

  Columns 13 through 16

    0.0669    0.0502    0.0376    0.0282

  Columns 17 through 20

    0.0212    0.0159    0.0119    0.0089

  Column 21

    0.0067
```

Figure 19: Output

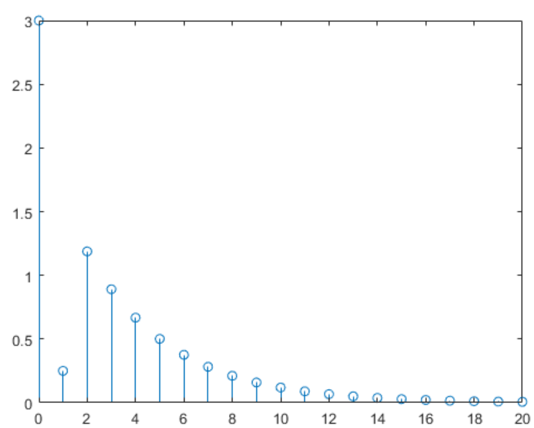

Figure 20: Plot of output of system

2.6 (f)

```
%2_f
B = [3, -2, 1];
A = [1, 0.75];
N = 0:20;

h_f = impz(B,A,N);
stem(N,h_f);
```

Figure 21: Code to compute response when K = -0.75

```
h_f =

    3.0000
   -4.2500
    4.1875
   -3.1406
    2.3555
   -1.7666
    1.3250
   -0.9937
    0.7453
   -0.5590
    0.4192
   -0.3144
    0.2358
   -0.1769
    0.1326
   -0.0995
    0.0746
   -0.0560
    0.0420
   -0.0315
    0.0236
```

Figure 22: Response of System when k = -0.75

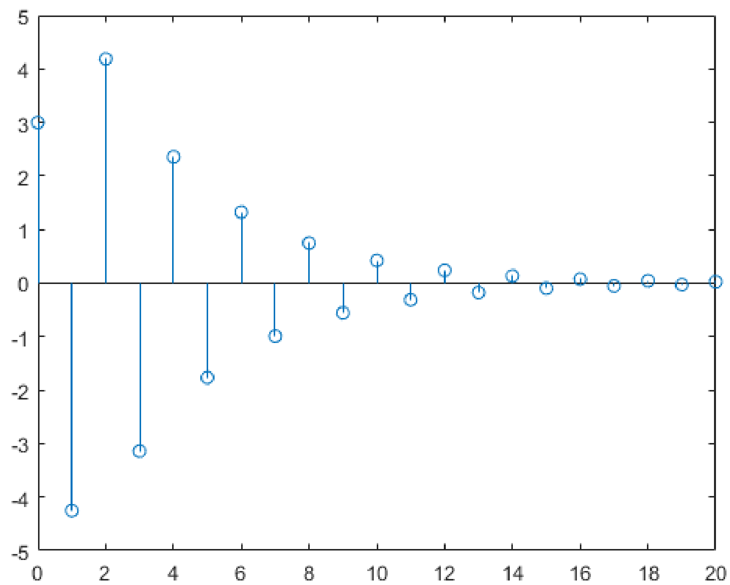

Figure 23: Plot of Response of System when k = -0.75

In this case and in part (d), both systems begin at 3 and converge to 0; in this case the systems oscillates to steady state.

2.7 (g)

```
%2_g
B = [3, -2, 1];
A = [1, -1.2];
N = 0:20;

h_g = impz(B,A,N);
stem(N,h_g);
```

Figure 24: Code to compute response when k=1.2

```
h_g =

    3.0000
    1.6000
    2.9200
    3.5040
    4.2048
    5.0458
    6.0549
    7.2659
    8.7191
   10.4629
   12.5555
   15.0666
   18.0799
   21.6958
   26.0350
   31.2420
   37.4904
   44.9885
   53.9862
   64.7834
   77.7401
```

Figure 25: Output of system when k = 1.2

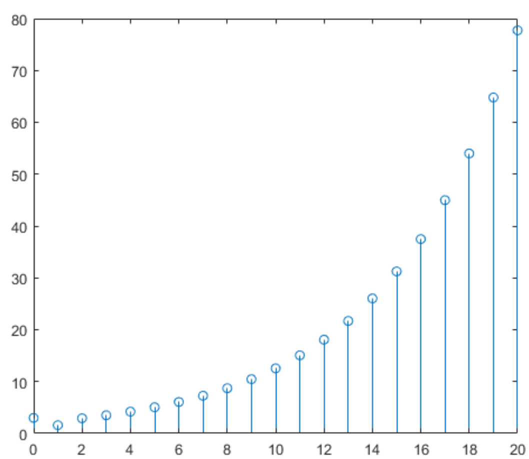

Figure 26: Plot of System's response when k = 1.2

2.8 (h)

```
%2_h
B = [3, -2, 1];
A = [1, -0.75];


x_h = ones(1,length(N));

y_h = filter(B,A,x_h);


stem(N,y_h);
```

Figure 27: Code to compute unit step response when k = 0.75 using filter() function

```
y_h =

  Columns 1 through 5

    3.0000    3.2500    4.4375    5.3281    5.9961

  Columns 6 through 10

    6.4971    6.8728    7.1546    7.3660    7.5245

  Columns 11 through 15

    7.6433    7.7325    7.7994    7.8495    7.8872

  Columns 16 through 20

    7.9154    7.9365    7.9524    7.9643    7.9732

  Column 21

    7.9799
```

Figure 28: Unit step response of system when k = 0.75

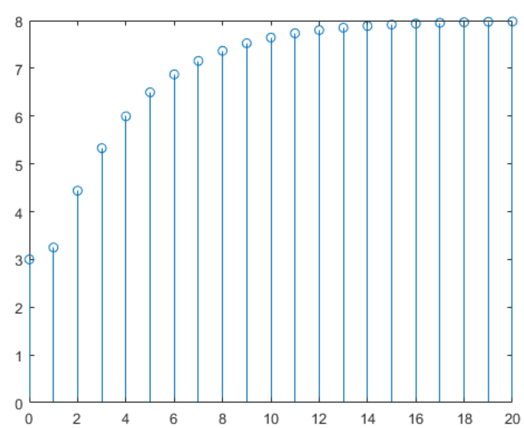

Figure 29: Plot of unit step response

2.9 (i)

The system is stable when k = 0.75. In the case of the unit impulse response, the system converges to 0. In the case of the unit step response, the system converges to 8.

Put another way, there is a Bound Output for the Bound Inputs $\rightarrow$ BIBO.

2.10 (j)

The system is not stable when k = 1.2. Given Figures 25 & 26, the system clearly increases exponentially without end: there is no bound output for the input. BIBO fails.