## Table of Contents

# Niruyan Rakulan 214343438

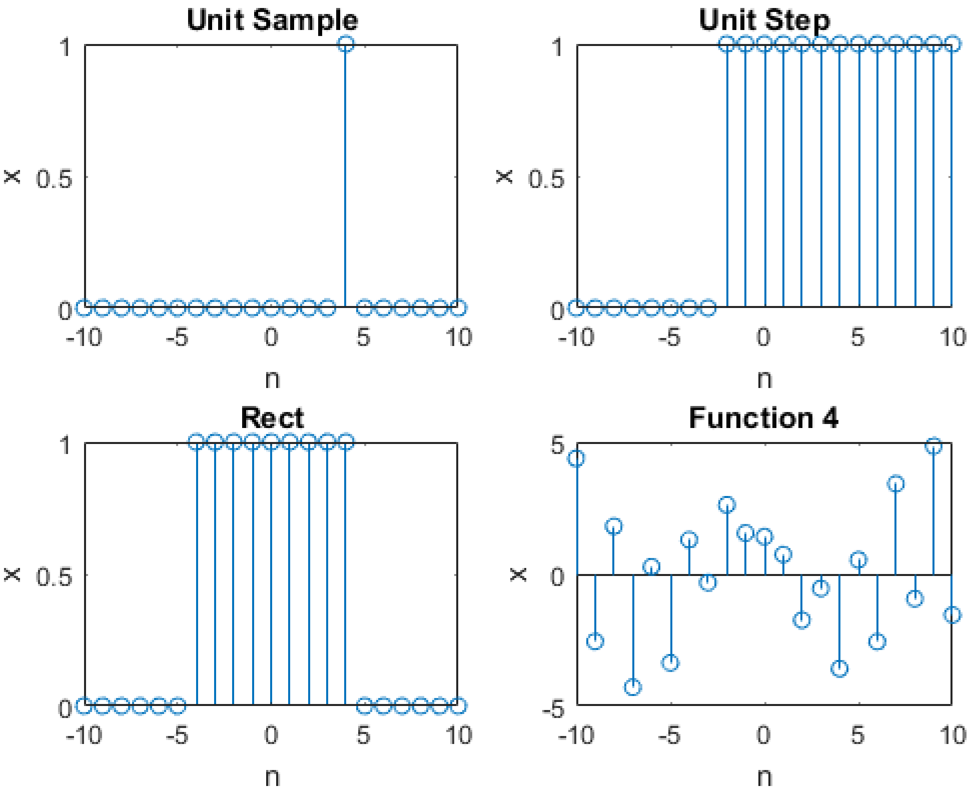
```
%Introduction: The purpose of the lab was to understand the basics
%of making discrete functions in matlab, and how they can be
 manipulated
%(shifting, interpolation, decimation).
%Equipment: MATLAB, PC
```

# Q1

# Q2

```
%plot unit sample fucntion from -10 to 10 when n0=4
clear all;
close all;
n1=-10;
n2=10;
n0=4;

subplot(2,2,1);
[x1,n]=unitsample(n0,n1,n2);
stem(n,x1)
xlabel('n')
ylabel('x')
title('Unit Sample')

%plot unit step
n0=-2;
subplot(2,2,2);
x2=unitstep(n0,n1,n2);
stem(n,x2)
xlabel('n')
ylabel('x')
title('Unit Step')

%plot rect(n/10)
```

```
start=floor((10-1)/2)*-1;
last=start*-1+1;
x3=unitstep(start,n1,n2);
x4=unitstep(last,n1,n2);
x5=x3-x4;
subplot(2,2,3);
stem(n,x5)
xlabel('n')
ylabel('x')
title('Rect')

%plot cos
x6=2*cos(0.2*pi.*n+pi/4)+3*sin(3.*n);
subplot(2,2,4);
stem(n,x6)
xlabel('n')
ylabel('x')
title('Function 4')
%The function is not peridoic. Graphicaly it can be determined since
 the
%pattern does not repeat.Algebraically it can be determined that the
 sin
%portion of the fucntion is not periodic since w/2=3/2pi which is not
 a
%rational number therfore the whole function is not periodic.
```


Unit Sample

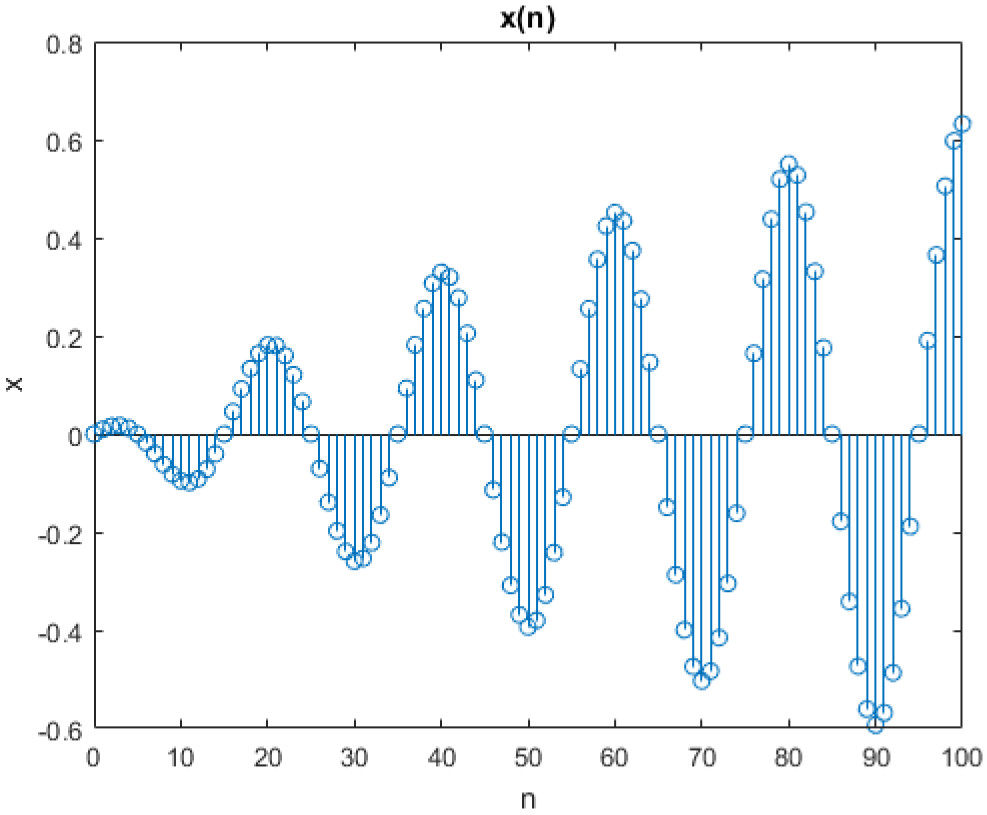
Unit Step

Rect

Function 4

# Q3

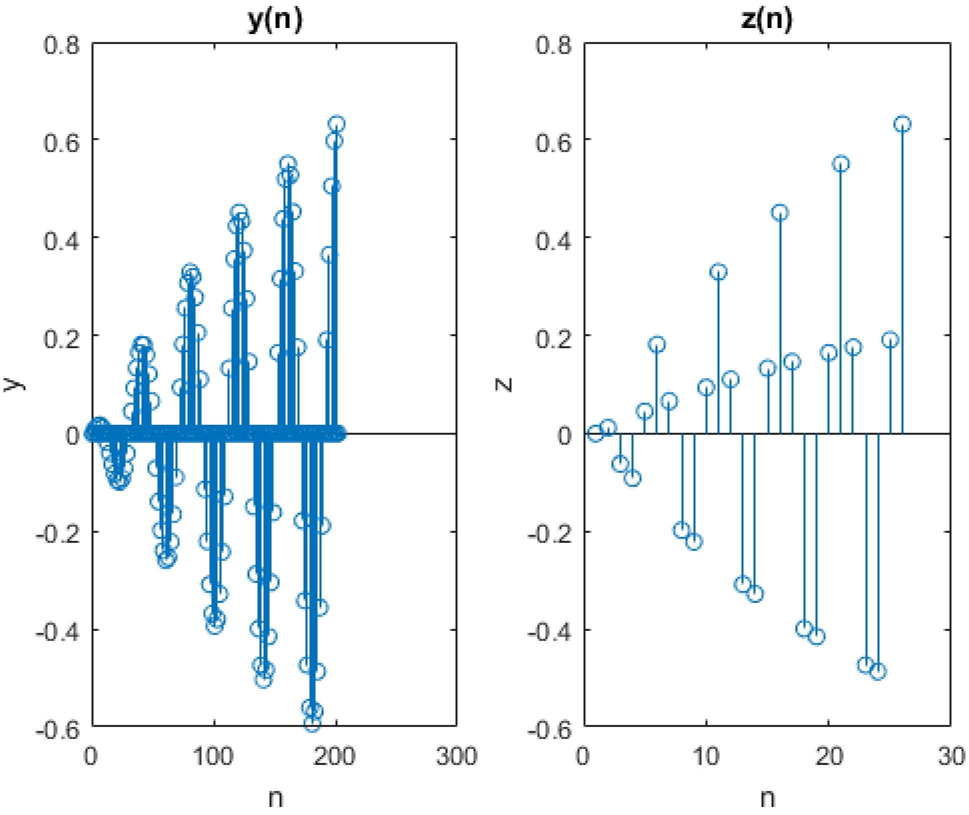
```
%3.1
%Plot x(n)
clear all;
close all;
n=0:100;
x=(1-exp(-0.01.*n)).*cos(pi.*n/10);
stem(n,x);
xlabel('n')
ylabel('x')
title('x(n)')
```



# 3.2

```
%Plot y(n)=x(n/2)
clear all;
n=0:100;
x=(1-exp(-0.01.*n)).*cos(pi.*n/10);
y=upsample(x,2);
subplot(1,2,1);
k= length(y);
stem((1:k),y);
xlabel('n')
ylabel('y')
```

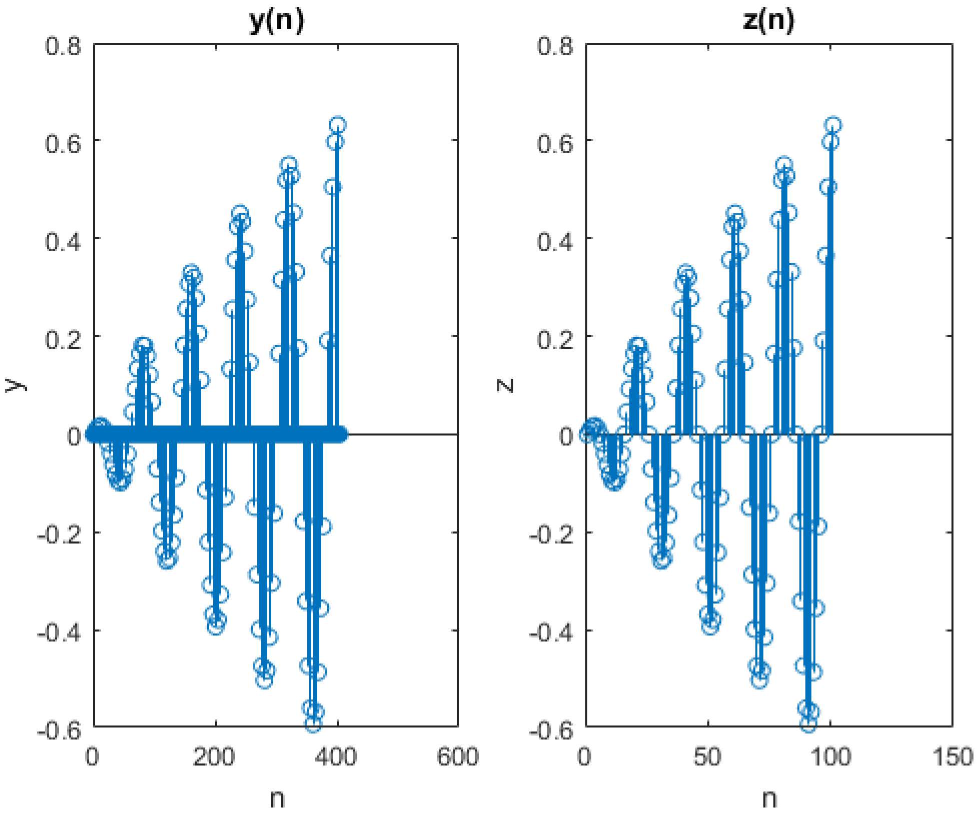
```
title('y(n)')

%Plot z(n)=x(4n)
clear all;
n=0:100;
x=(1-exp(-0.01.*n)).*cos(pi.*n/10);
z=downsample(x,4);
subplot(1,2,2);
k=length(z);
stem((1:k),z);
xlabel('n')
ylabel('z')
title('z(n)')
```



# 3.3

```
%Plot y(n)=x(n/4)
clear all;
n=0:100;
x=(1-exp(-0.01.*n)).*cos(pi.*n/10);
y=upsample(x,4);
k=length(y);
subplot(1,2,1);
stem((1:k),y);
xlabel('n')
ylabel('y')
```

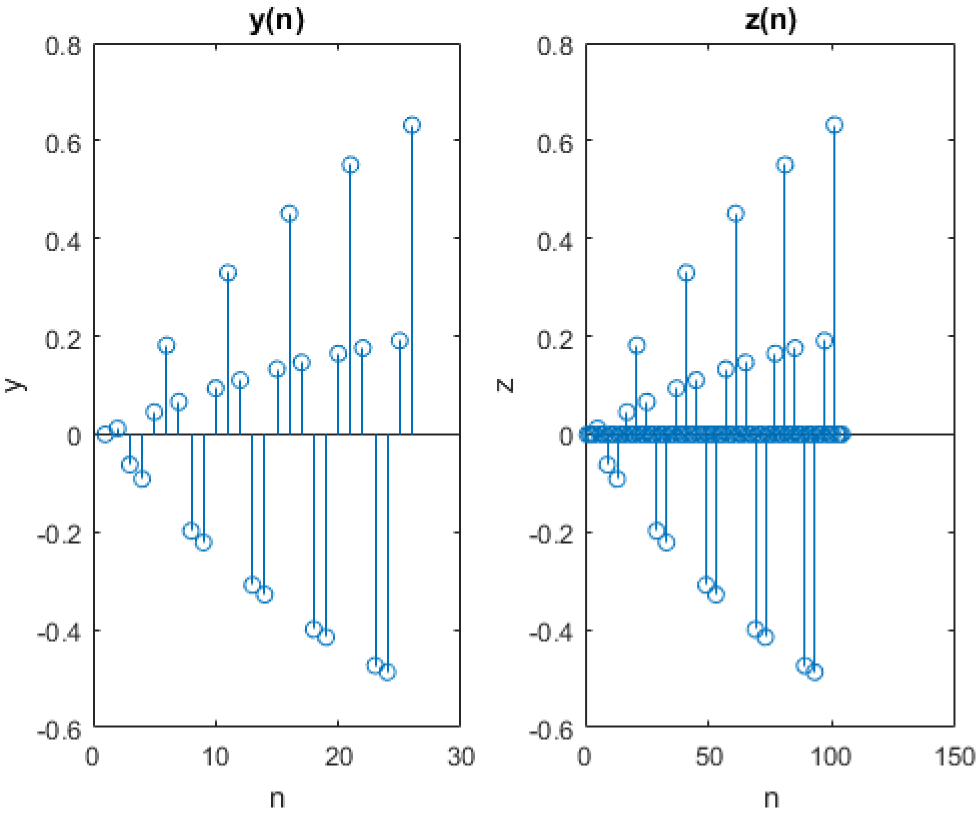
```
title('y(n)')

%Plot z(n)=y(4n)
z=downsample(y,4);
k=length(z);
subplot(1,2,2);
stem((1:k),z);
xlabel('n')
ylabel('z')
title('z(n)')

%x(n)=z(n). This is because x(n) is first upsampled(y(n)), and than
 down sampled (z(n)).
%This returns the origianl function x(n).
```



# 3.4

```
%plot y(n)=x(4n)
clear all;
n=0:100;
x=(1-exp(-0.01.*n)).*cos(pi.*n/10);
y=downsample(x,4);
k=length(y);
subplot(1,2,1);
stem((1:k),y);
xlabel('n')
```

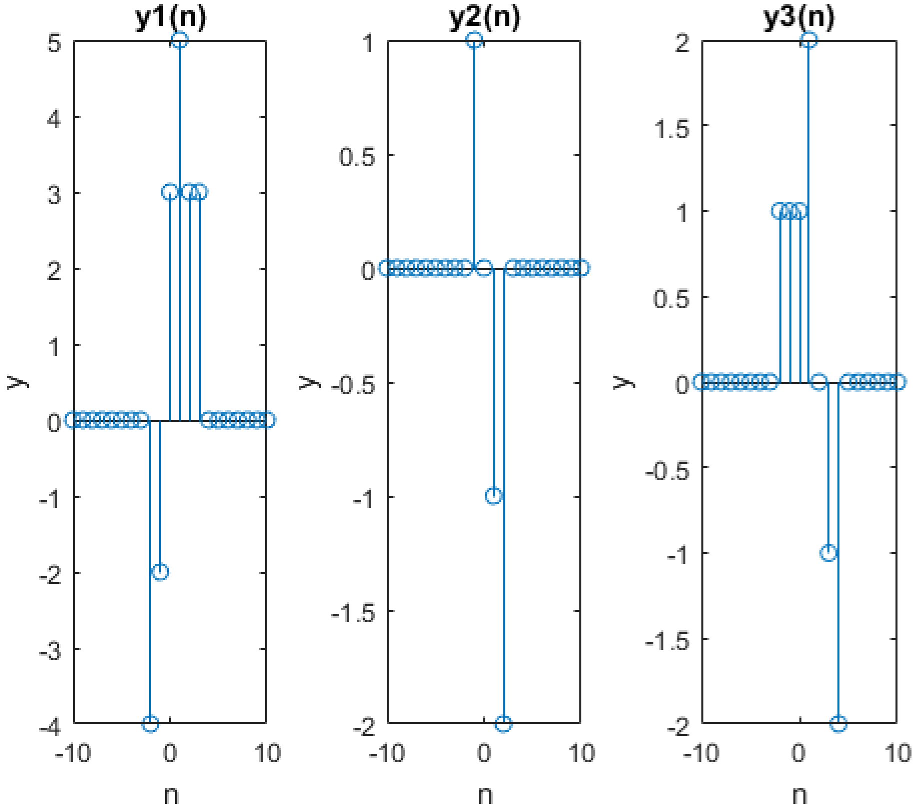
```
ylabel('y')
title('y(n)')

%plot z(n)=y(n/4)
z=upsample(y,4);
subplot(1,2,2);
k=length(z);
stem((1:k),z);
xlabel('n')
ylabel('z')
title('z(n)')
%x(n)!=z(n). This is because the signal is first downsampled (y(n)),
 and
%than upsampled (z(n). Information is lost when it is downsampled and
 when
%it is upsampled again, information is missing.
```



# 4

```
clear all;
n1=-10;
n2=10;
n0=-2;
[xa,n]=unitstep(n0,n1,n2);
n0=2;
[xb,n]=unitstep(n0 ,n1,n2);
```

```
x1=n.*(xa-xb);


n0=0;
[xc,n]=unitstep(n0,n1,n2);
n0=4;
[xd,n]=unitstep(n0,n1,n2);
x2= xc-xd;

y1= 2*x1+3*x2;
y2= fliplr(x1);

y3_a= zeros(size(n));
x1inv=fliplr(x1);
y3_a(3:end)= x1inv(1:end-2);


y3_b= zeros(size(n));
y3_b(1:end-2)= x2(3:end);

y3=y3_a+y3_b;
%plot y1
subplot(1,3,1);
stem(n,y1);
xlabel('n')
ylabel('y')
title('y1(n)')
%plot y2
subplot(1,3,2);
stem(n,y2);
xlabel('n')
ylabel('y')
title('y2(n)')
%plot y3
subplot(1,3,3);
stem(n,y3);
xlabel('n')
ylabel('y')
title('y3(n)')
```



# Conclusion

```
%The lab went as planned. The lab displays the fundamentals of DT
%functions.
```

*Published with MATLAB® R2017a*