Z Transform

1. Discrete Z Transform

Code:

```
x = input("Enter the Input Sequence : ");
l = length(x);
X = 0;
z = sym('z');

for i = 0 : (l - 1)
    X = X + x(i + 1) * z ^ (-i);
end

disp('Z Transform of Given Input Sequence is ');
disp(X);
```

Result:

>> discrete

Enter the Input Sequence : [2 3 4 5]

Z Transform of Given Input Sequence is

3/z + 4/z^2 + 5/z^3 + 2

2. Finding Zeros and Poles of a Transfer Function

Code:

```
disp('For Plotting Zeros and Poles');
b = input('Enter Numerator Polynomial Coefficients : ');
a = input('Enter Denominator Polynomial Coefficient : ');

[z, p, k] = residuez(b, a);
zplane(z, p);
disp('Zeros : ');
disp(z);
disp('Poles : ');
disp(p);
disp('K : ');
disp(k);
```

Result:

>> zerosPoles

For Plotting Zeros and Poles

Enter Numerator Polynomial Coefficients : [-4 8]

Enter Denominator Polynomial Coefficient : [1 6 8]

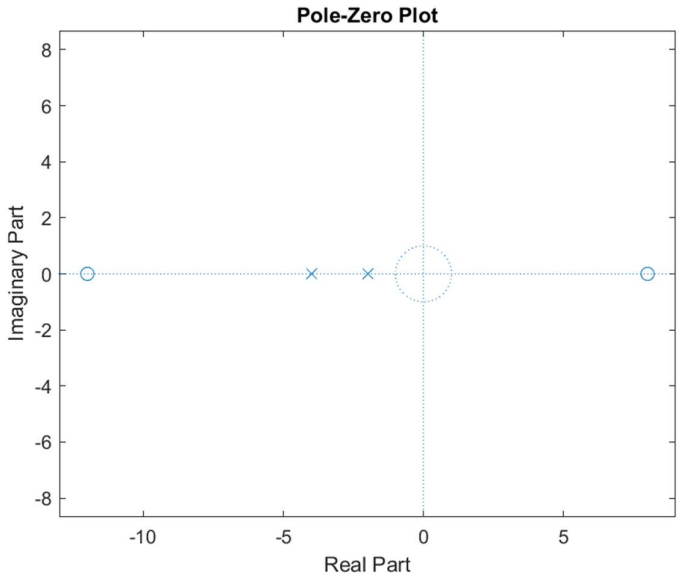
Zeros :

-12 8

Poles :

-4 -2

K :

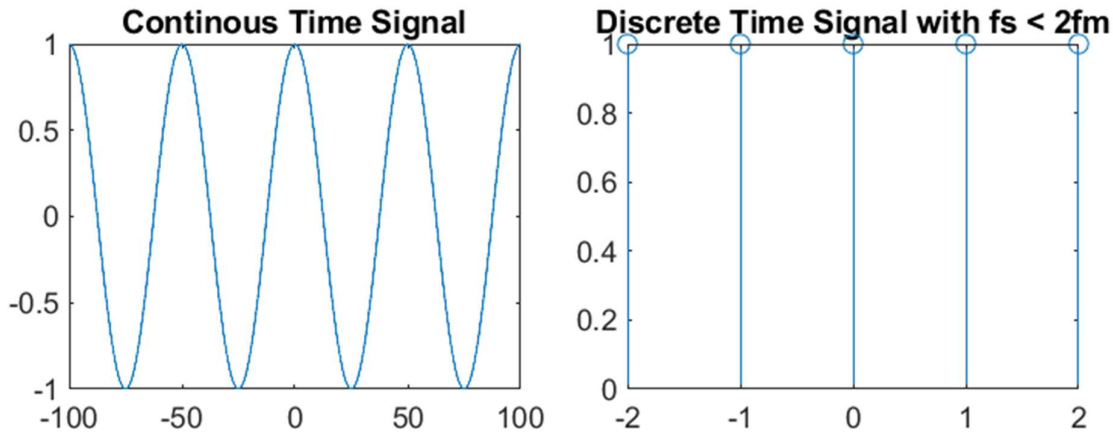


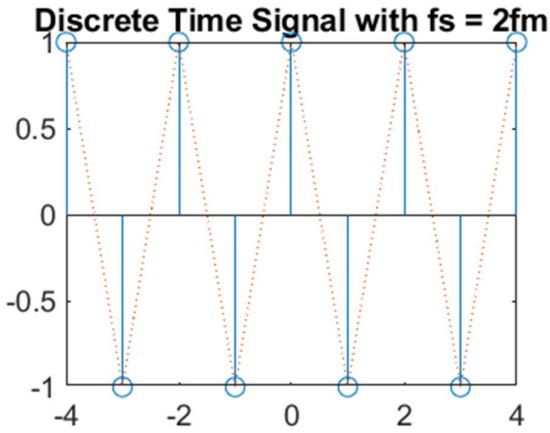
Sampling Theorem

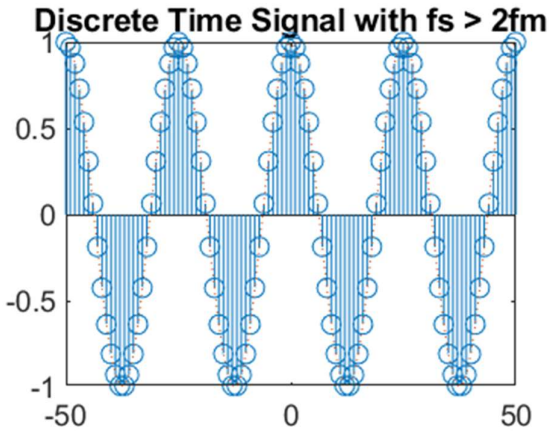
Code:

```
t = -100 : 0.01 : 100;
fm = 0.02;
x = cos(2 * pi * t * fm);
subplot(221);
plot(t, x);
title('Continous Time Signal');

fs1 = 0.02;
n = -2 : 2;
x1 = cos(2 * pi * fm * n / fs1);
subplot(222);
stem(n, x1);
title('Discrete Time Signal with fs < 2fm');

fs2 = 0.04;
n1= -4 : 4;
x2 = cos(2 * pi * f m * n1 / fs2);
subplot(223);
stem(n1, x2);
title('Discrete Time Signal with fs = 2fm');
hold on
subplot(223);
plot(n1, x2, ':');

fs3 = 0.5;
n2 = -50 : 50;
x3 = cos(2 * pi * fm * n2 / fs3);
subplot(224);
stem(n2, x3);
title('Discrete Time Signal with fs > 2fm');
hold on
subplot(224);
plot(n2, x3, ':');
```

Result:




Discrete Time Signal with fs = 2fm
1
0.5
0
-0.5
-1
-4
-2
0
2
4


Discrete Time Signal with fs > 2fm
1
0.5
0
-0.5
-1
-50
0
50