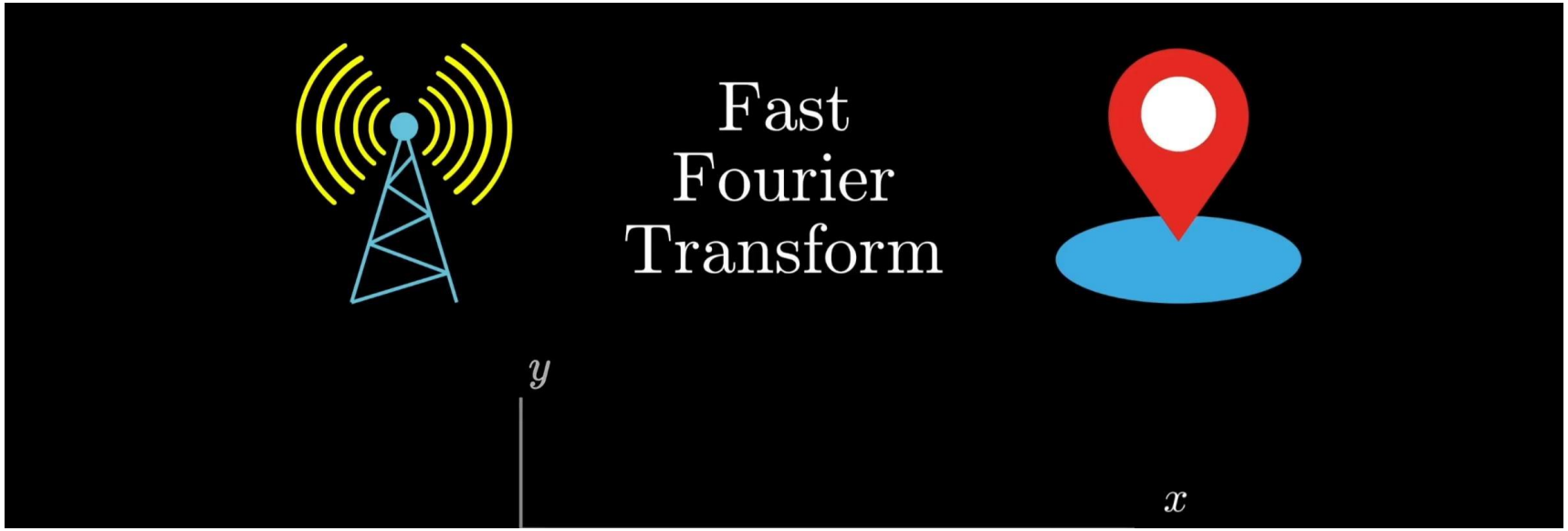
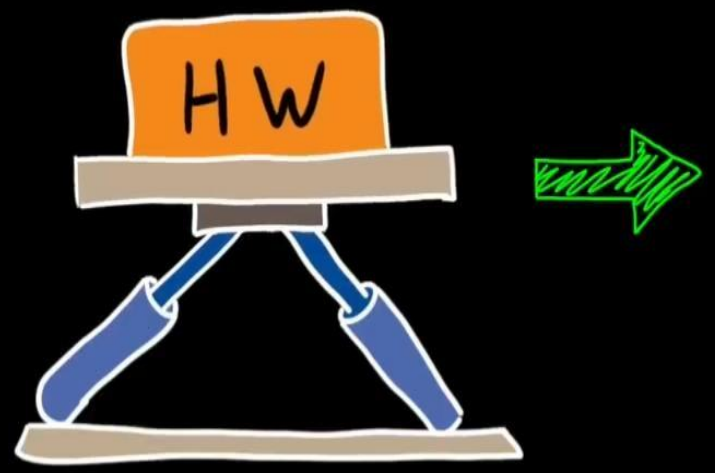
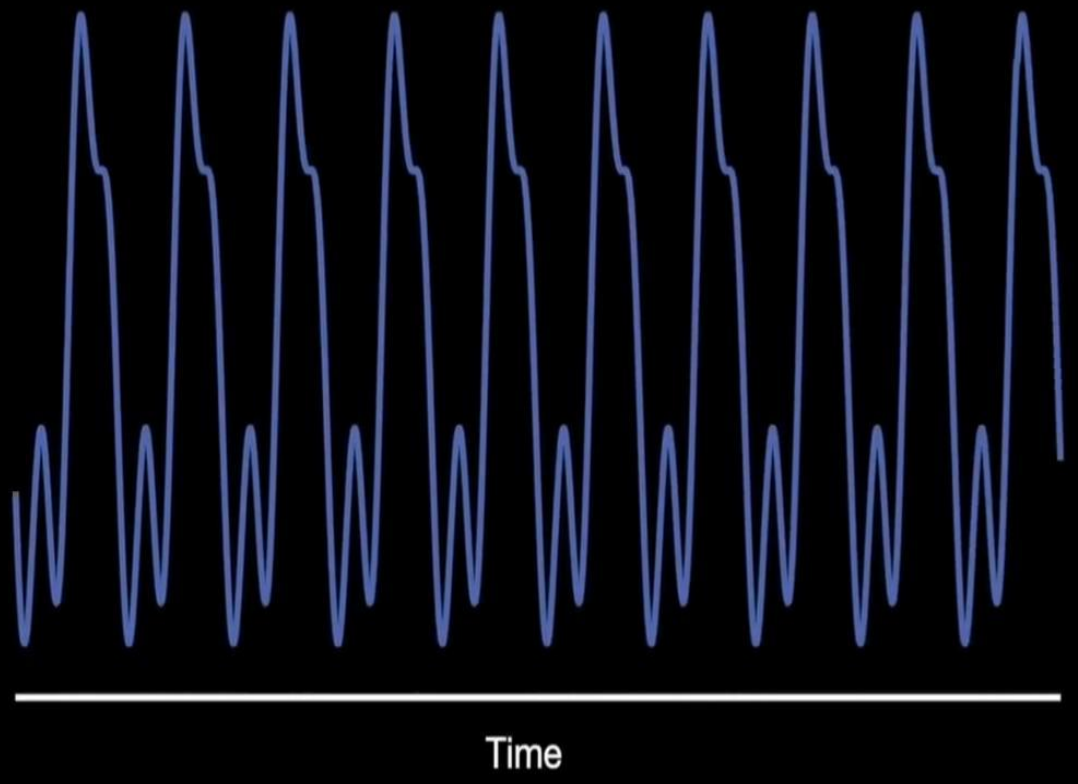

Fast
Fourier
Transform
y
x

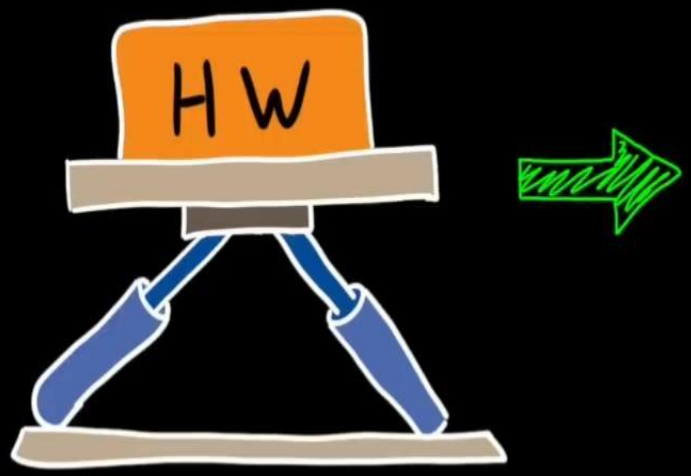
Accelerometer Data

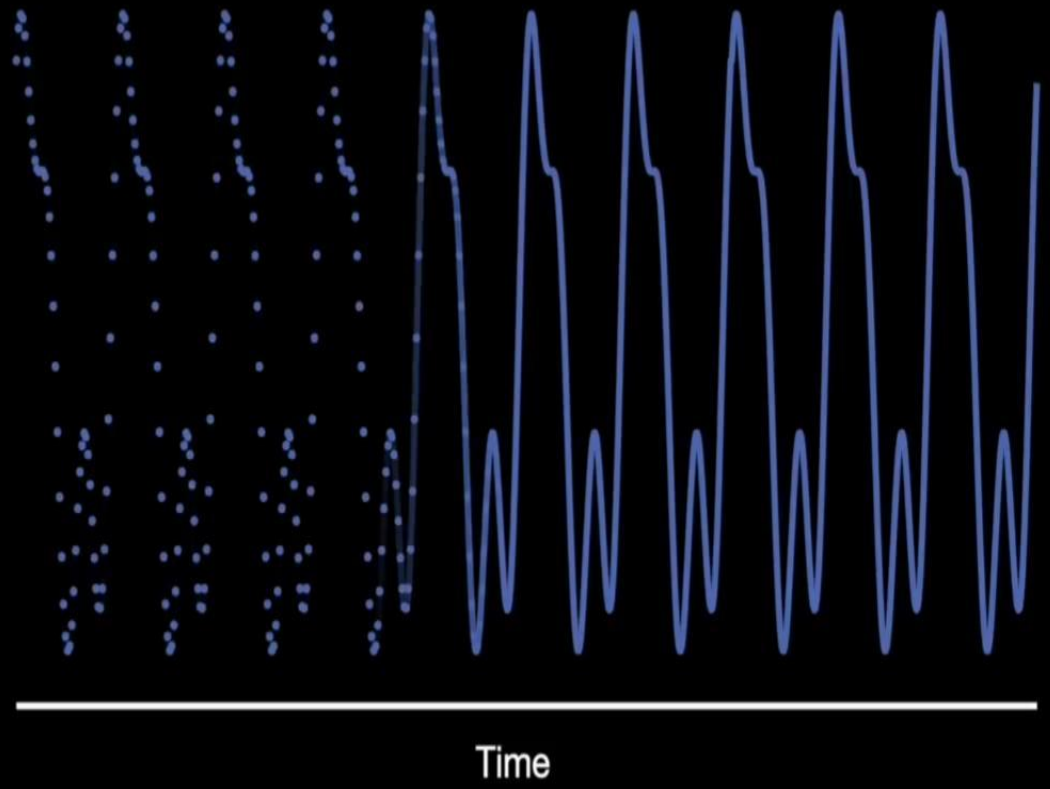




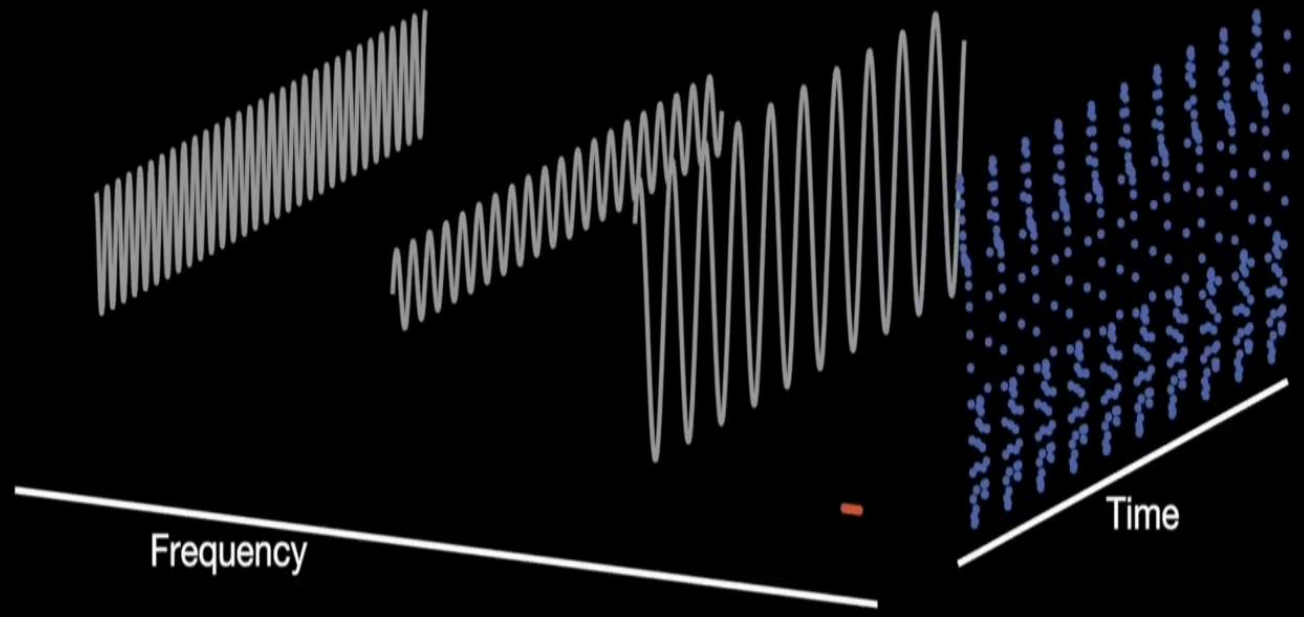
Accelerometer Data

finite and sampled



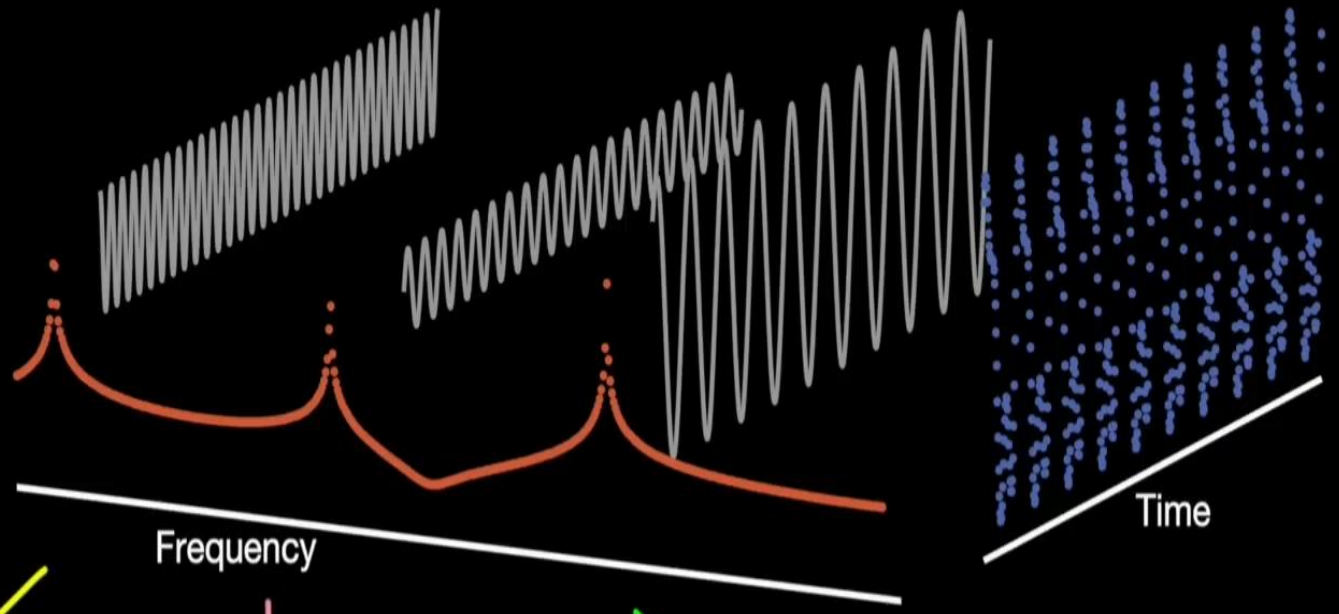


HW


Frequency
Time

Discrete Fourier Transform (DFT)

F.



Amplitude spectrum

Power spectrum

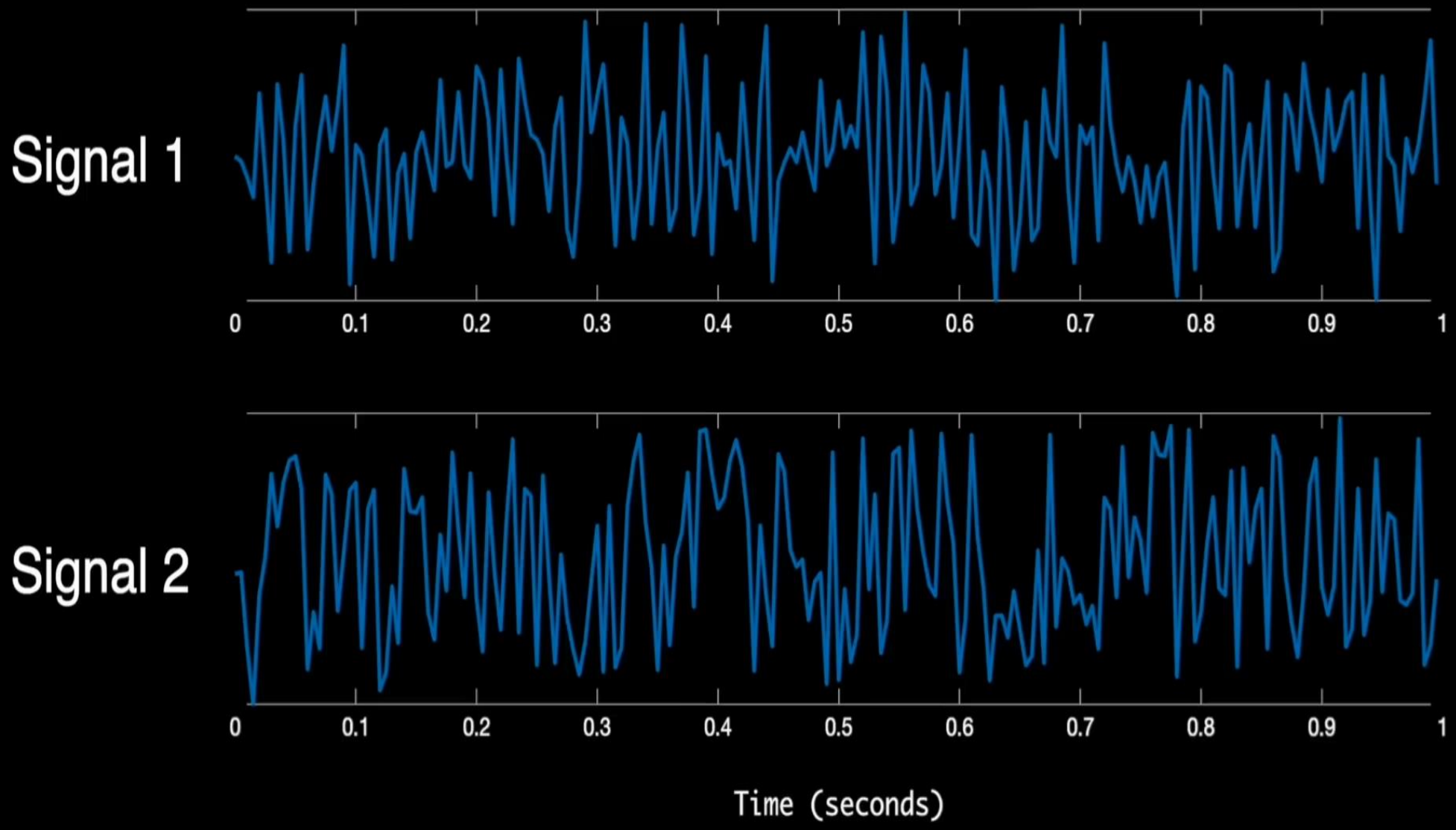
Power spectral density

Why do we use the discrete Fourier transform?

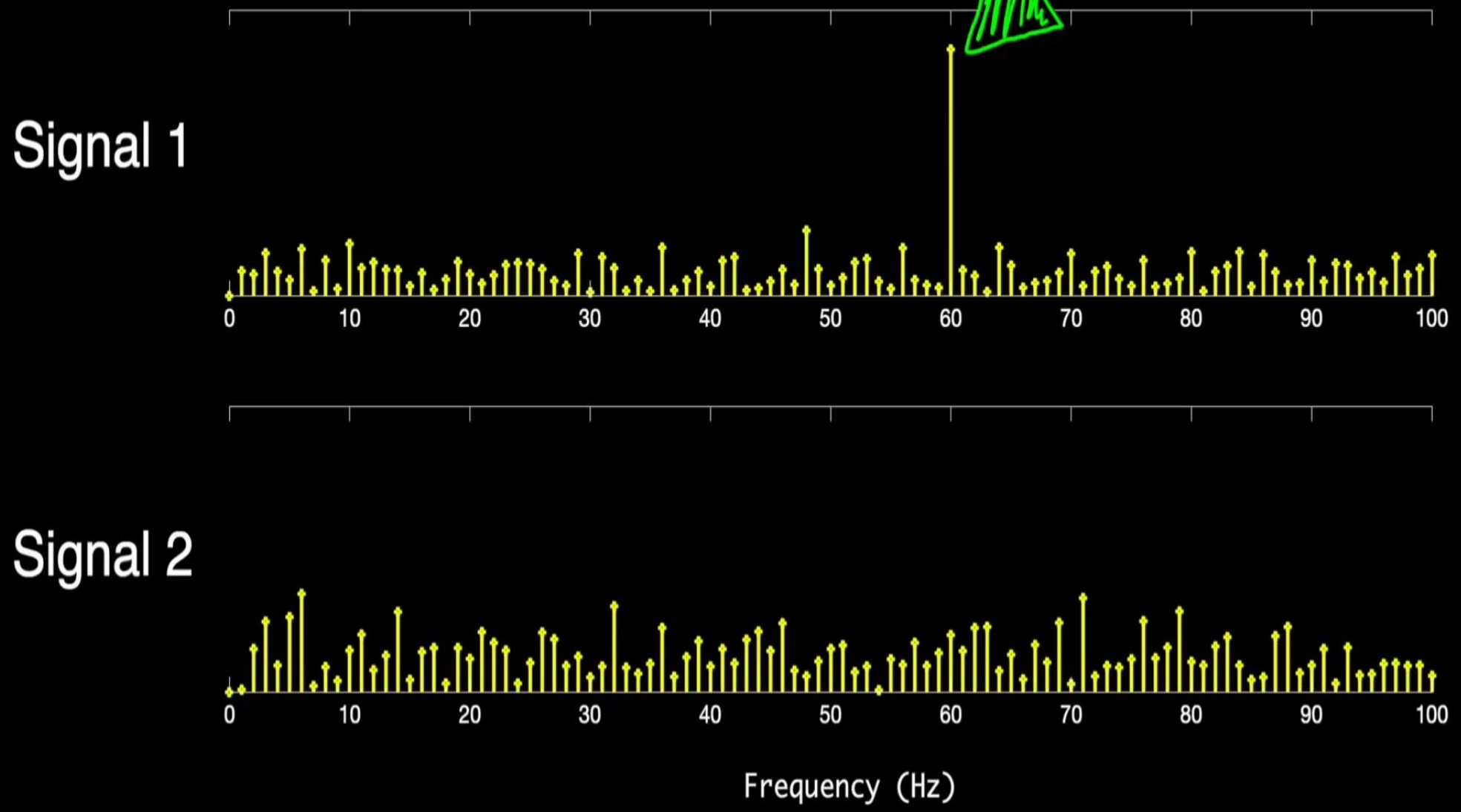
Time → DFT → Frequency

Spatial → DFT

Time Domain


Signal 1
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Signal 2
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
Time (seconds)

Frequency Domain



Discrete Fourier Transform

$$X[k] = \sum_{n=0}^{N-1} x_n \cdot e^{-\frac{i2\pi}{N}kn}$$

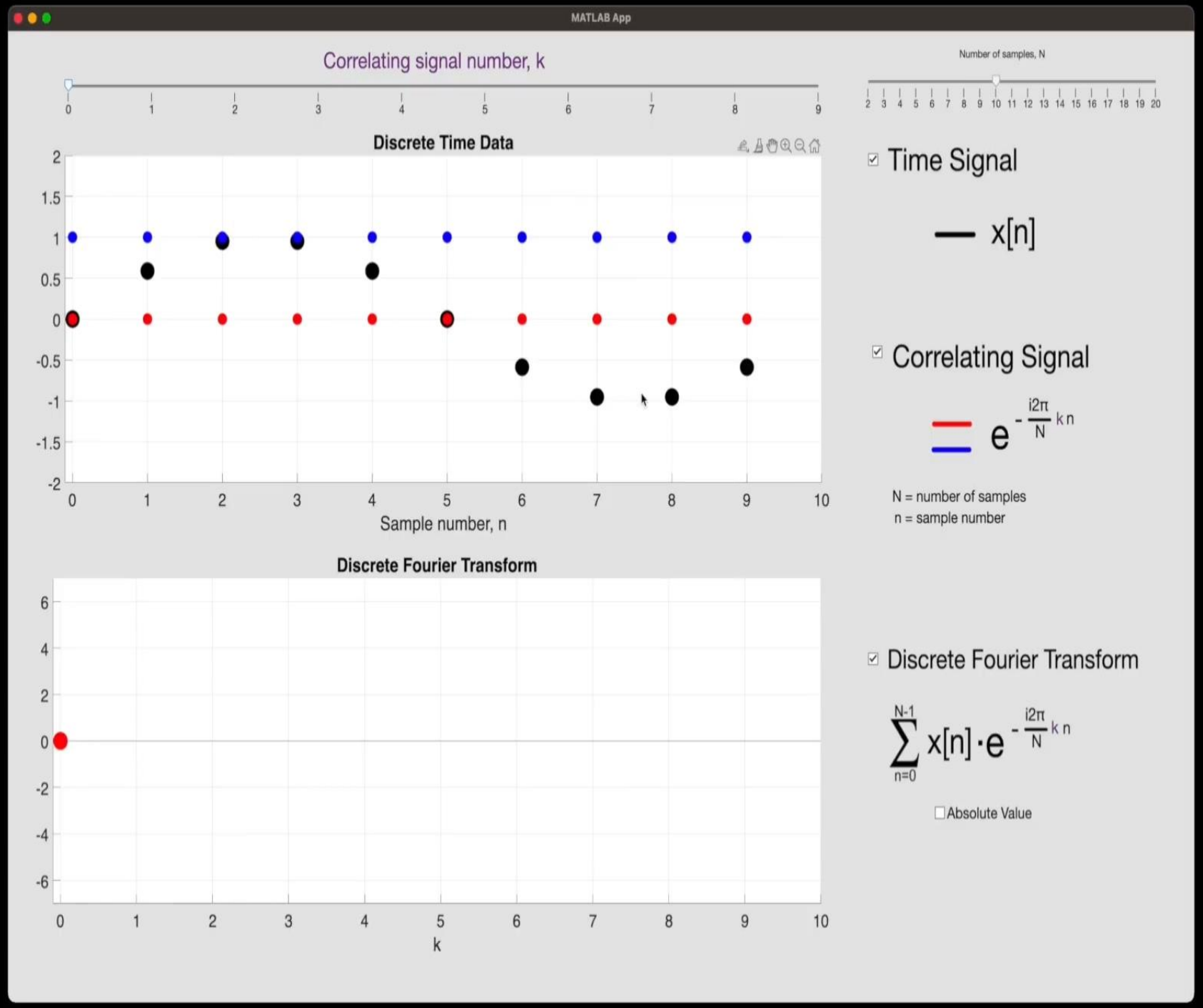
Discrete Fourier Transform

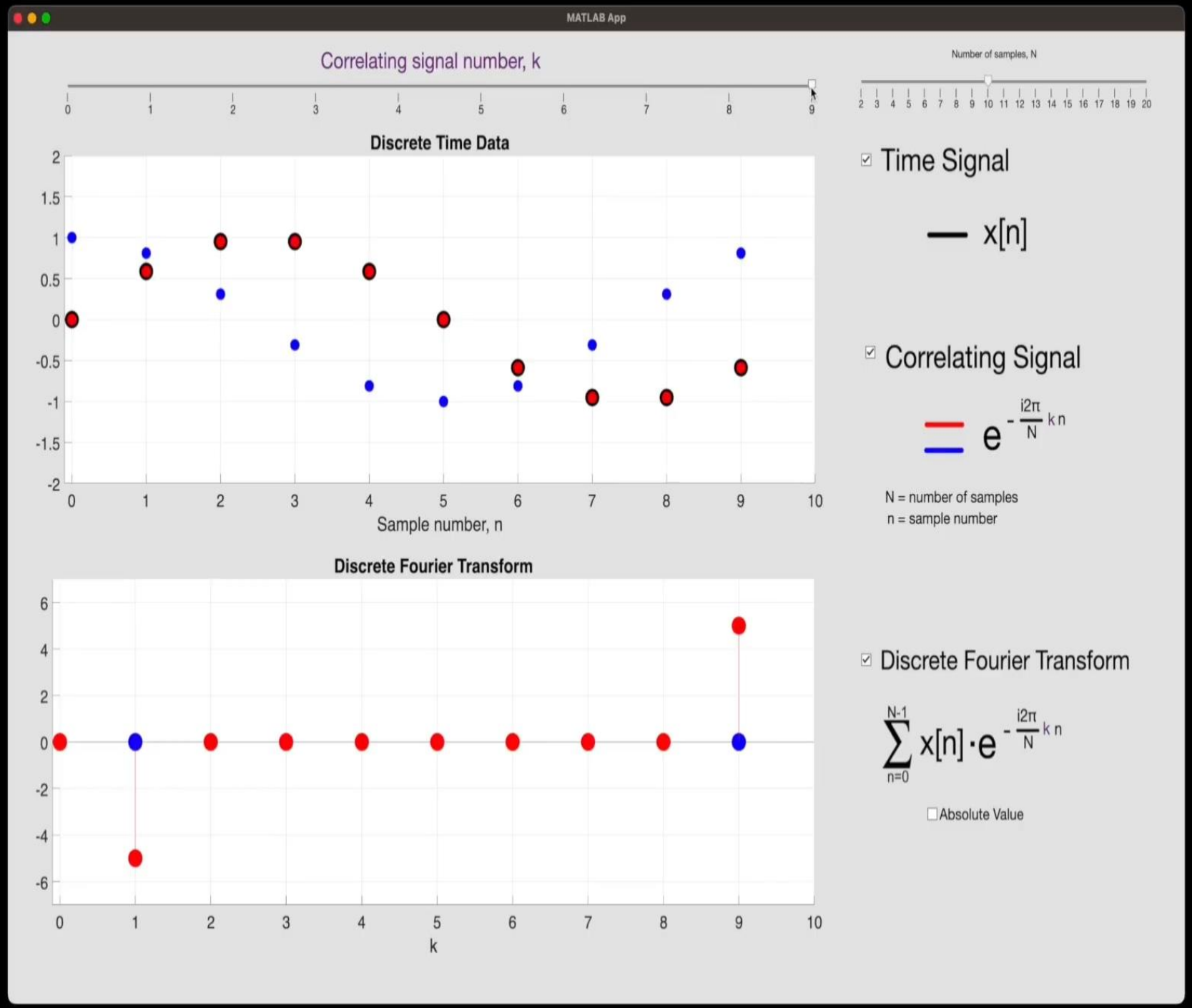
$$X[k] = \sum_{n=0}^{N-1} x_n \cdot e^{-\frac{i2\pi}{N}kn}$$

We're creating sines and cosines of different frequencies

$$e^{in} = \cos(n) + i\sin(n)$$

Discrete-time signal


MATLAB App
Correlating signal number, k
0 1 2 3 4 5 6 7 8 9
Number of samples, N
2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
Discrete Time Data
2 1.5 1 0.5 0 -0.5 -1 -1.5 -2
0 1 2 3 4 5 6 7 8 9 10
Sample number, n
Time Signal
x[n]
Correlating Signal
$e^{-\frac{i2\pi}{N}kn}$
N = number of samples
n = sample number
Discrete Fourier Transform
6 4 2 0 -2 -4 -6
0 1 2 3 4 5 6 7 8 9 10
k
Discrete Fourier Transform
$\sum_{n=0}^{N-1} x[n] \cdot e^{-\frac{i2\pi}{N}kn}$
Absolute Value


MATLAB App
Correlating signal number, k
Number of samples, N
Discrete Time Data
Sample number, n
Time Signal
x[n]
Correlating Signal
N = number of samples
n = sample number
Discrete Fourier Transform
k
Discrete Fourier Transform
Absolute Value

Discrete Fourier Transform

Matrix Multiplication

$$X[k] = \sum_{n=0}^{N-1} x_n \cdot e^{-\frac{i2\pi}{N}kn}$$

$$\underbrace{\begin{bmatrix} X_0 & X_1 & X_2 & \cdots & X_{N-1} \end{bmatrix}}_{1 \times N} = \underbrace{\begin{bmatrix} x_0 & x_1 & x_2 & \cdots & x_{N-1} \end{bmatrix}}_{1 \times N} \cdot \underbrace{\begin{bmatrix} k=0 & k=1 & k=2 & \cdots & k=N-1 \end{bmatrix}}_{N \times N}$$

Discrete Fourier Transform

Matrix Multiplication

$$X[k] = \sum_{n=0}^{N-1} x_n \cdot e^{-\frac{i2\pi}{N}kn}$$

$$\underbrace{\begin{bmatrix} X_0 & X_1 & X_2 & \cdots & X_{N-1} \end{bmatrix}}_{1 \times N} = \underbrace{\begin{bmatrix} x_0 & x_1 & x_2 & \cdots & x_{N-1} \end{bmatrix}}_{1 \times N} \cdot \underbrace{\begin{bmatrix} 1 & K=1 & K=2 & \cdots & K=N-1 \\ x_0 & \downarrow & \downarrow & & \downarrow \\ x_1 & \downarrow & \downarrow & & \downarrow \\ x_2 & \downarrow & \downarrow & & \downarrow \\ \vdots & \downarrow & \downarrow & & \downarrow \\ x_{N-1} & \downarrow & \downarrow & & \downarrow \end{bmatrix}}_{N \times N}$$

Discrete Fourier Transform

Matrix Multiplication

$$X[k] = \sum_{n=0}^{N-1} x_n \cdot e^{-\frac{i2\pi}{N}kn}$$

$$\underbrace{\begin{bmatrix} X_0 & X_1 & X_2 & \cdots & X_{N-1} \end{bmatrix}}_{\times N} = \underbrace{\begin{bmatrix} x_0 & x_1 & x_2 & \cdots & x_{N-1} \end{bmatrix}}_{1 \times N} \cdot \underbrace{\begin{bmatrix} k=0 & \begin{bmatrix} x_0 & x_1 & x_2 & \cdots & x_{N-1} \end{bmatrix} & k=2 & \cdots & k=N-1 \end{bmatrix}}_{N \times N}$$

Discrete Fourier Transform

Matrix Multiplication

$$X[k] = \sum_{n=0}^{N-1} x_n \cdot e^{-\frac{i2\pi}{N}kn}$$

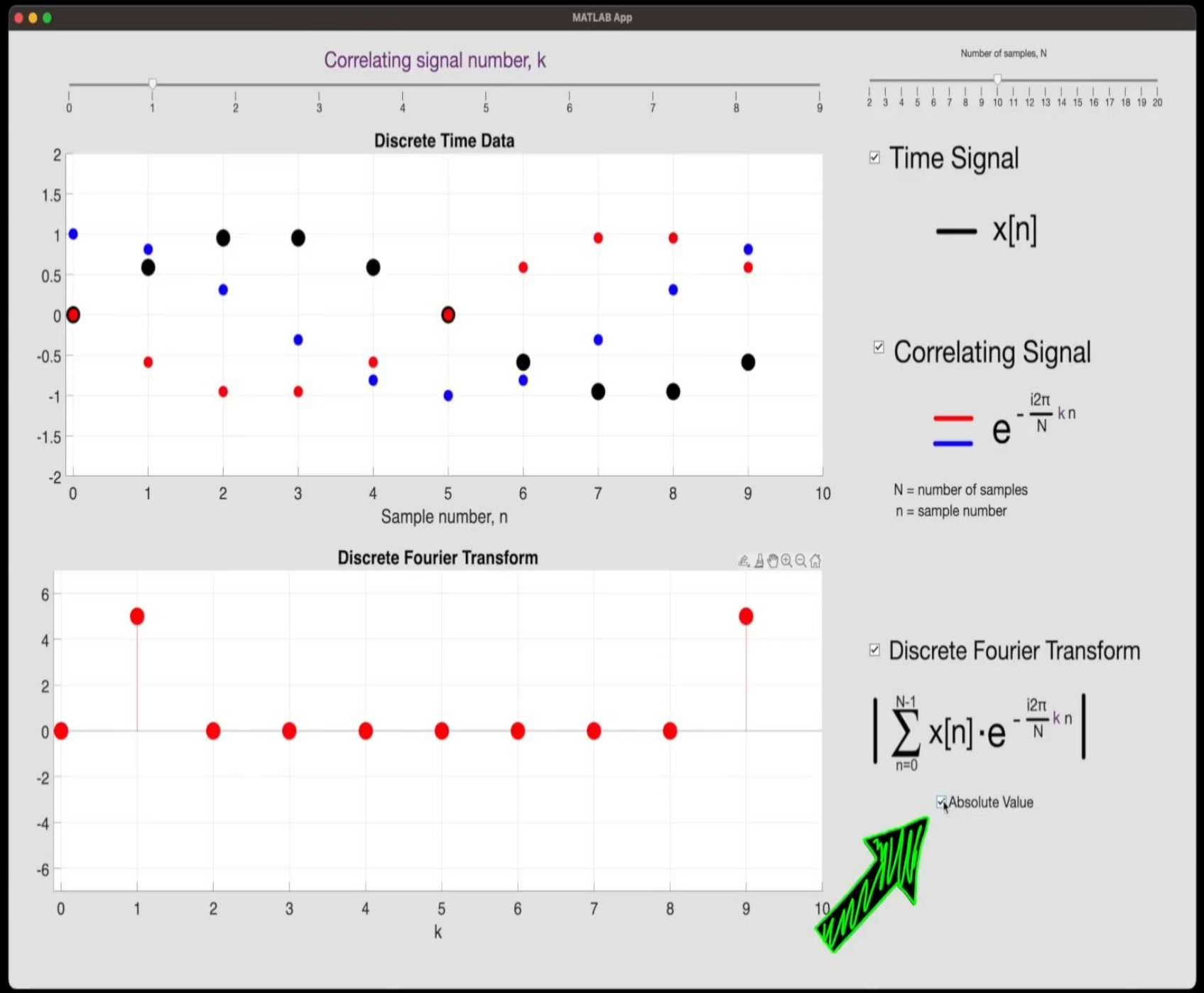
Rotates x_n into X_k

$$\overset{1 \times N}{\begin{bmatrix} X_0 & X_1 & X_2 & \cdots & X_{N-1} \end{bmatrix}} = \overset{1 \times N}{\begin{bmatrix} x_0 & x_1 & x_2 & \cdots & x_{N-1} \end{bmatrix}} \cdot \overset{N \times N}{\begin{bmatrix} k=0 & k=1 & k=2 & \cdots & k=N-1 \end{bmatrix}}$$

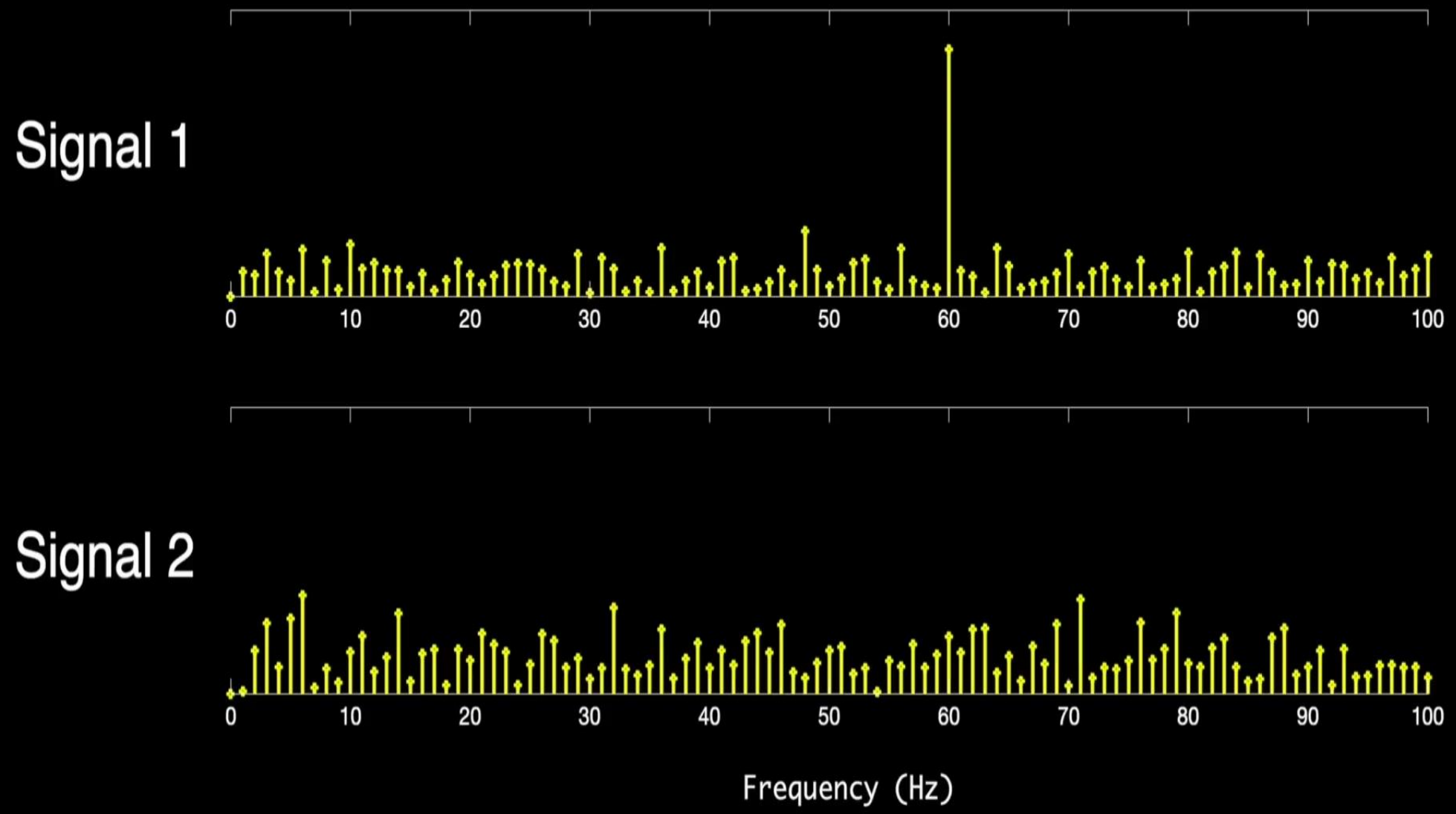
FFTs take advantage of duplicate operations to reduce the number of calculations

Why do we sometimes just look at the absolute value of the FFT?

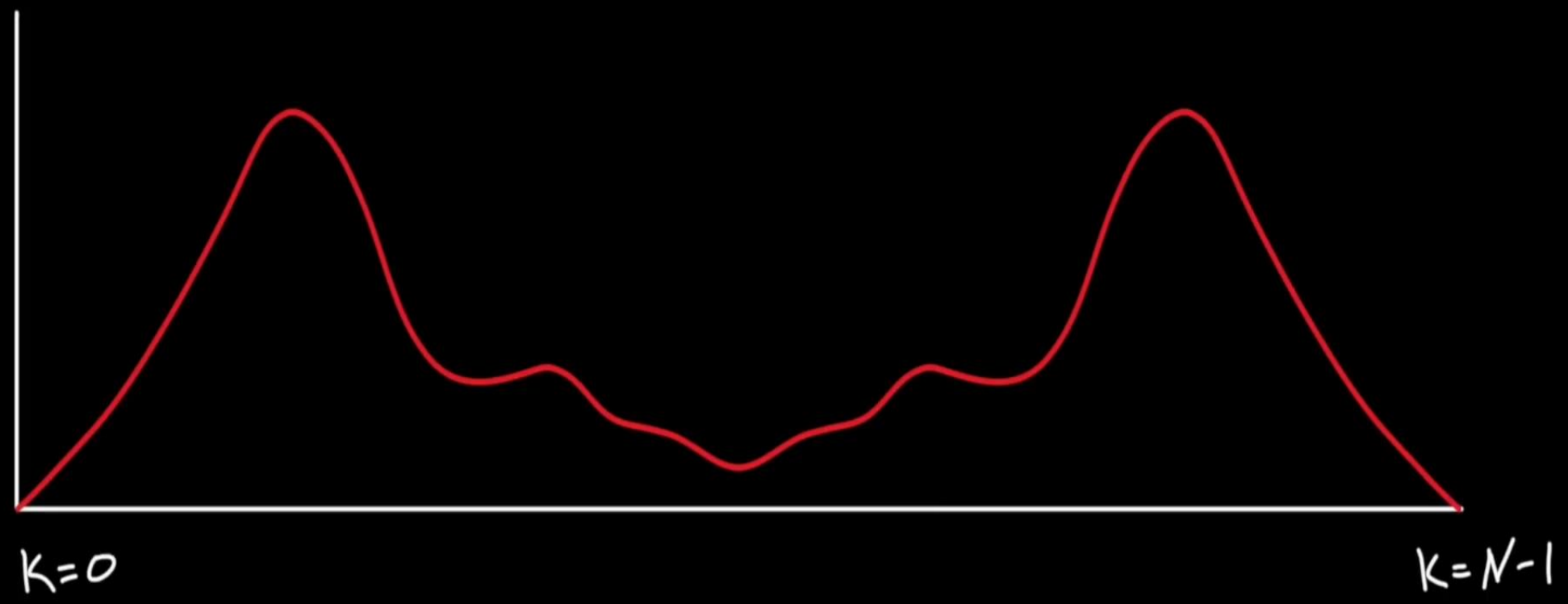
$$X = \mathrm{abs}(\mathrm{fft}(x))$$


MATLAB App
Correlating signal number, k
Number of samples, N
Discrete Time Data
Sample number, n
Time Signal
x[n]
Correlating Signal
N = number of samples
n = sample number
Discrete Fourier Transform
k
Discrete Fourier Transform
Absolute Value

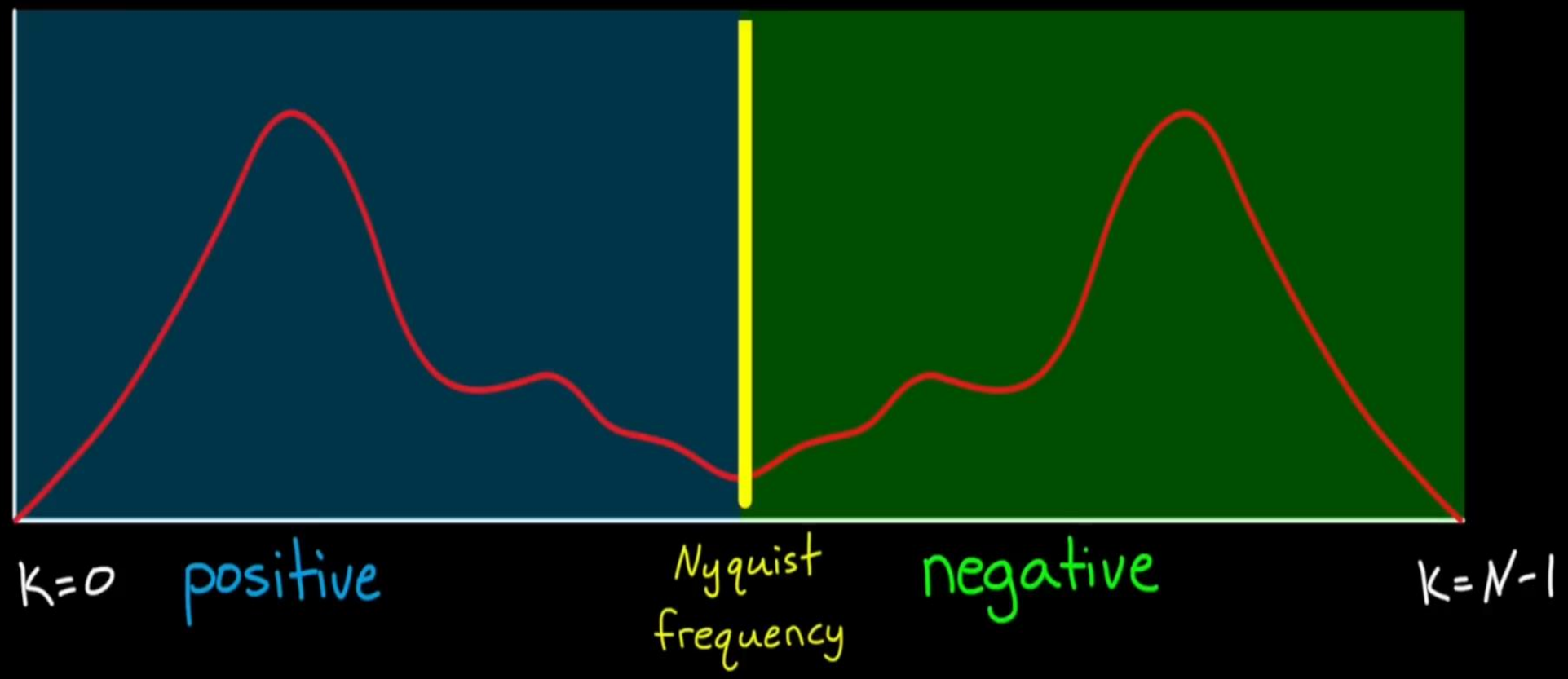
Frequency Domain



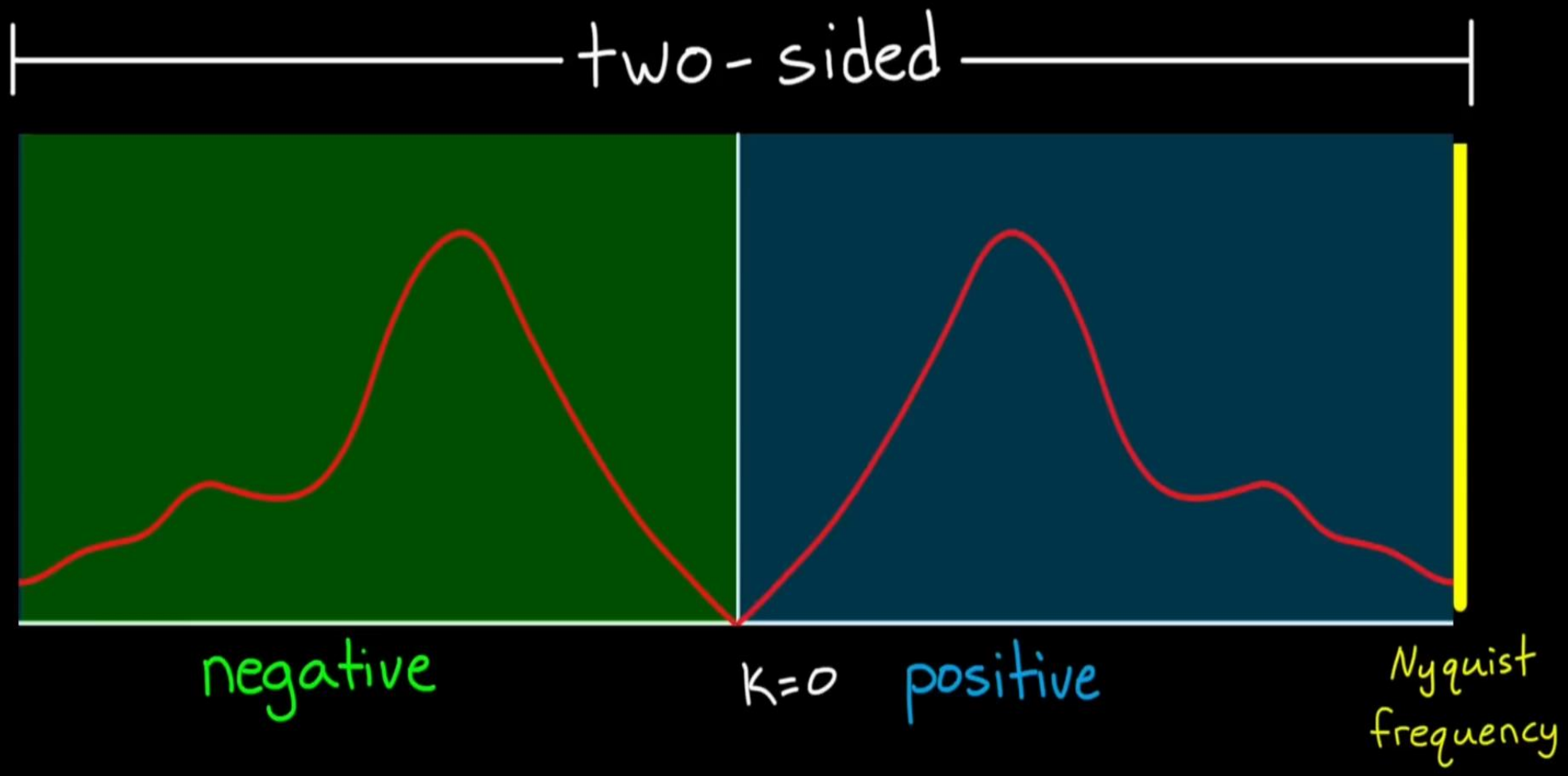
What's the difference between a one-sided and two-sided FFT?



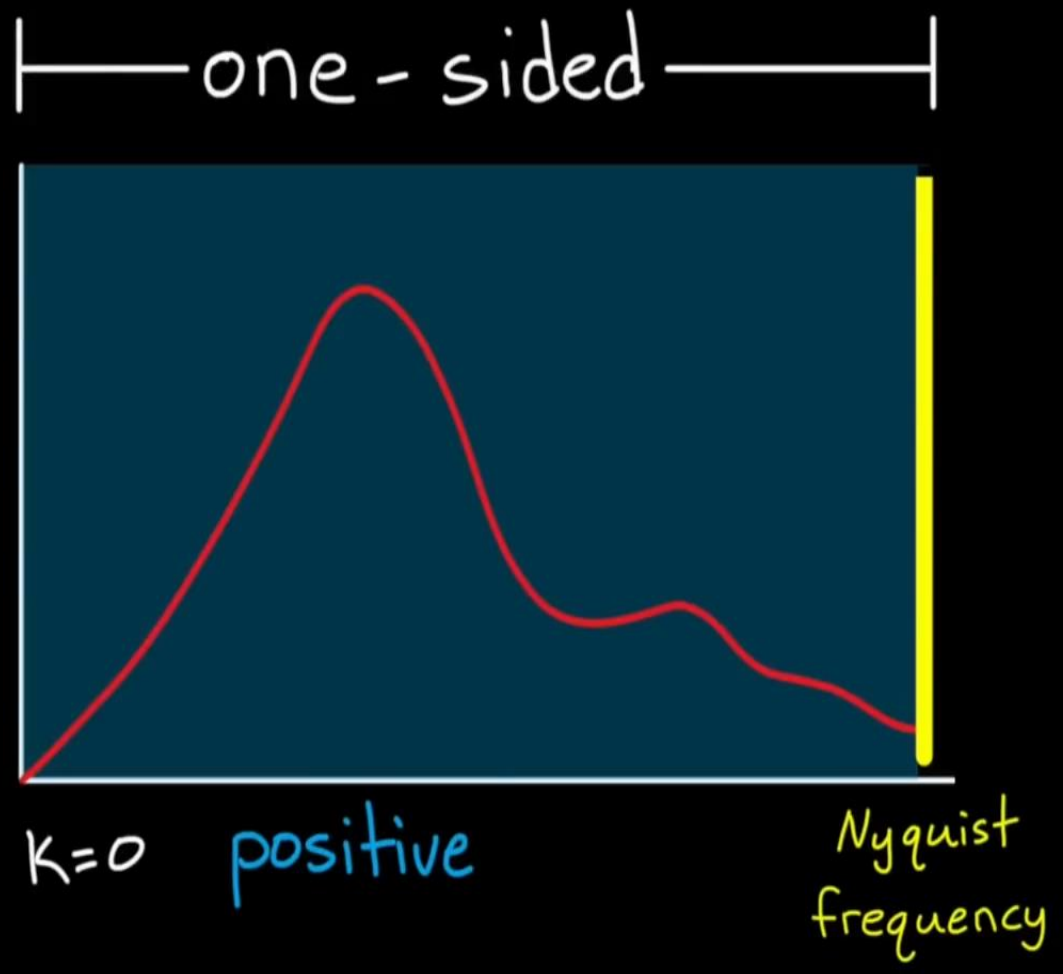
What's the difference between a one-sided and two-sided FFT?

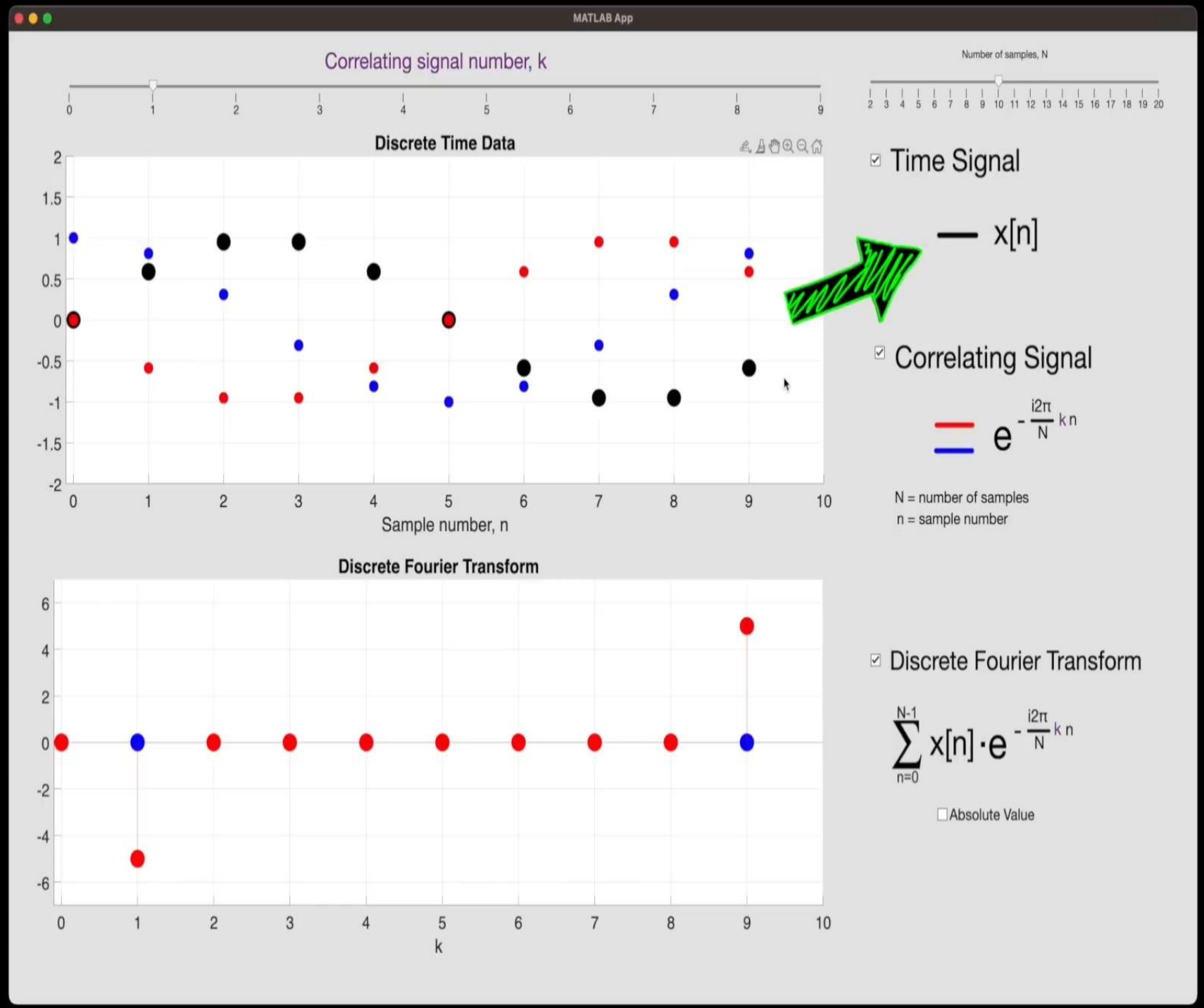


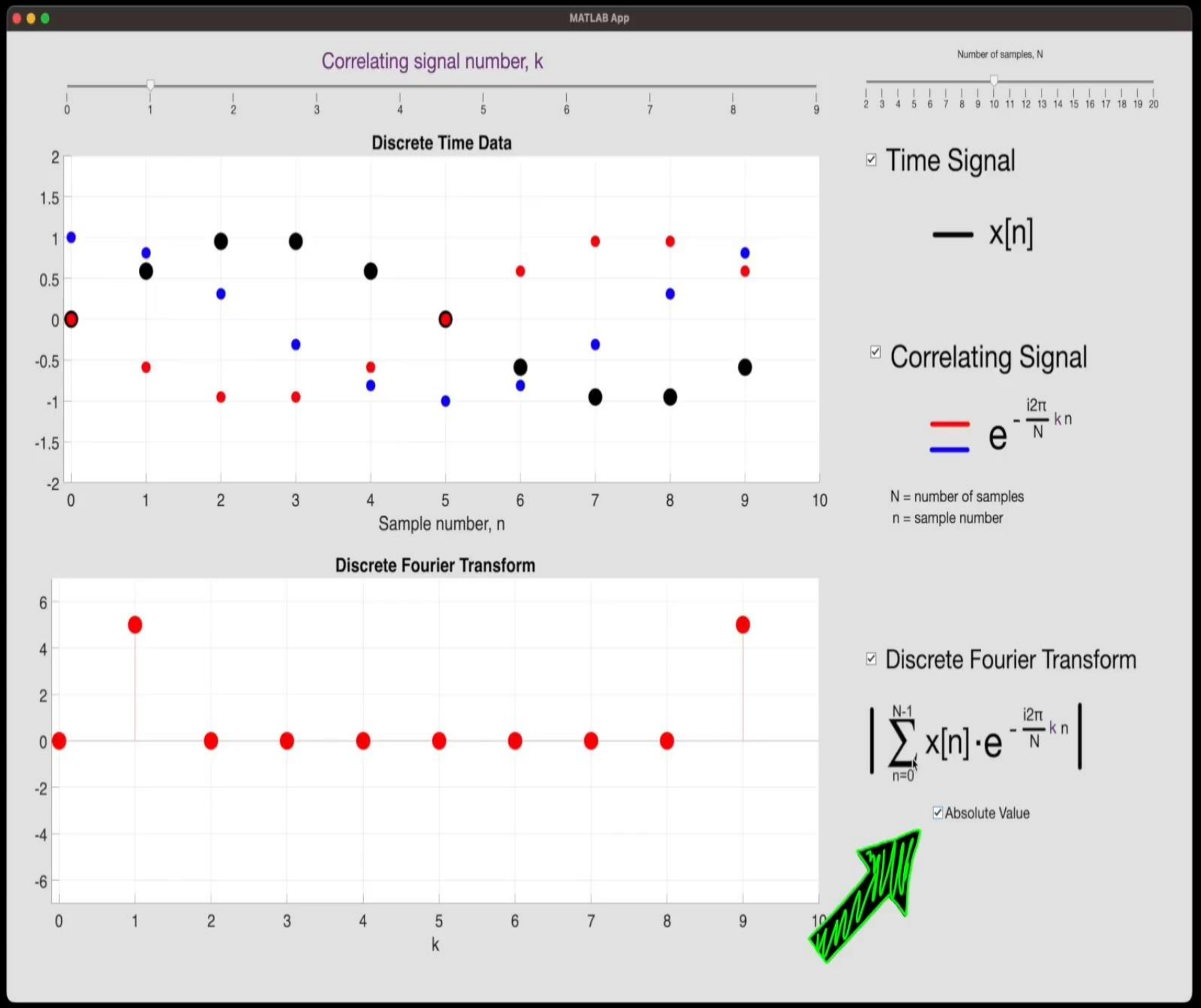
What's the difference between a one-sided and two-sided FFT?

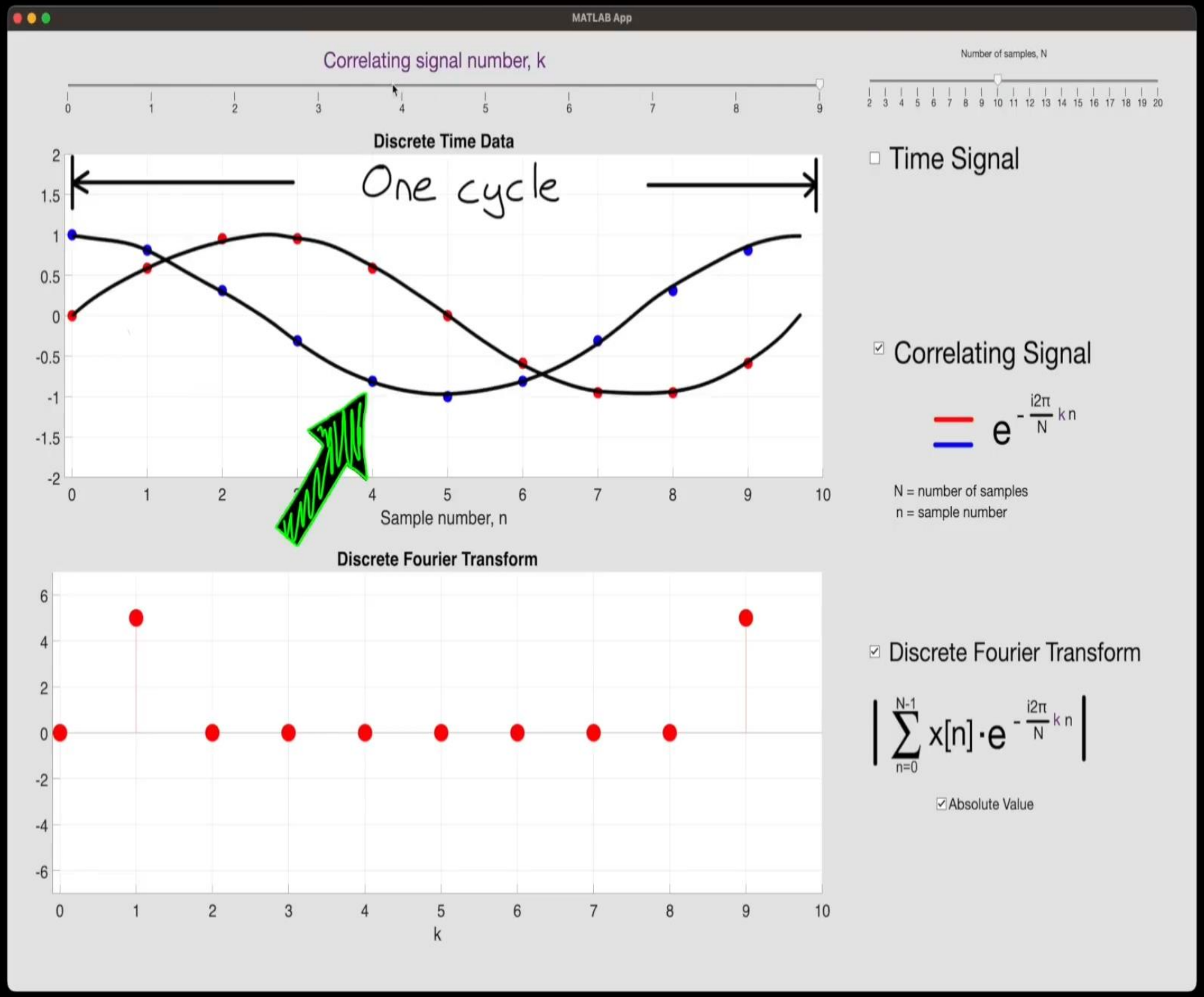


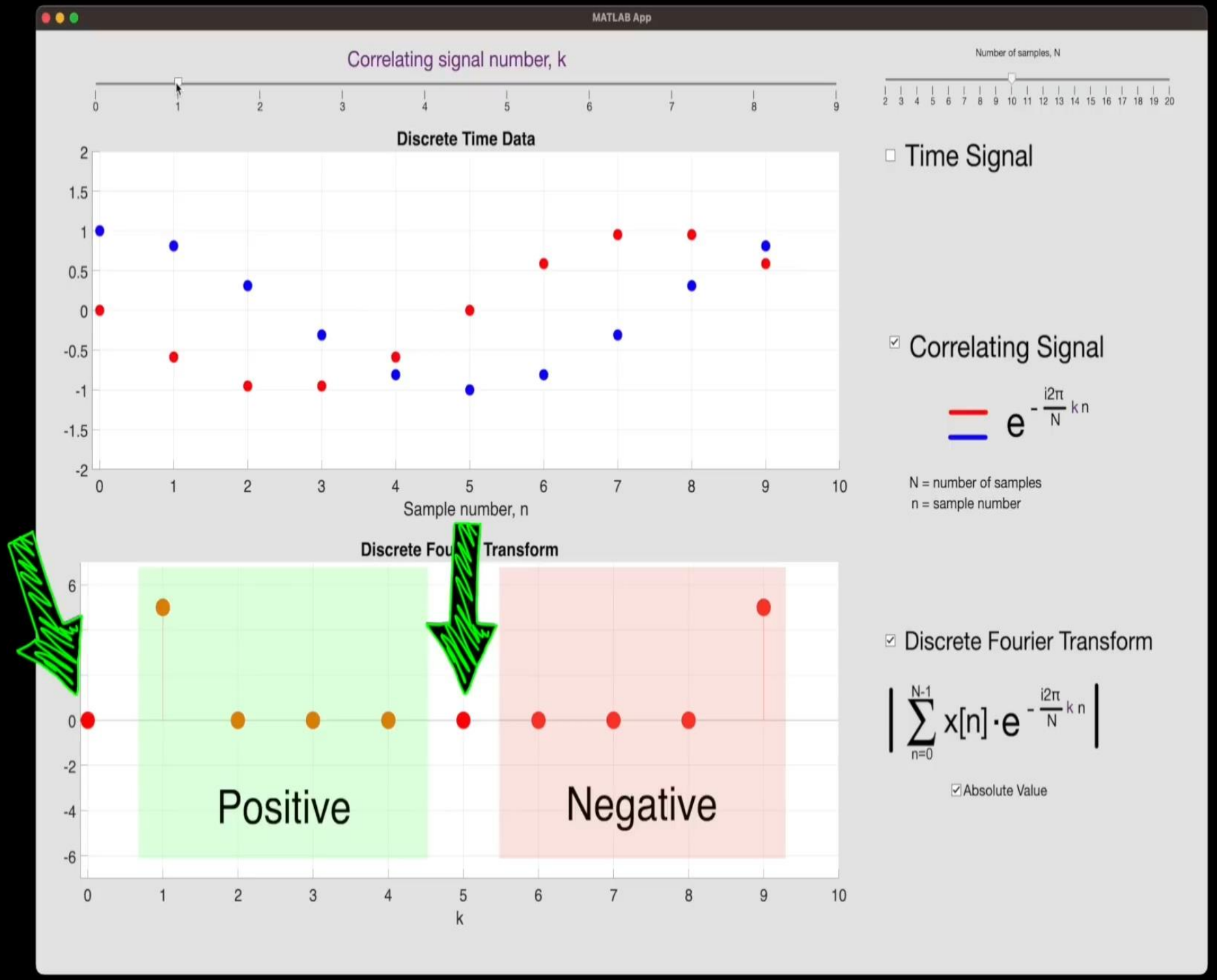
What's the difference between a one-sided and two-sided FFT?

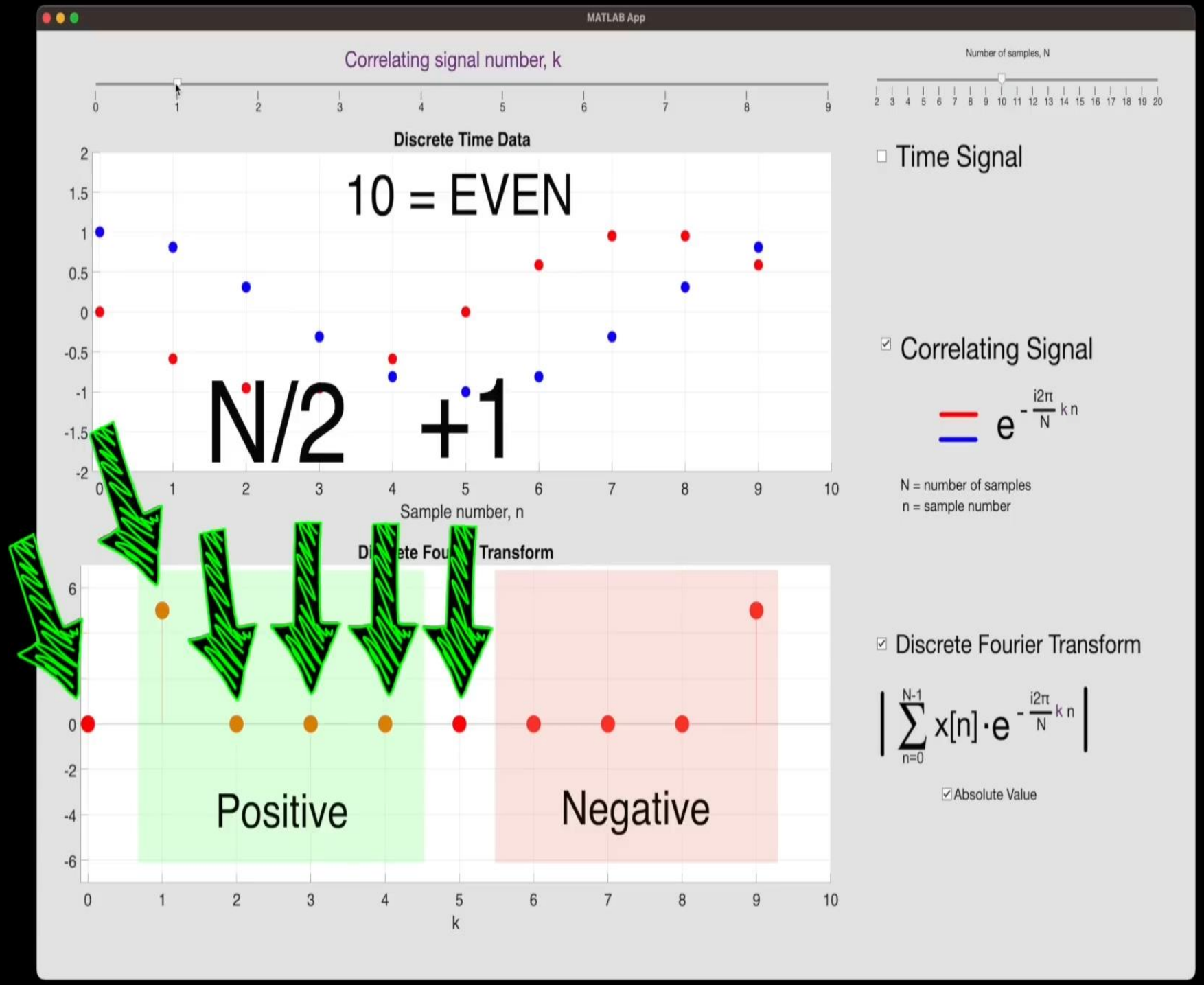


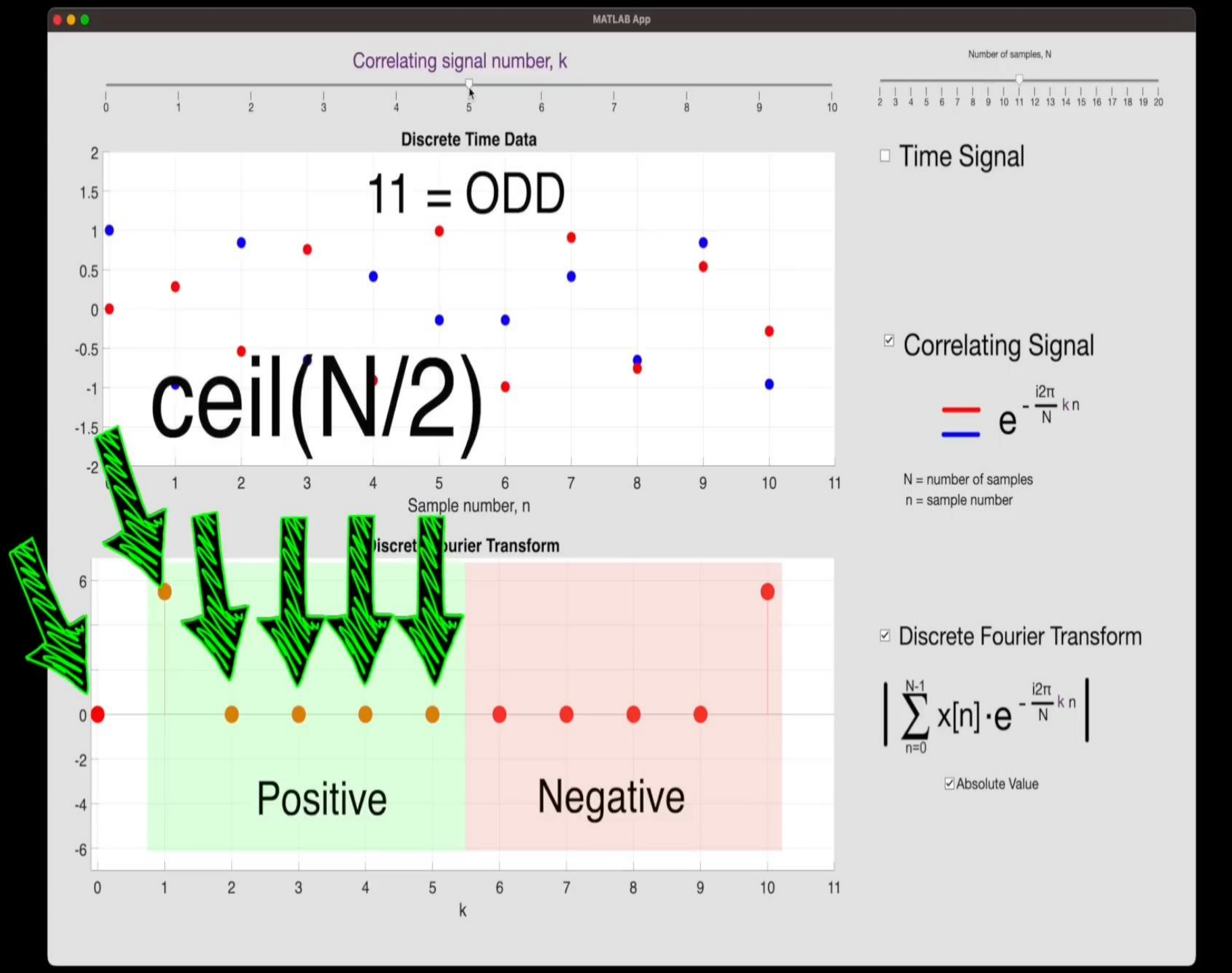

MATLAB App
Correlating signal number, k
Number of samples, N
Discrete Time Data
Sample number, n
Time Signal
x[n]
Correlating Signal
$e^{-\frac{i2\pi}{N}kn}$
N = number of samples
n = sample number
Discrete Fourier Transform
k
Discrete Fourier Transform
$\sum_{n=0}^{N-1} x[n] \cdot e^{-\frac{i2\pi}{N}kn}$
Absolute Value


MATLAB App
Correlating signal number, k
Number of samples, N
Discrete Time Data
Sample number, n
Time Signal
x[n]
Correlating Signal
N = number of samples
n = sample number
Discrete Fourier Transform
k
Discrete Fourier Transform
Absolute Value

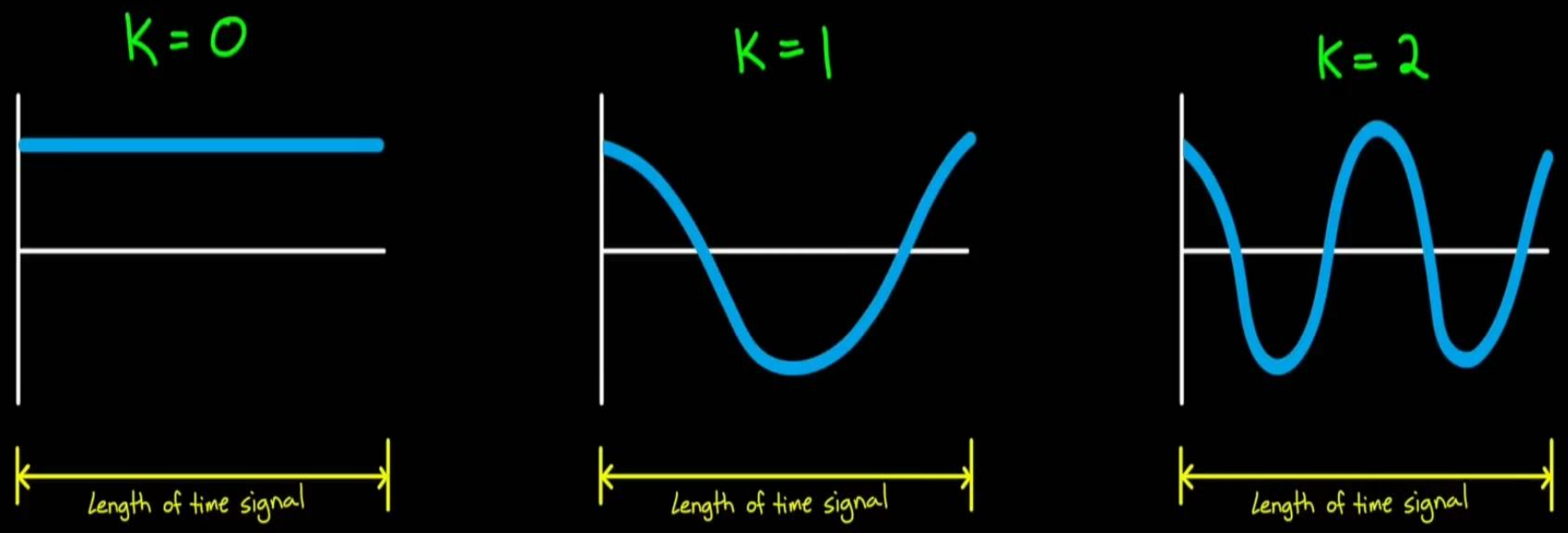

MATLAB App
Correlating signal number, k
Number of samples, N
Discrete Time Data
One cycle
Sample number, n
Time Signal
Correlating Signal
$e^{-\frac{i2\pi}{N}kn}$
N = number of samples
n = sample number
Discrete Fourier Transform
k
Discrete Fourier Transform
$\left| \sum_{n=0}^{N-1} x[n] \cdot e^{-\frac{i2\pi}{N}kn} \right|$
Absolute Value


MATLAB App
Correlating signal number, k
Number of samples, N
Discrete Time Data
Sample number, n
Time Signal
Correlating Signal
$e^{-\frac{i2\pi}{N}kn}$
N = number of samples
n = sample number
Discrete Fourier Transform
k
Positive
Negative
Discrete Fourier Transform
$\left| \sum_{n=0}^{N-1} x[n] \cdot e^{-\frac{i2\pi}{N}kn} \right|$
Absolute Value


MATLAB App
Correlating signal number, k
Number of samples, N
Discrete Time Data
10 = EVEN
N/2 +1
Sample number, n
Time Signal
Correlating Signal
N = number of samples
n = sample number
Discrete Fourier Transform
Positive
Negative
k
Discrete Fourier Transform
Absolute Value

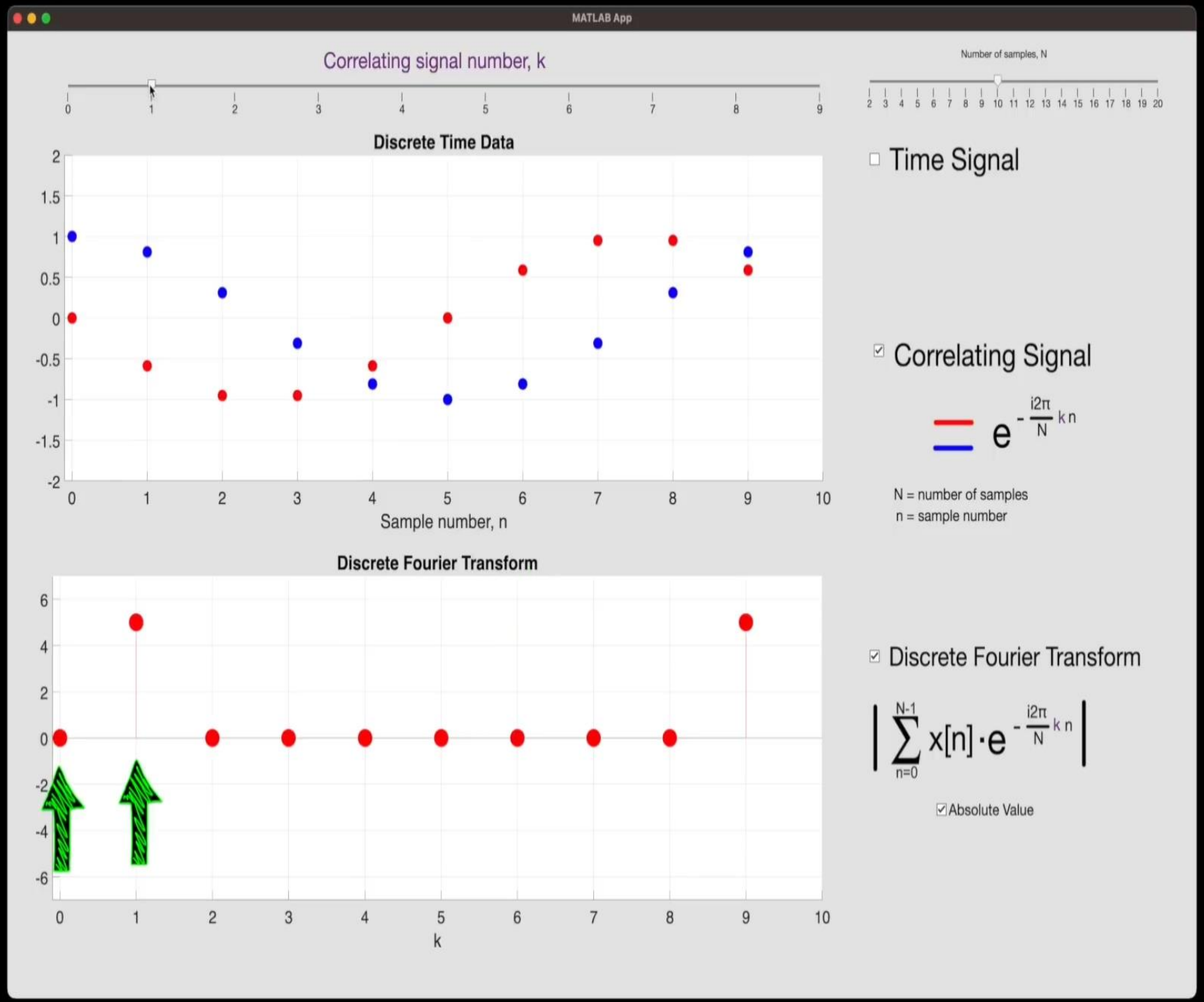

MATLAB App
Correlating signal number, k
Number of samples, N
Discrete Time Data
11 = ODD
ceil(N/2)
Sample number, n
Time Signal
Correlating Signal
N = number of samples
n = sample number
Discrete Fourier Transform
Positive
Negative
k
Absolute Value

What are these frequencies?

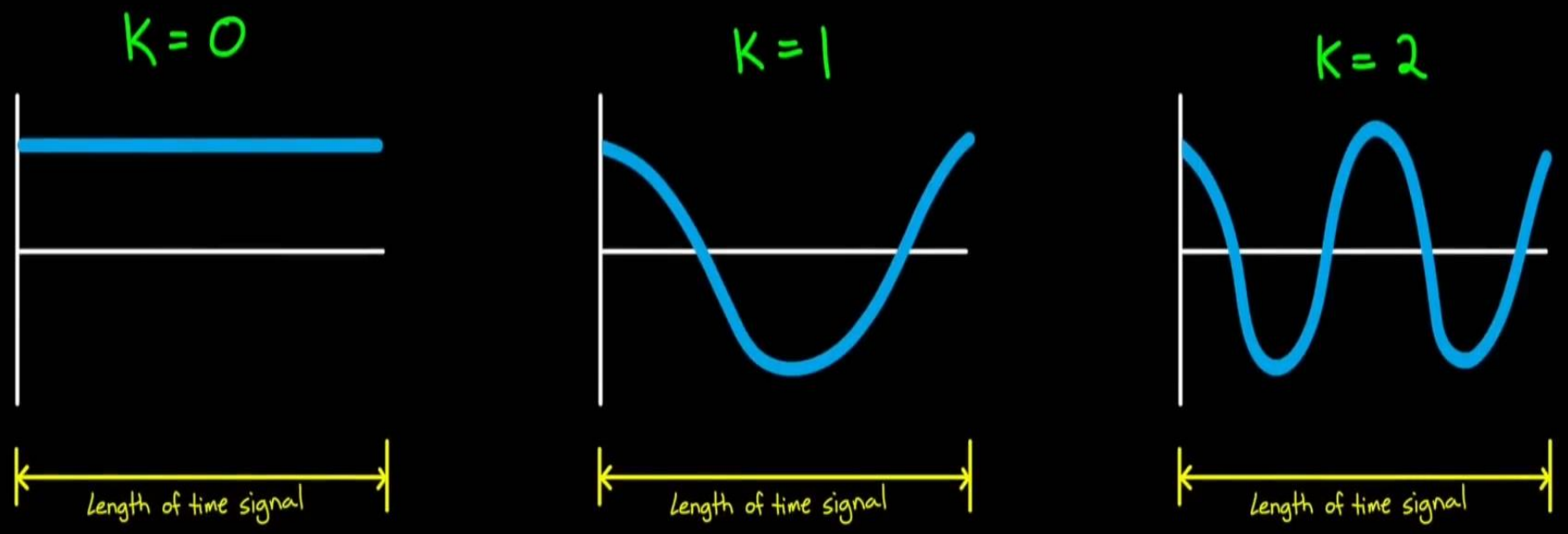


Bin width?

$$\text{frequency} = \frac{K}{\text{Length of time signal}} = \frac{K \cdot f_s}{N}$$

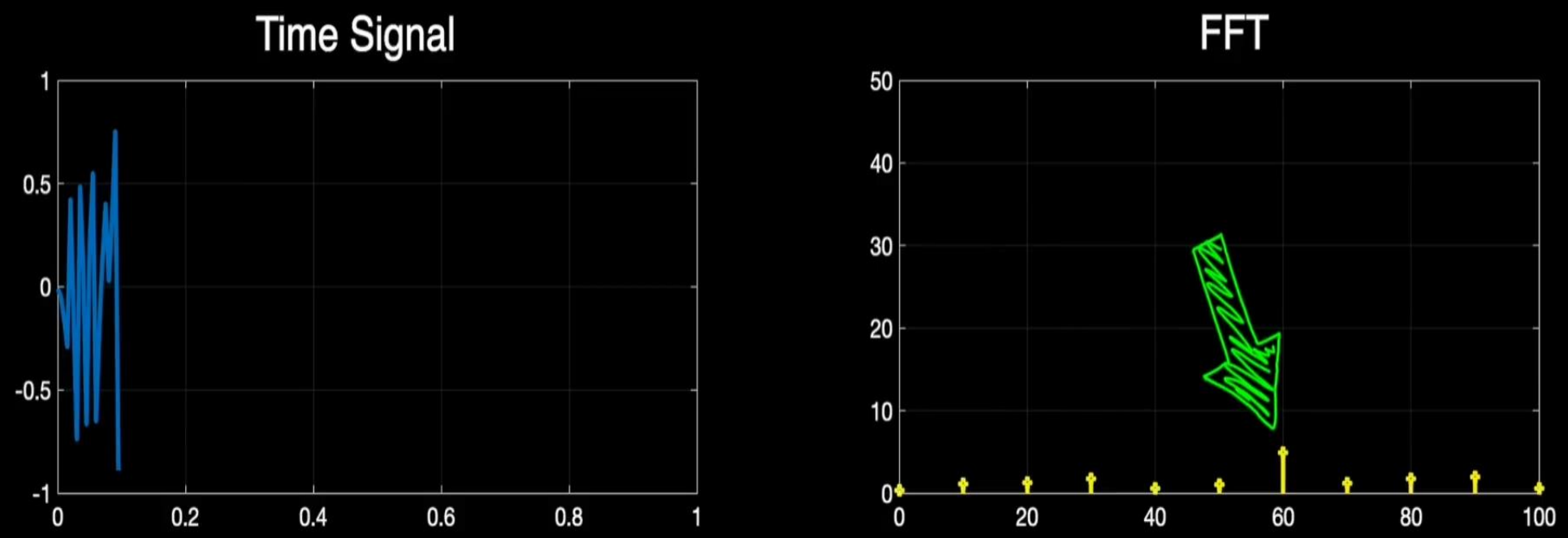

MATLAB App
Correlating signal number, k
Number of samples, N
Discrete Time Data
Sample number, n
Time Signal
Correlating Signal
$e^{-\frac{i2\pi}{N}kn}$
N = number of samples
n = sample number
Discrete Fourier Transform
k
Discrete Fourier Transform
$\left| \sum_{n=0}^{N-1} x[n] \cdot e^{-\frac{i2\pi}{N}kn} \right|$
Absolute Value

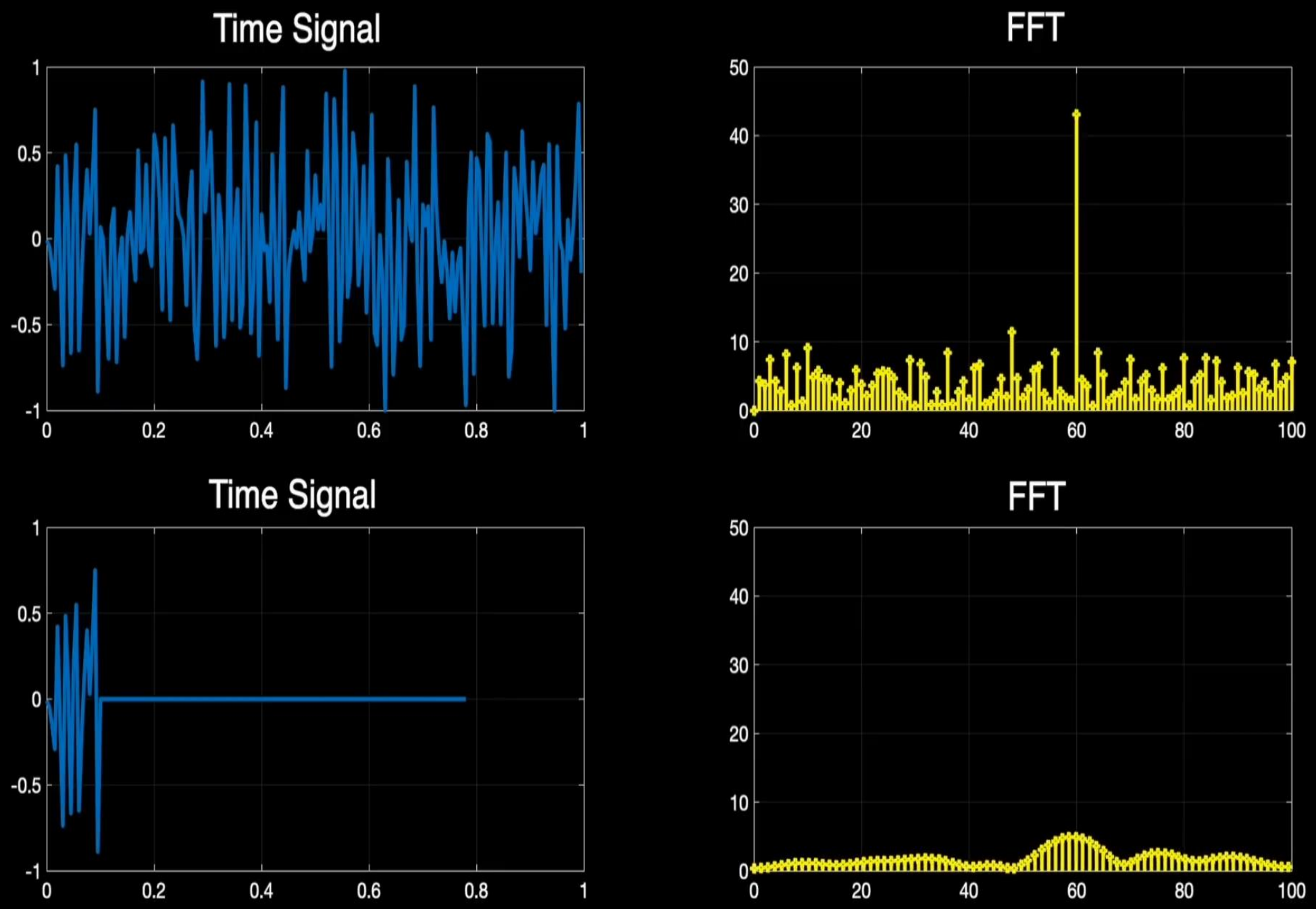
What are these frequencies?



$$\text{Bin width?} = \frac{f_s}{N}$$

$$\text{frequency} = \frac{K}{\text{Length of time signal}} = \frac{K \cdot f_s}{N}$$


Time Signal
1
0.5
0
-0.5
-1
0
0.2
0.4
0.6
0.8
1
FFT
50
40
30
20
10
0
0
20
40
60
80
100

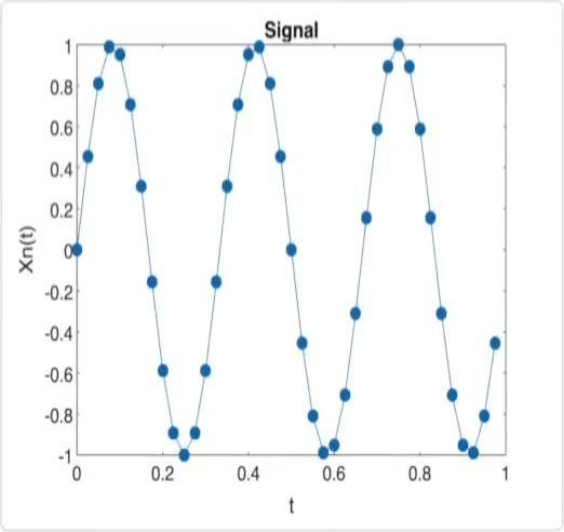

Time Signal
1
0.5
0
-0.5
-1
0
0.2
0.4
0.6
0.8
1
FFT
50
40
30
20
10
0
0
20
40
60
80
100
Time Signal
1
0.5
0
-0.5
-1
0
0.2
0.4
0.6
0.8
1
FFT
50
40
30
20
10
0
0
20
40
60
80
100

/Users/brian/Documents/MATLAB/FFT/FFT_video.mlx

FFT Example

Create time domain signal

```
fs = 40;        % Sampling frequency
T = 1/fs;       % Sampling period
N = 40;         % Length of signal
t = (0:N-1)*T';  % Time vector

Xn = sin(2*pi*3*t);       % 3 Hz Signal

plot(t, Xn, '.-', 'MarkerSize', 20);
title("Signal")
xlabel("t")
ylabel("Xn(t)")
```

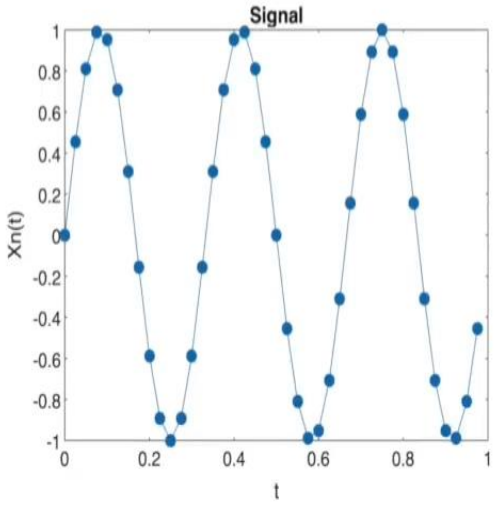


/Users/brian/Documents/MATLAB/FFT/FFT_video.mlx *

FFT Example

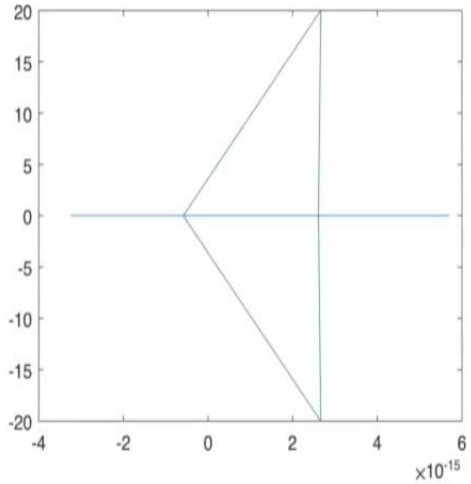
Create time domain signal

```
fs = 40;        % Sampling frequency
T = 1/fs;       % Sampling period
N = 40;         % Length of signal
t = (0:N-1)*T';  % Time vector

Xn = sin(2*pi*3*t);       % 3 Hz Signal

plot(t, Xn, '.-', 'MarkerSize', 20);
title("Signal")
xlabel("t")
ylabel("Xn(t)")
```



One-sided FFT

```
Y = fft(Xn);
figure
plot(Y)
```



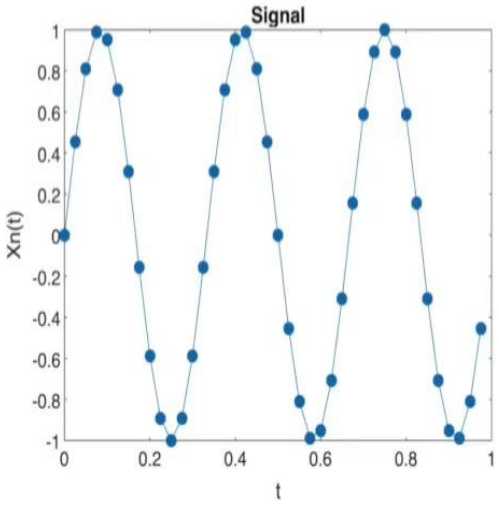
Zoom: 175% UTF-8 LF script Ln 14 Col 8

/Users/brian/Documents/MATLAB/FFT/FFT_video.mlx *

FFT Example

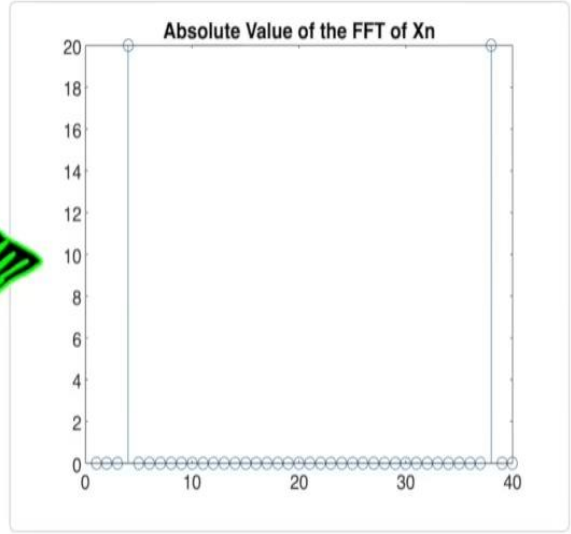
Create time domain signal

```
fs = 40;        % Sampling frequency
T = 1/fs;       % Sampling period
N = 40;         % Length of signal
t = (0:N-1)*T';  % Time vector

Xn = sin(2*pi*3*t);       % 3 Hz Signal

plot(t, Xn, '.-', 'MarkerSize', 20);
title("Signal")
xlabel("t")
ylabel("Xn(t)")
```



One-sided FFT

```
Y = fft(Xn);
figure
stem(abs(Y));
title("Absolute Value of the FFT of Xn");
```

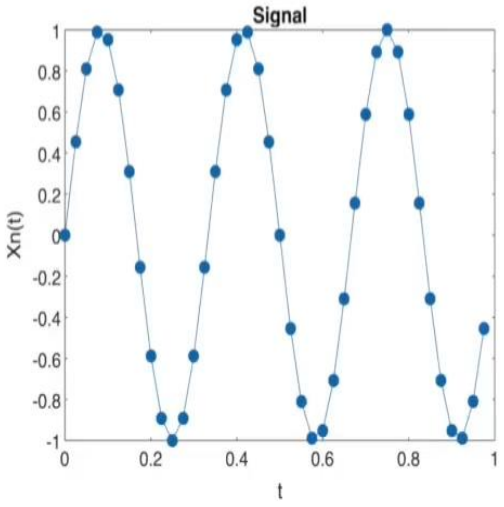


/Users/brian/Documents/MATLAB/FFT/FFT_video.mlx *

FFT Example

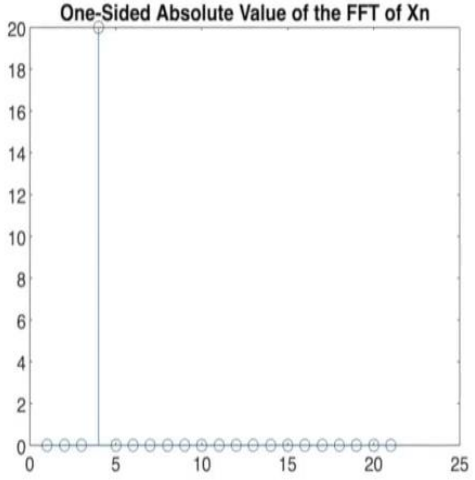
Create time domain signal

```
fs = 40;        % Sampling frequency
T = 1/fs;       % Sampling period
N = 40;         % Length of signal
t = (0:N-1)*T';  % Time vector

Xn = sin(2*pi*3*t);       % 3 Hz Signal

plot(t, Xn, '.-', 'MarkerSize', 20);
title("Signal")
xlabel("t")
ylabel("Xn(t)")
```



One-sided FFT

```
Y = fft(Xn);
figure
stem(abs(Y(1:(N/2)+1)));
title("One-Sided Absolute Value of the FFT of Xn");
```



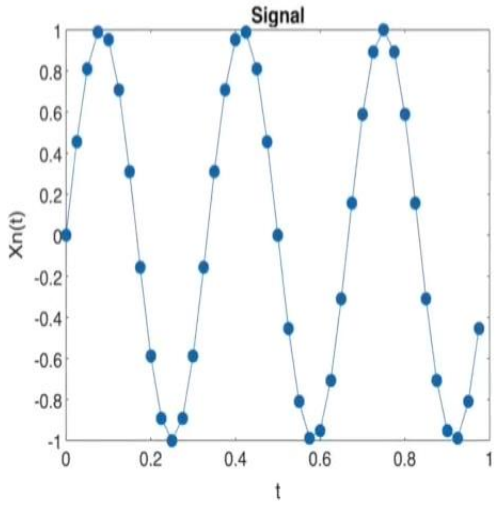
Zoom: 175% UTF-8 LF script Ln 15 Col 52

/Users/brian/Documents/MATLAB/FFT/FFT_video.mlx *

FFT Example

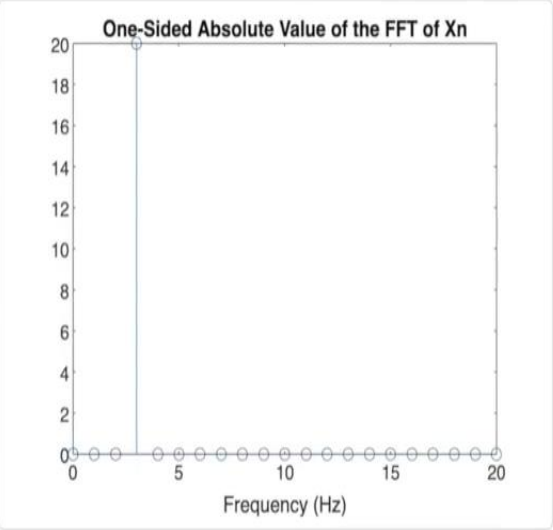
Create time domain signal

```
fs = 40;        % Sampling frequency
T = 1/fs;       % Sampling period
N = 40;         % Length of signal
t = (0:N-1)*T';  % Time vector

Xn = sin(2*pi*3*t);       % 3 Hz Signal

plot(t, Xn, '.-', 'MarkerSize', 20);
title("Signal")
xlabel("t")
ylabel("Xn(t)")
```



One-sided FFT

```
Y = fft(Xn);

k = 0:N/2;      % k starts at 0 so the length of k here is N/2+1
freq = k*fs/N;  % Cycles per length of the signal in seconds

figure
stem(freq, abs(Y(1:(N/2)+1)));
title("One-Sided Absolute Value of the FFT of Xn");
xlabel("Frequency (Hz)");
```



Zoom: 175% UTF-8 LF script Ln 20 Col 26