Class Activity

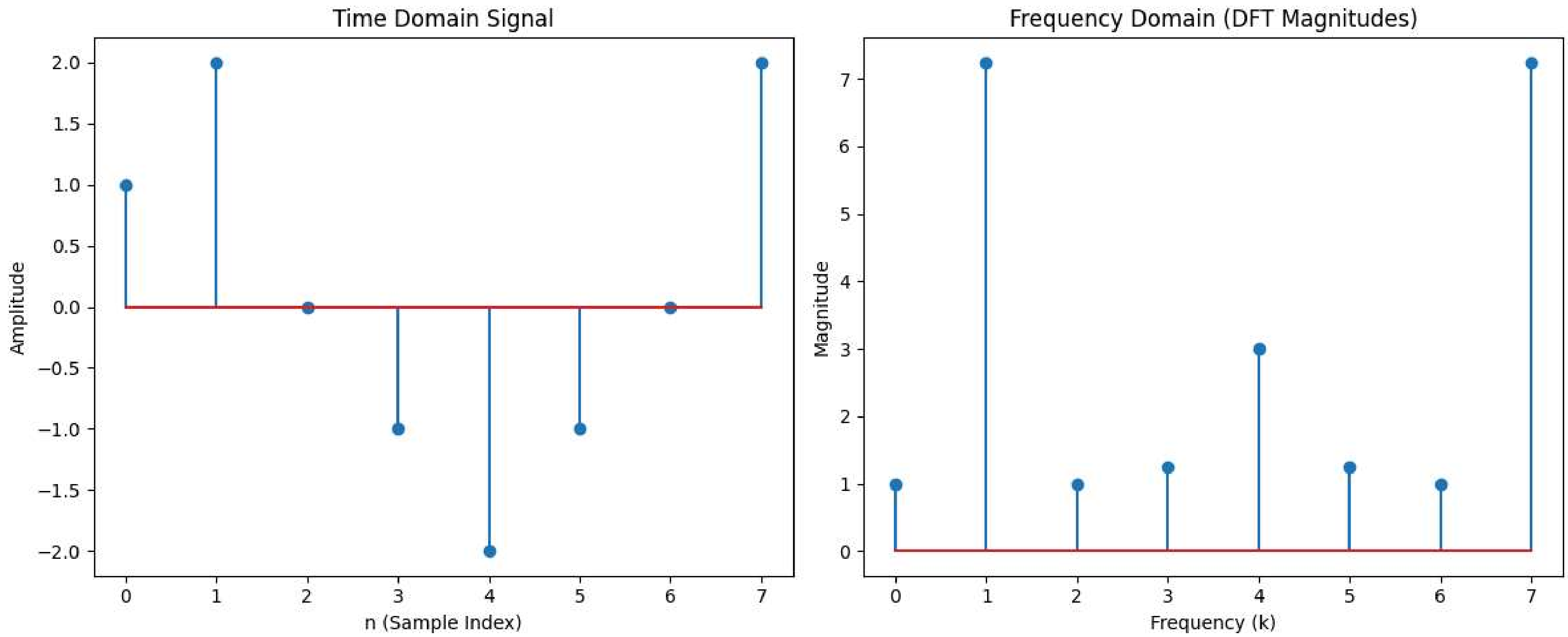
Arsyadana Estu Aziz (121140068)

Discrete Fourier Transform.

In [19]:
```
import numpy as np
import matplotlib.pyplot as plt

# Sample Signal
x_n = [1, 2, 0, -1, -2, -1, 0, 2]

# Dft Function from Slides Template
def dft(signal):
    N = len(signal)
    # Create an empty list to store the DFT result
    X = np.zeros(N, dtype=complex)

    # DFT calculation using the formula
    for k in range(N): # For each frequency component
        sum_value = 0
        for n in range(N): # For each time component
            angle = -2j * np.pi * k * n / N
            sum_value += signal[n] * np.exp(angle)
        X[k] = sum_value

    return X

# Calculate DFT of the signal
dft_output = dft(x_n)
print(dft_output)

# Compute magnitudes for the frequency domain
dft_magnitudes = np.abs(dft_output)

# Plot the time domain signal
plt.figure(figsize=(12, 5))
```

```python
plt.subplot(1, 2, 1)
plt.stem(range(len(x_n)), x_n)
plt.title('Time Domain Signal')
plt.xlabel('n (Sample Index)')
plt.ylabel('Amplitude')

# Plot the frequency domain (DFT magnitudes)
plt.subplot(1, 2, 2)
plt.stem(range(len(dft_magnitudes)), dft_magnitudes)
plt.title('Frequency Domain (DFT Magnitudes)')
plt.xlabel('Frequency (k)')
plt.ylabel('Magnitude')

plt.tight_layout()
plt.show()
```

```
[ 1.        +0.00000000e+00j  7.24264069+4.44089210e-16j
 -1.        -4.44089210e-16j -1.24264069+0.00000000e+00j
 -3.        -1.95943488e-15j -1.24264069-2.44249065e-15j
 -1.        -1.33226763e-15j  7.24264069+9.10382880e-15j]
```


Time Domain Signal
Amplitude
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
0
1
2
3
4
5
6
7
n (Sample Index)
Frequency Domain (DFT Magnitudes)
Magnitude
7
6
5
4
3
2
1
0
0
1
2
3
4
5
6
7
Frequency (k)