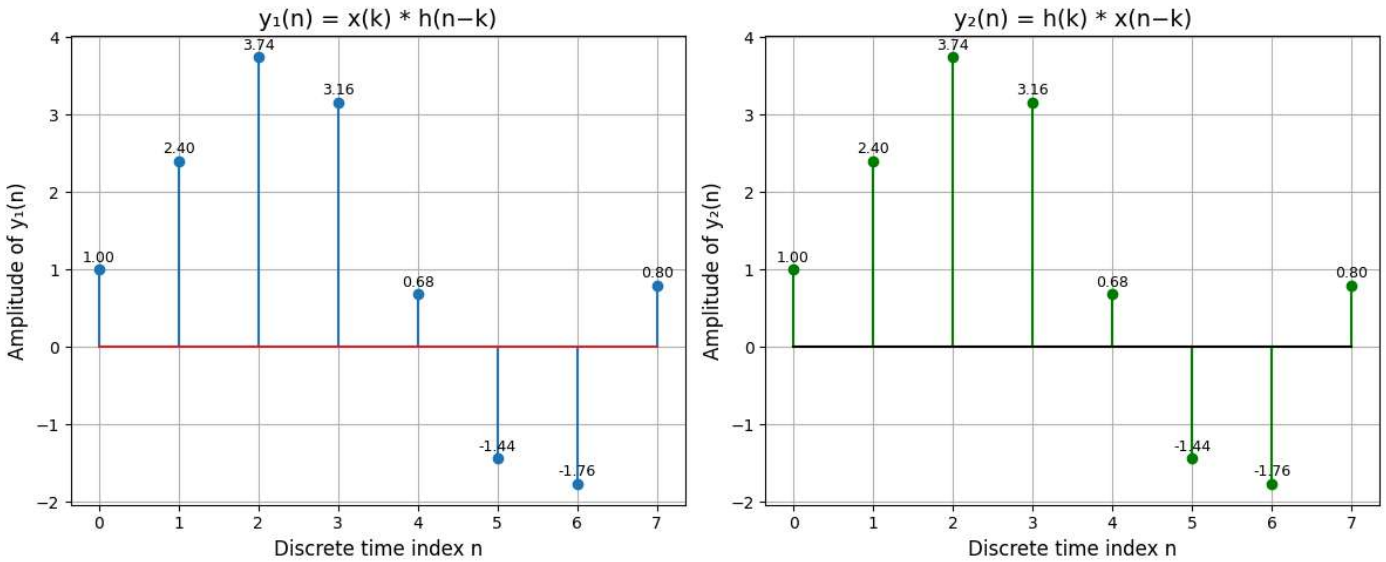
```
import numpy as np
import matplotlib.pyplot as plt

# Given sequences
x = np.array([1, 1.2, 1.5, 1.2, -1])
h = np.array([1, 1.2, 0.8, -0.8])

def graphical_convolution(x, h, mode='x*h'):
    L = len(x) + len(h) - 1
    y = np.zeros(L)

    for n in range(L):
        if mode == 'x*h':
            h_shifted = np.zeros(len(x))
            for k in range(len(x)):
                if 0 <= n - k < len(h):
                    h_shifted[k] = h[n - k]
            y[n] = np.sum(x * h_shifted)

        elif mode == 'h*x':
            x_shifted = np.zeros(len(h))
            for k in range(len(h)):
                if 0 <= n - k < len(x):
                    x_shifted[k] = x[n - k]
            y[n] = np.sum(h * x_shifted)

    return y

# Compute both convolutions
y1 = graphical_convolution(x, h, mode='x*h')
y2 = graphical_convolution(x, h, mode='h*x')
n = np.arange(len(y1))

# Plotting
plt.figure(figsize=(12, 5))

# Plot y1(n)
plt.subplot(1, 2, 1)
plt.stem(n, y1)
plt.title('y₁(n) = x(k) * h(n-k)', fontsize=14)
plt.xlabel('Discrete time index n', fontsize=12)
plt.ylabel('Amplitude of y₁(n)', fontsize=12)
plt.grid(True)
for i in range(len(n)):
    plt.text(n[i], y1[i] + 0.1, f'{y1[i]:.2f}', ha='center', fontsize=9)

# Plot y2(n)
plt.subplot(1, 2, 2)
plt.stem(n, y2, linefmt='g-', markerfmt='go', basefmt='k-')
plt.title('y₂(n) = h(k) * x(n-k)', fontsize=14)
plt.xlabel('Discrete time index n', fontsize=12)
plt.ylabel('Amplitude of y₂(n)', fontsize=12)
plt.grid(True)
for i in range(len(n)):
    plt.text(n[i], y2[i] + 0.1, f'{y2[i]:.2f}', ha='center', fontsize=9)

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


y₁(n) = x(k) * h(n−k)
Amplitude of y₁(n)
Discrete time index n
1.00
2.40
3.74
3.16
0.68
-1.44
-1.76
0.80
y₂(n) = h(k) * x(n−k)
Amplitude of y₂(n)
Discrete time index n
1.00
2.40
3.74
3.16
0.68
-1.44
-1.76
0.80