Marwadi University	**Marwari University** **Faculty of Technology** **Department of Information and Communication Technology**	
Subject: Digital Signal and Image Processing(01CT0513)	**Aim:** Simulate Cross Correlation and Auto correlation on Discrete Time Signals.	
Experiment No: 03	**Date: 20-08-2025**	**Enrollment No: 92301733054**

Theory:-

Cross-correlation and autocorrelation are mathematical operations used to measure the similarity or correlation between two signals. They are widely used in various applications, such as signal processing, image processing, and pattern recognition.

Cross-correlation measures the similarity between two signals at different time shifts. It computes the dot product of one signal with a time-shifted version of the other signal. The resulting cross-correlation signal indicates the similarity between the two signals at different time lags.

Autocorrelation, on the other hand, measures the similarity of a signal with a time-shifted version of itself. It computes the cross-correlation of a signal with itself. The autocorrelation signal shows how the signal is correlated with itself at different time lags.

Code:-

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

def cross_correlation(signal1, signal2):
    # Compute the cross-correlation
    cross_corr = np.correlate(signal1, signal2, mode='full')
    return cross_corr

def autocorrelation(signal):
    # Compute the autocorrelation
    auto_corr = np.correlate(signal, signal, mode='full')
    return auto_corr

# Define the discrete-time signals
signal1 = np.array([1, 2, 3, 4, 5])
signal2 = np.array([2, 4, 6, 8, 10])

# Compute the cross-correlation
cross_corr = cross_correlation(signal1, signal2)

# Compute the autocorrelation
auto_corr = autocorrelation(signal1)

# Create the time lags for plotting
lags_cross = np.arange(-len(signal1) + 1, len(signal2))
lags_auto = np.arange(-len(signal1) + 1, len(signal1))
```

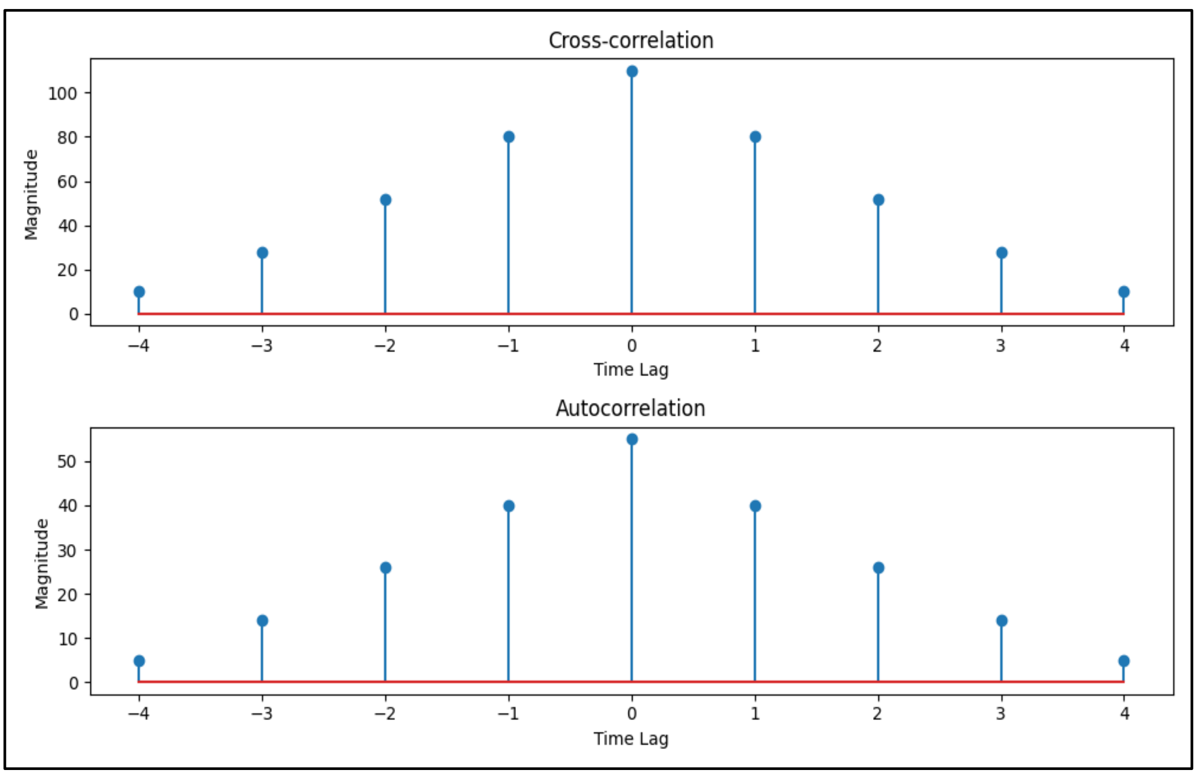
```
# Plot the cross-correlation and autocorrelation signals
plt.figure(figsize=(10, 6))

plt.subplot(2, 1, 1)
plt.stem(lags_cross, cross_corr)
plt.title('Cross-correlation')
plt.xlabel('Time Lag')
plt.ylabel('Magnitude')

plt.subplot(2, 1, 2)
plt.stem(lags_auto, auto_corr)
plt.title('Autocorrelation')
plt.xlabel('Time Lag')
plt.ylabel('Magnitude')

plt.tight_layout()
plt.savefig("./Corelation.png")
plt.show()
```

Output :-



Observation:-

Cross-Correlation:

1. The cross-correlation between signal1 and signal2 shows a symmetric triangular pattern.
2. The maximum peak occurs at lag = 0, which means the two signals are best aligned without any shift.
3. Since signal2 is exactly 2 × signal1, the cross-correlation values are proportionally scaled and higher in magnitude.
4. The plot is symmetric about zero lag, confirming the similarity between the signals.

Autocorrelation:

1. The autocorrelation of signal1 shows a symmetric triangular pattern as well.
2. The highest value is at lag = 0, representing the signal energy (sum of squares of signal1).
3. As the lag increases in either direction (positive or negative), the overlap between the signal and its shifted version decreases, hence the correlation values reduce.
4. The plot is symmetric around zero lag, which is a property of autocorrelation functions.

Conclusion:-

We learn that cross-correlation finds similarity between two signals, with maximum alignment at zero lag. Autocorrelation shows a signal is most similar to itself at lag = 0, where the peak represents its energy