**Exp no:3c) CROSS CORRELATION**

**Date :**

**AIM:**

To compute and analyse cross correlation using python.

**ALGORITHM:**

1. Start
2. Input two signals : signal 1 and signal 2.
3. Reverse signal 2
4. Slide signal 2 over signal 1 and at each step multiply overlapping values , sum the result and store the result as the cross correlation values for each lag.
5. Repeat 4 for all possible lags.
6. Display the cross correlation result.
7. End

**MATHEMATICL EXPRESSION:**

R

**FLOW CHART:**

Start

End

Display result

Compute cross correlation

Input two signals

**PROGRAM:**

import numpy as np

import matplotlib.pyplot as plt

# Step 2: Create two signals (arrays)

signal1 = np.array([1, 2, 3, 4, 5])

signal2 = np.array([5, 4, 3, 2, 1])

# Step 3: Compute cross-correlation using numpy

cross\_corr = np.correlate(signal1, signal2, mode='full')

# Step 4: Display the signals and the cross-correlation result

lags = np.arange(-len(signal2) + 1, len(signal1))

plt.figure(figsize=(12, 4))

plt.subplot(1, 3, 1)

plt.stem(signal1)

plt.title("Signal 1")

plt.subplot(1, 3, 2)

plt.stem(signal2)

plt.title("Signal 2")

plt.subplot(1, 3, 3)

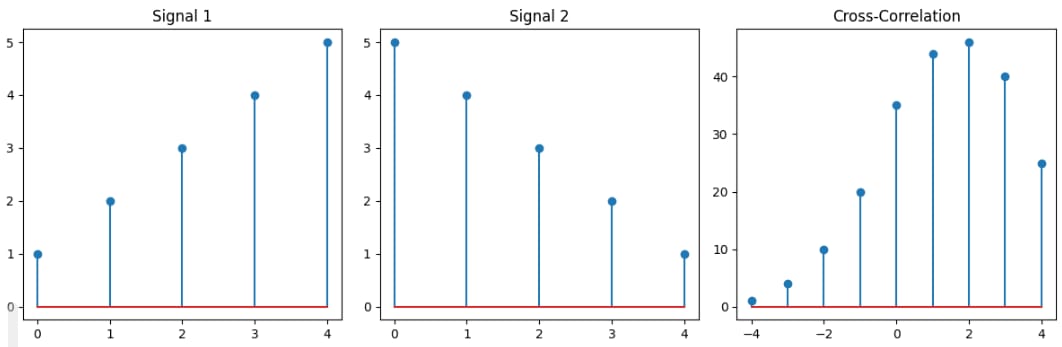
plt.stem(lags, cross\_corr)

plt.title("Cross-Correlation")

plt.tight\_layout()

plt.show()

**OUTPUT:**



**<form method="POST" action="{{ url\_for('apply\_job', job\_id=job.id) }}">**

**<label for="job\_preference">Job Preference</label>**

**<select name="job\_preference" id="job\_preference" required>**

**<option value="" disabled selected>Select your preference</option>**

**<option value="Full-time">Full-time</option>**

**<option value="Part-time">Part-time</option>**

**<option value="Contract">Contract</option>**

**<option value="Internship">Internship</option>**

**</select>**

**<button type="submit">Submit Application</button>**

**</form>**

**RESULT:**

The implement and analysis of cross correlation is and the output is plotted.