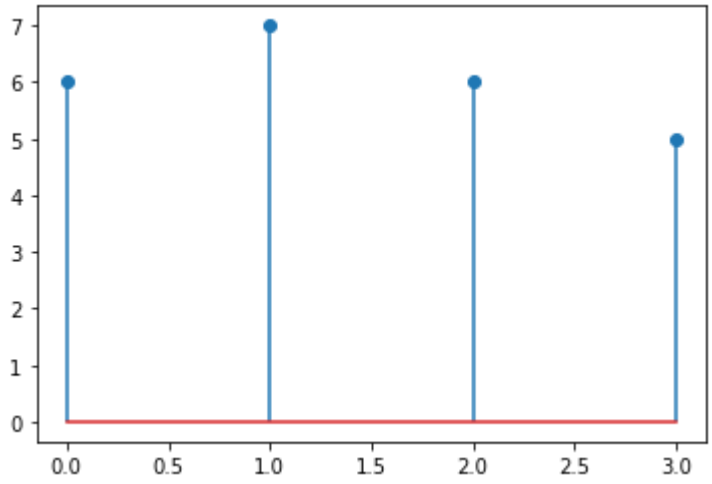
In [11]:

```
import numpy as np
import matplotlib.pyplot as plt
from scipy import signal


def circularconvo(x,y):
    size = len(x)
    ans = []

    for i in range(0,size):
        sum = 0
        for j in range(0,size):
            sum += x[j]*y[(i-j)%size]
        ans.append(sum)

    return ans

# 1  circular convolution
x = [1,2,0,1]
y = [2,2,1,1]

lenX = len(x)
lenY = len(y)
newy = circularconvo(x,y)
plt.stem(newy)
print(newy)
```

```
[6, 7, 6, 5]
```

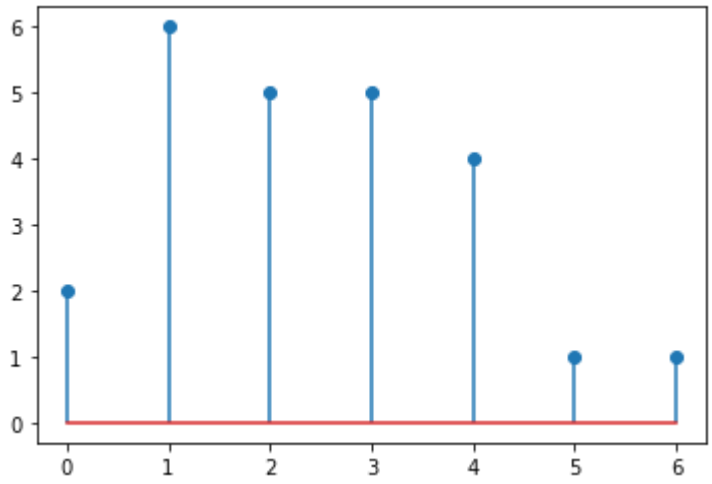


In [16]:

```
#adding len(x) - 1 zeros in y
y1 = y
x1 = x

# 2.2 Linear convolution of x and y using circular convolution

for i in range(0,lenX - 1):
    y1.append(0)

#adding len(y) - 1 zeros in y

for i in range(0,lenY - 1):
    x1.append(0)
```

```
print(x1)
print( y1)
z  = circularconvo(x1,y1)

plt.stem(z)
print(z)
```

```
[1, 2, 0, 1, 0, 0, 0]
[2, 2, 1, 1, 0, 0, 0]
[2, 6, 5, 5, 4, 1, 1]
```



In [12]:

```
# 3 Circular convolution of x and y using frequency domain approach

x_FFT = np.fft.fft(x)
y_FFT = np.fft.fft(y)
print(x_FFT)
print(y_FFT)

z_FTT = x_FFT*y_FFT
print(z_FTT)

z_IFFT = np.fft.ifft(z_FTT)

print(z_IFFT)
```

```
[ 4.+0.j  1.-1.j -2.+0.j  1.+1.j]
[6.+0.j 1.-1.j 0.+0.j 1.+1.j]
[24.+0.j  0.-2.j -0.+0.j  0.+2.j]
[6.+0.j 7.+0.j 6.+0.j 5.+0.j]
```

In [15]:

```
# 4 Circular convolution of x and y using frequency domain approach
x1_FFT = np.fft.fft(x1)
y1_FFT = np.fft.fft(y1)
print(x1_FFT)
print(y1_FFT)
z1_FTT = x1_FFT*y1_FFT
print(z1_FTT)
z1_IFFT = np.fft.ifft(z1_FTT)
print(z1_IFFT)
```

```
[ 4.        +0.j          1.34601074-1.9975467j   1.17844793-1.16802434j
 -1.02445867-1.84269539j -1.02445867+1.84269539j  1.17844793+1.16802434j
```

```
  1.34601074+1.9975467j ]
[6.        +0.j         2.1234898 -2.97247462j 1.27747907-0.7341406j
 0.59903113-1.06086391j 0.59903113+1.06086391j 1.27747907+0.7341406j
 2.1234898 +2.97247462j]
[24.        +0.j         -3.0794168 -8.2427528j   0.64794847-2.35727312j
 -2.56853167-0.01702068j -2.56853167+0.01702068j  0.64794847+2.35727312j
 -3.0794168 +8.2427528j ]
[2.+0.j 6.+0.j 5.+0.j 5.+0.j 4.+0.j 1.+0.j 1.+0.j]
```

In [ ]: