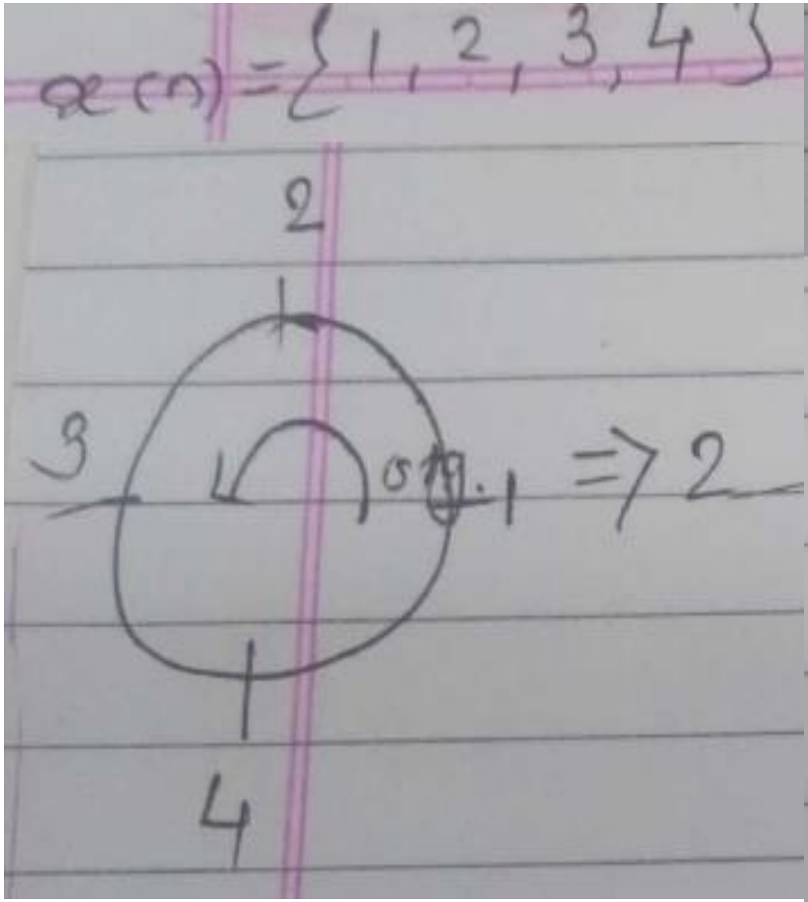
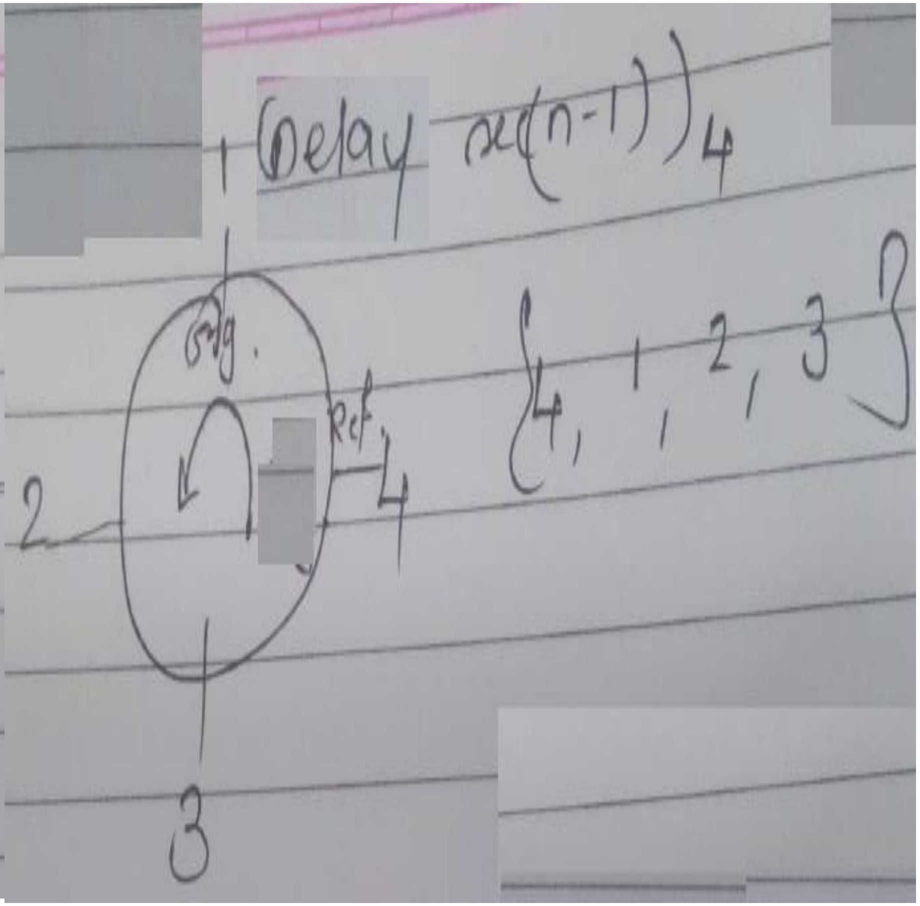
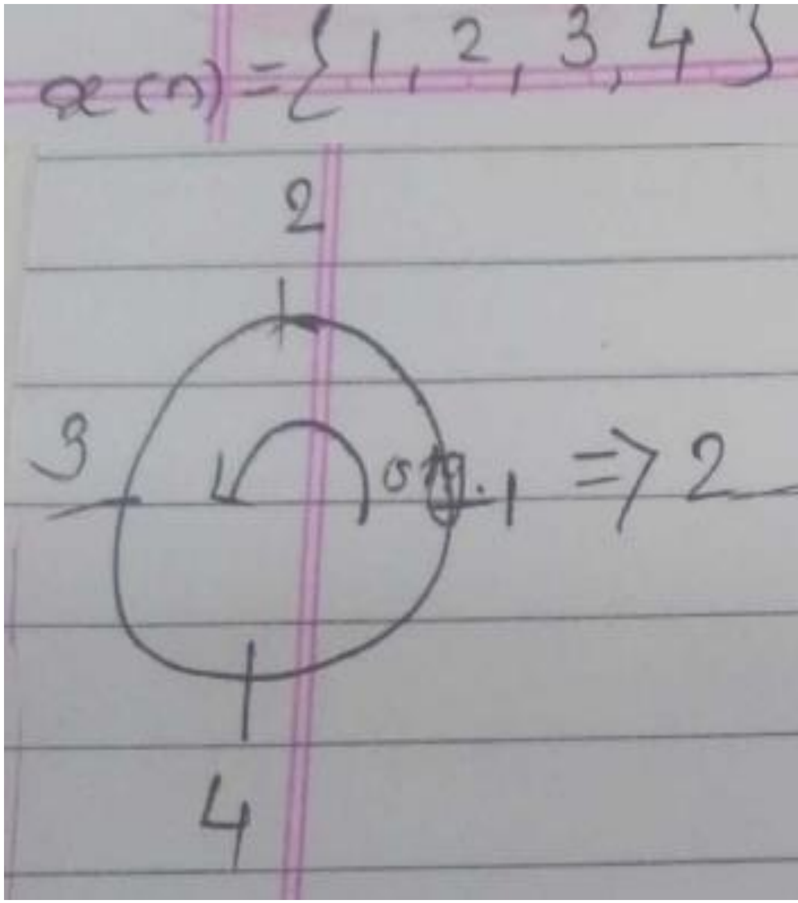
Lecture 3

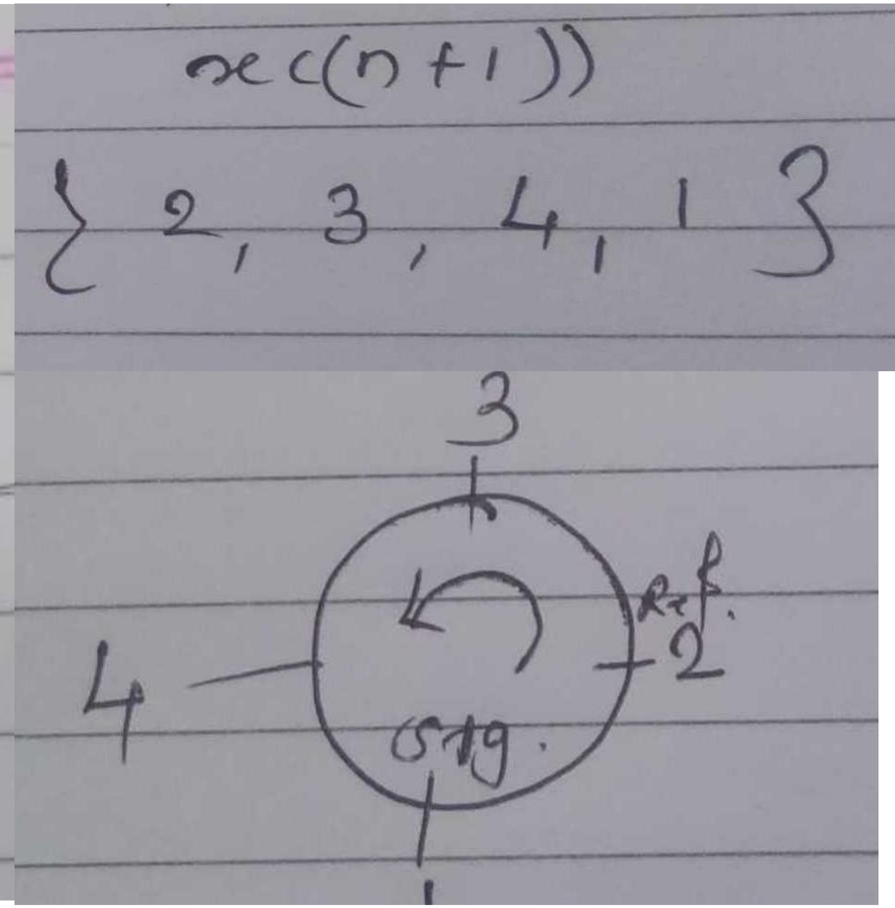
Circular shifting: Delaying



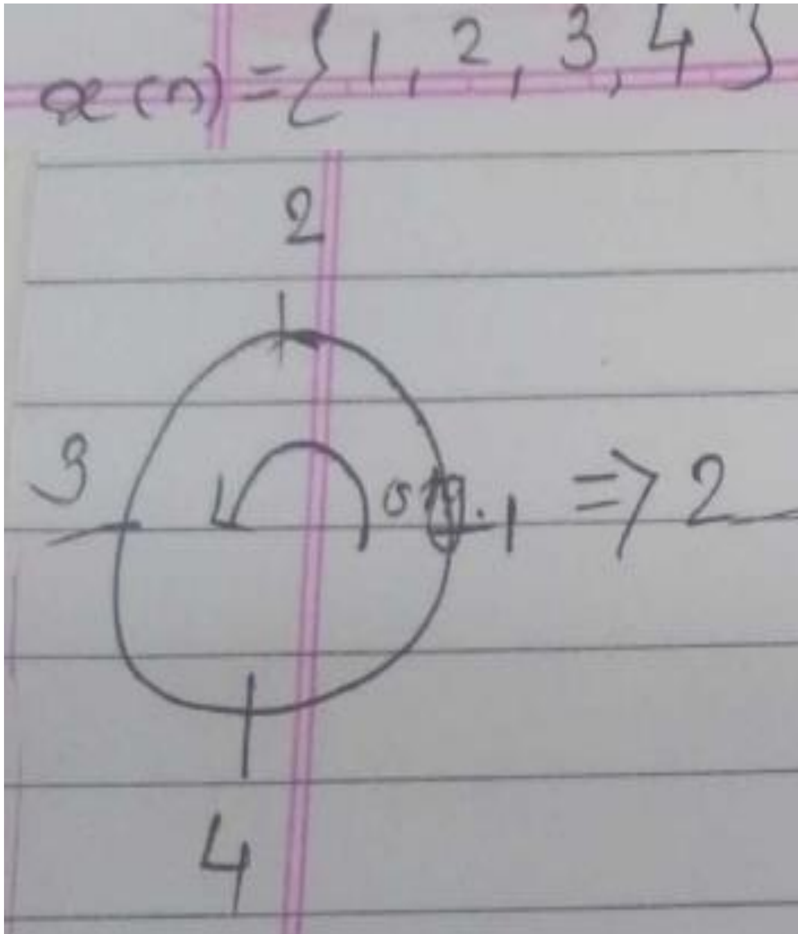


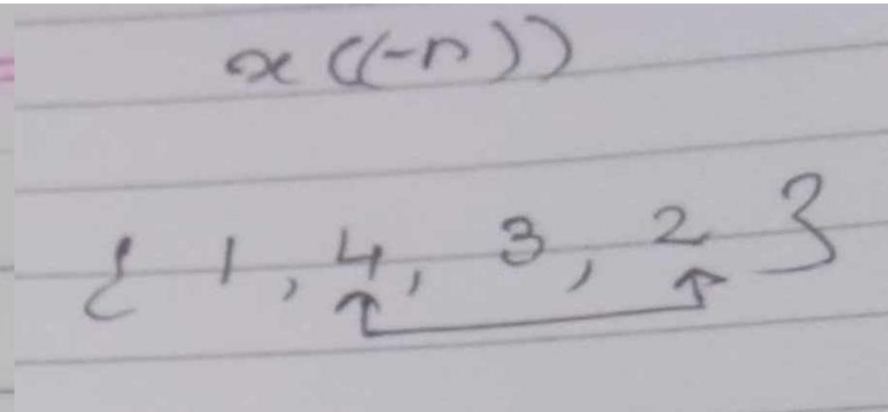
Circular shifting: Advancing

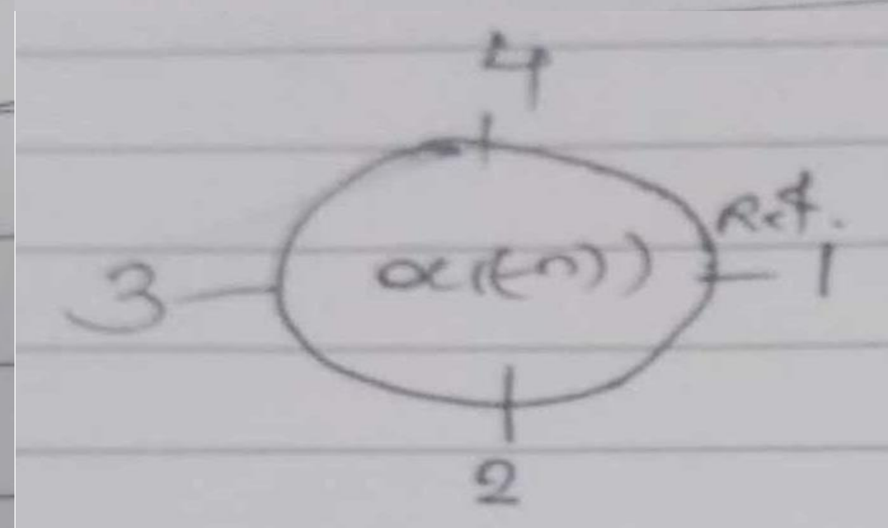




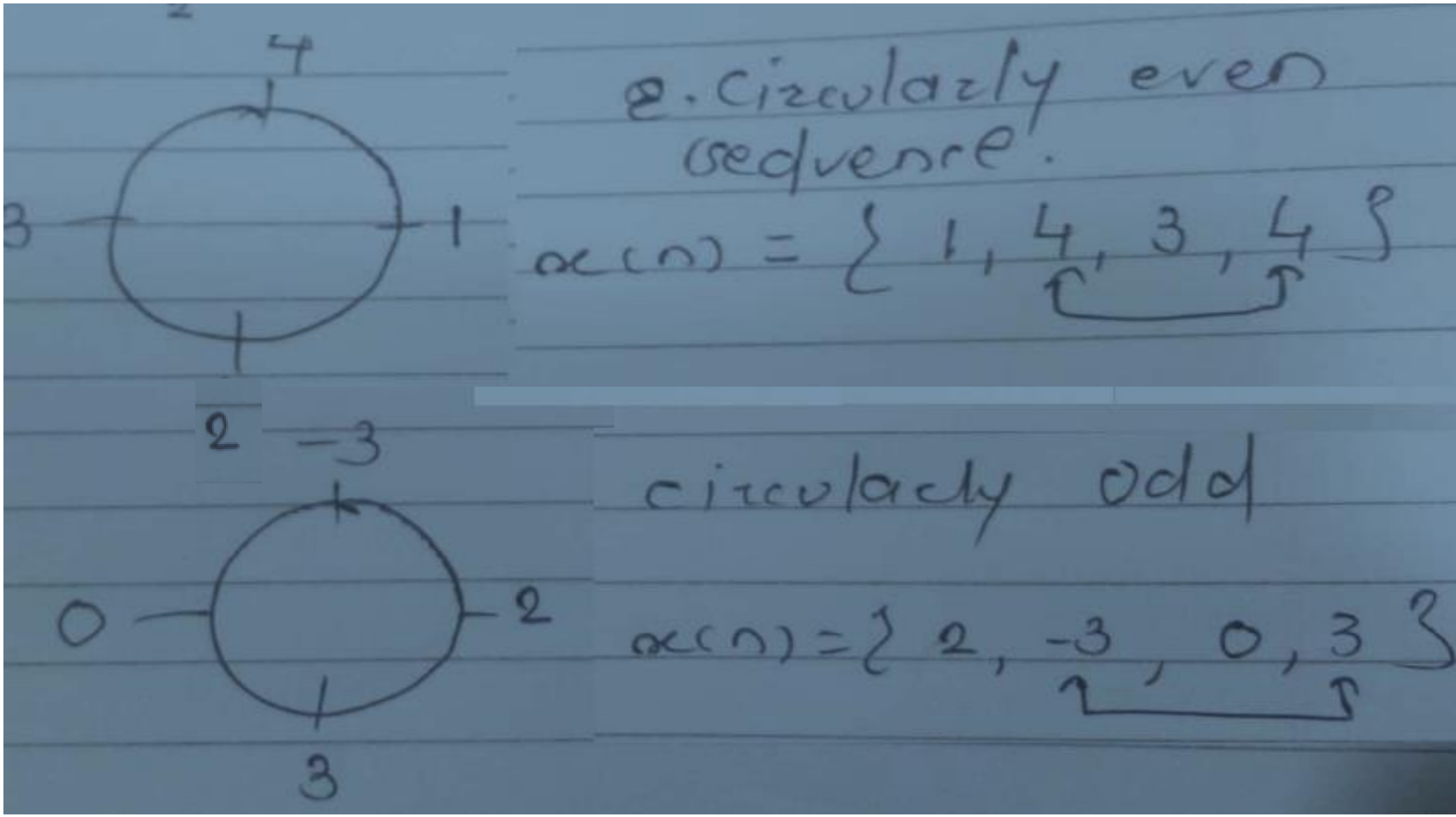
Circular Folding



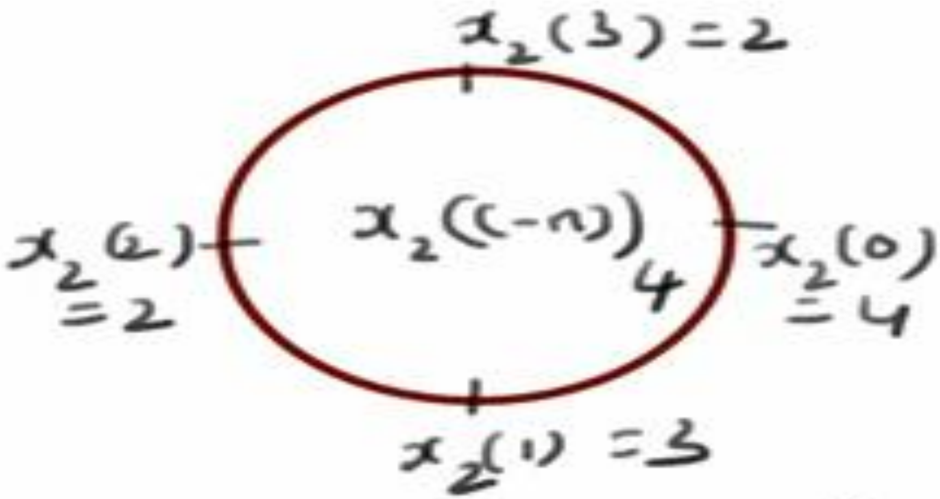




Circular Symmetry



$x_2(n) = \{4, 3, 2, 2\}$



Summary: Circular Operations

Sequence	Expression	Explanation
Input sequence	x((n))	Plot the samples of x(n) in anti-clockwise direction. Anti-clockwise means positive direction.
Circular delay	x((n – k))	Shift sequence x(n) in anticlock-wise direction by k samples.
Circular advance	x((n + k))	Shift sequence x(n) in clockwise direction by k samples.
Circular folding	x((– n))	Plot the samples of x(n) in clockwise direction. Clockwise means negative direction.
Circularly even	x(N – n) = x(n)	Sequence is symmetric about the point zero on the circle.
Circularly odd	x(N – n) = – x(n)	Sequence is anti-symmetric about the point zero on the circle.

- Methods of Circular Convolution
- Generally, there are two methods, which are adopted to perform circular convolution and they are −
- Concentric circle method,
- Matrix multiplication method.
- **Concentric Circle Method**
- Let x1(n)x1(n) and x2(n)x2(n) be two given sequences. The steps followed for circular convolution of x1(n)x1(n) and x2(n)x2(n) are
- Take two concentric circles. Plot N samples of x1(n)x1(n) on the circumference of the outer circle maintainingequaldistancesuccessivepointsmaintainingequaldistancesuccessivepoints in anti-clockwise direction.
- For plotting x2(n)x2(n), plot N samples of x2(n)x2(n) in clockwise direction on the inner circle, starting sample placed at the same point as 0^{th} sample of x1(n)x1(n)
- Multiply corresponding samples on the two circles and add them to get output.
- Rotate the inner circle anti-clockwise with one sample at a time.
- **Matrix Multiplication Method**
- Matrix method represents the two given sequence x1(n)x1(n) and x2(n)x2(n) in matrix form.
- One of the given sequences is repeated via circular shift of one sample at a time to form a N X N matrix.
- The other sequence is represented as column matrix.
- The multiplication of two matrices give the result of circular convolution.

Circular Convolution

- Circular convolution is defined by:

$$y_C[n] = \sum_{m=0}^{N-1} g[m]h[\langle n-m \rangle_N], \quad 0 \le n \le N-1$$

- N-point circular convolution is denoted as,

$$y[n] = g[n] \circledN h[n]$$

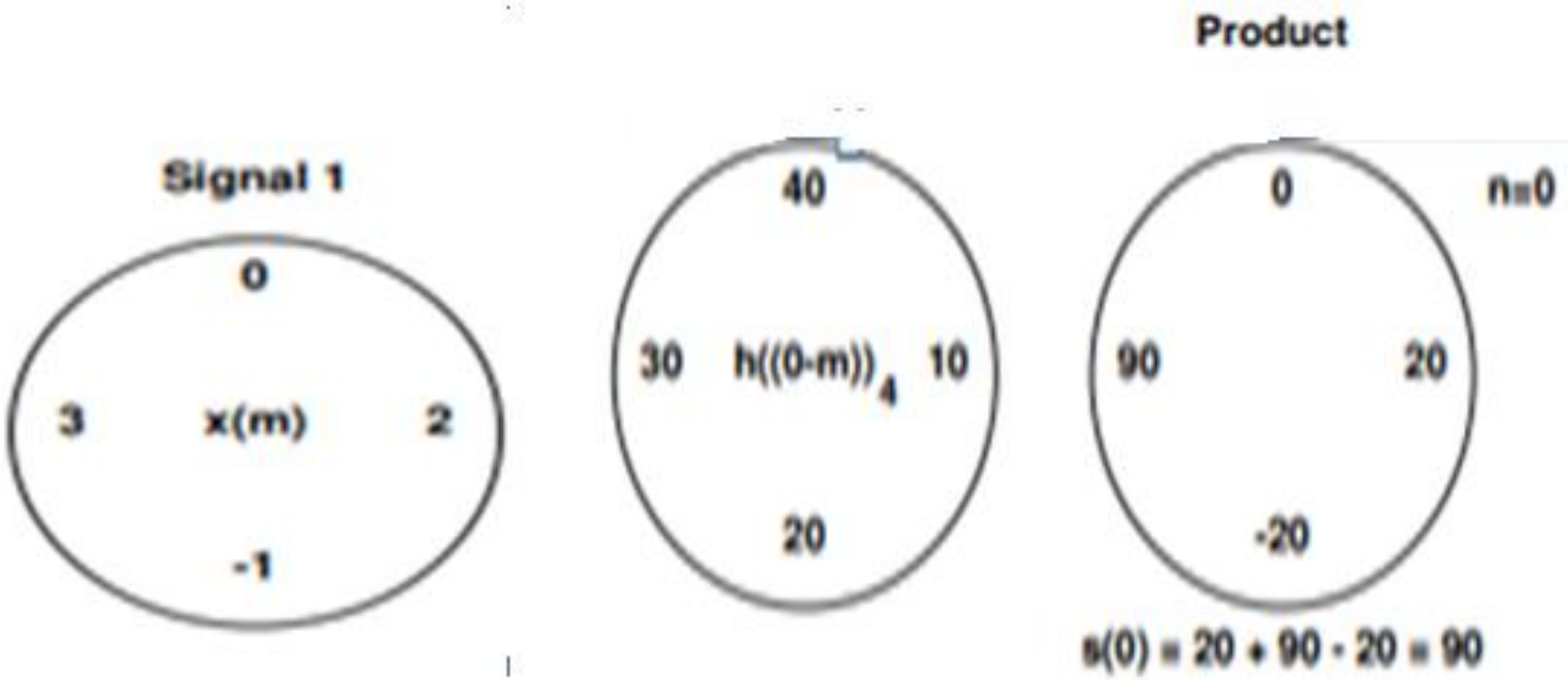
- The circular convolution is commutative, i.e

$$g[n] \circledN h[n] = h[n] \circledN g[n]$$

Circular Convolution: Method 1

x(n) = {2, 0, 3, -1}
h(n) = {10, 20, 30, 40}

N-point circular convolution $\sum_{m=0}^{N-1} g[m]h[\langle n-m\rangle_N]$

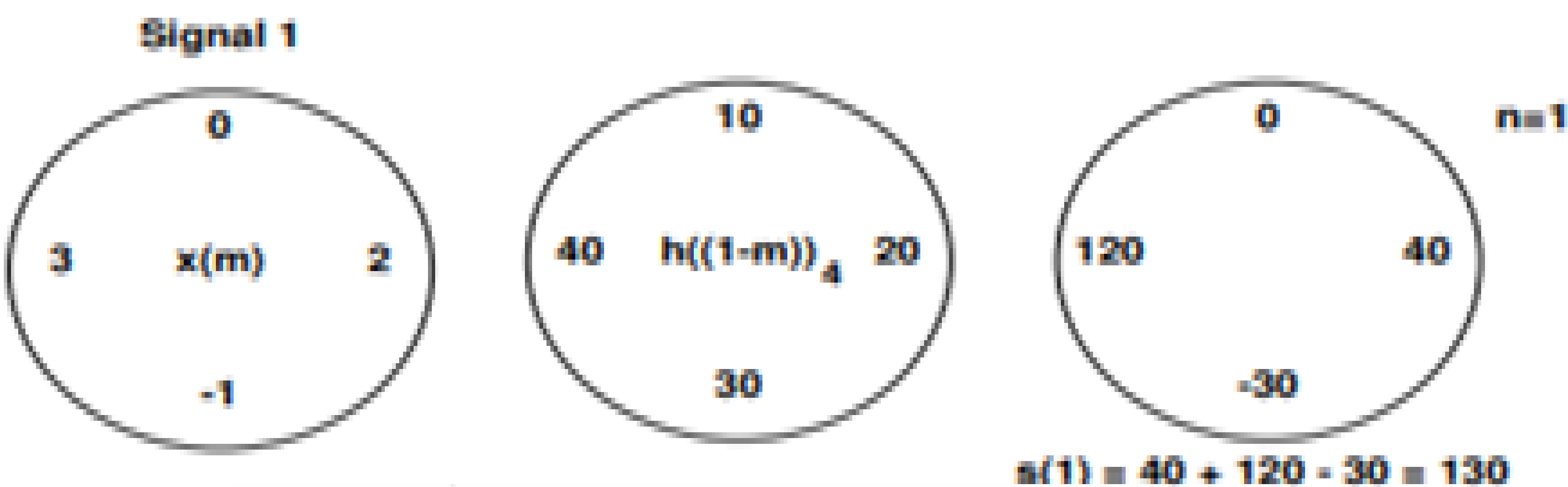


Circular Convolution: Graphical Method

x(n) = {2, 0, 3, -1}
h(n) = {10, 20, 30, 40}

Reversed Signal 2

Product



N-point circular convolution

$$\sum_{m=0}^{N-1} g[m]h[\langle n-m\rangle_N]$$

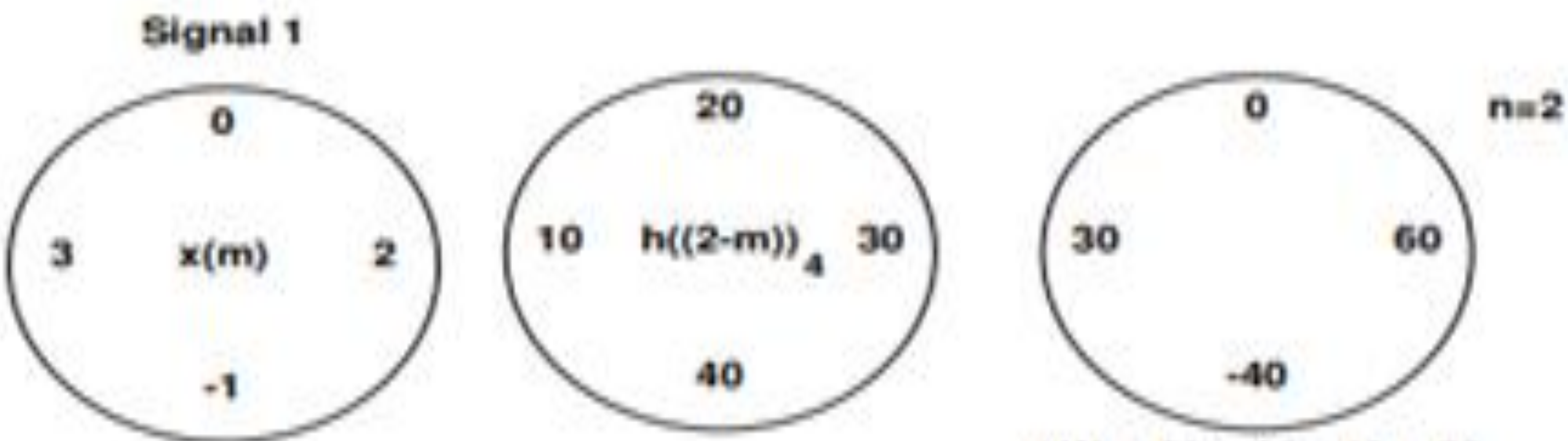
Circular Convolution: Graphical Method

x(n) = (2, 0, 3, -1)
h(n) = (10, 20, 30, 40)

N-point circular convolution $\sum_{m=0}^{N-1} g[m]h[\langle n-m \rangle_N]$

Reversed Signal 2

Product

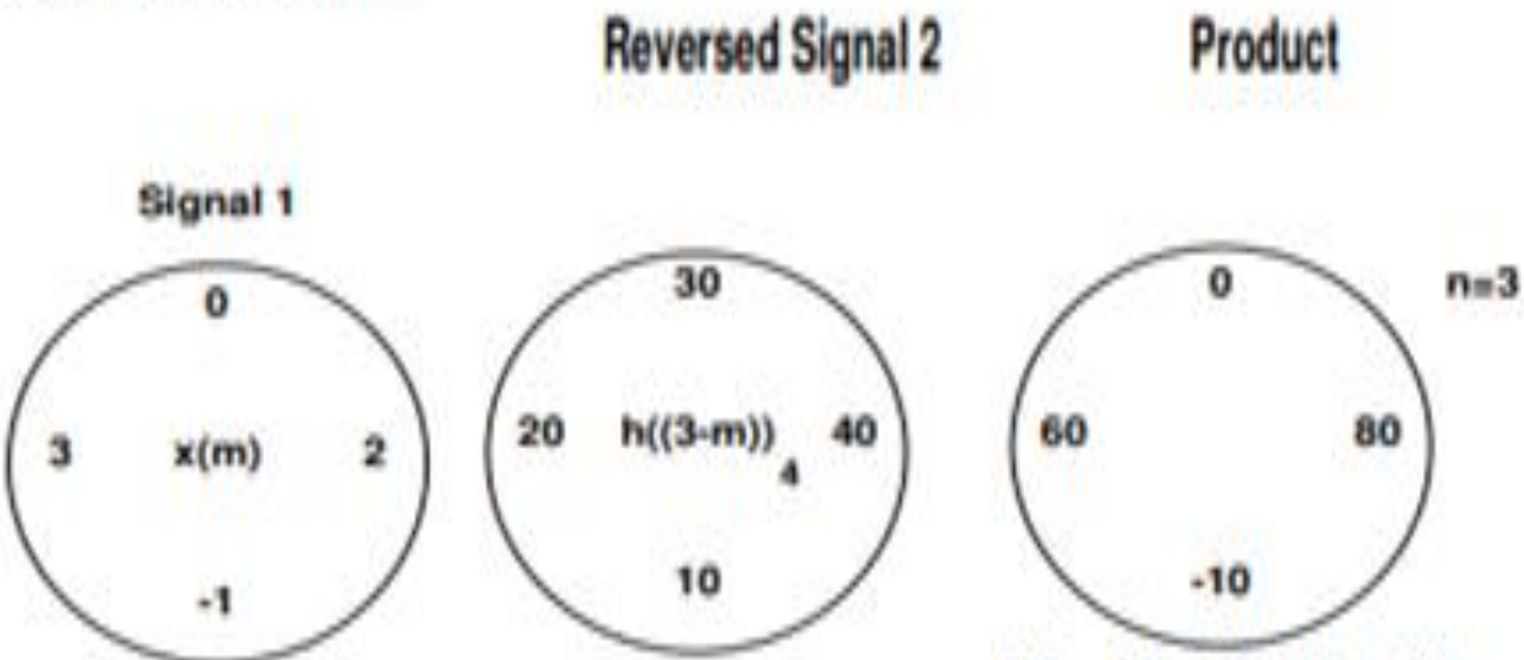


s(2) = 60 + 30 - 40 = 50

Circular Convolution: Graphical Method

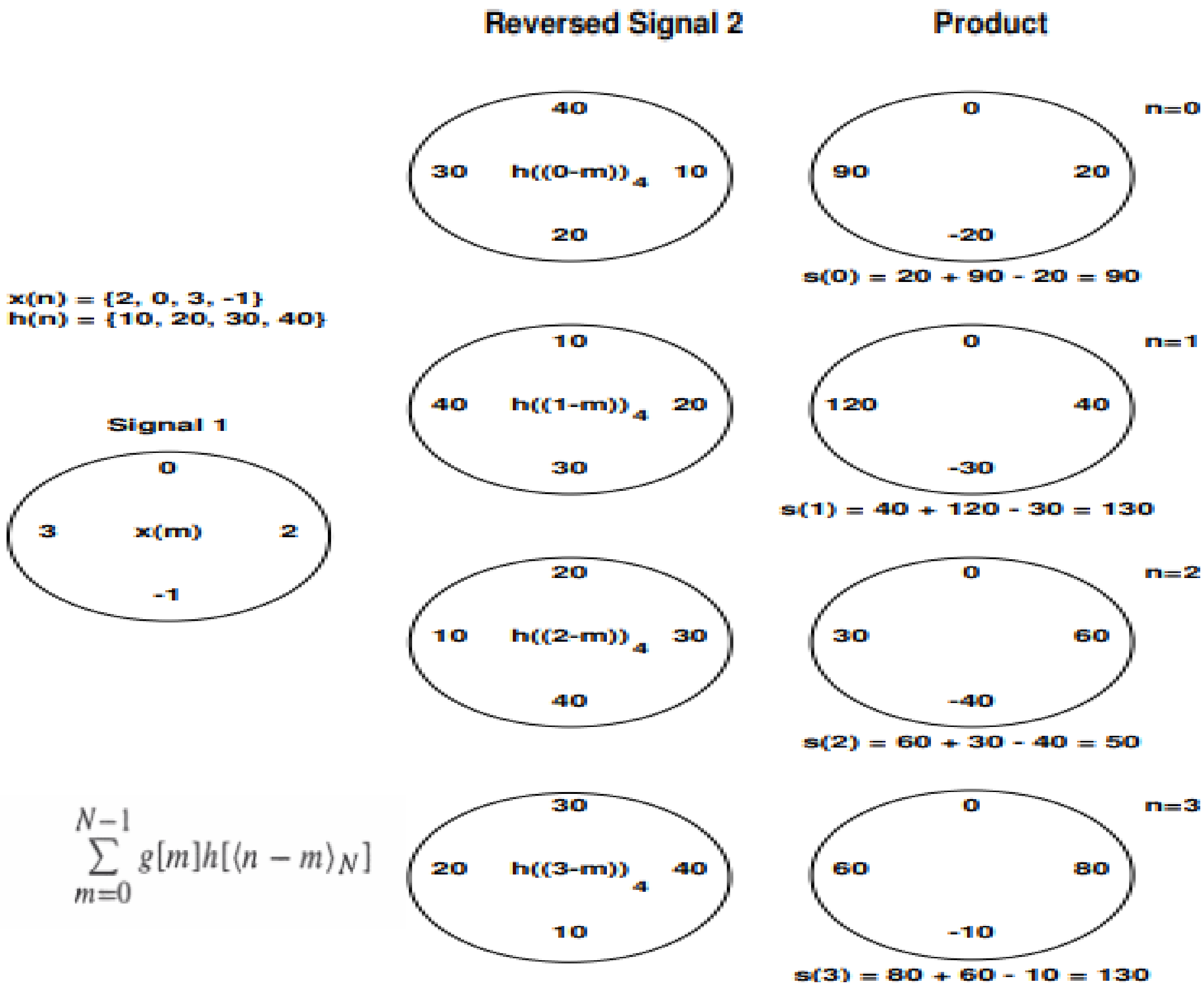
x(n) = (2, 0, 3, -1)
h(n) = (10, 20, 30, 40)

N-point circular convolution $\sum_{m=0}^{N-1} g[m]h[\langle n-m\rangle_N]$



s(3) = 80 + 60 - 10 = 130

Circular Convolution:-Graphical Method



N-point circular convolution

$$\sum_{m=0}^{N-1} g[m]h[\langle n-m\rangle_N]$$

Circular Convolution: Method 2

$$y(m) = x_2(n) \circledN x_1(n) = h(n) \circledN x(n)$$

$x(0) = 1, \quad x(1) = 2, \quad x(3) = 3, \quad x(3) = 1$

$h(0) = 4, \quad h(1) = 3, \quad h(2) = 2, \quad h(3) = 2$

$$\begin{bmatrix} y(0) \\ y(1) \\ y(2) \\ y(3) \end{bmatrix} = \begin{bmatrix} h(0) & h(3) & h(2) & h(1) \\ h(1) & h(0) & h(3) & h(2) \\ h(2) & h(1) & h(0) & h(3) \\ h(3) & h(2) & h(1) & h(0) \end{bmatrix} \begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ x(3) \end{bmatrix}$$

Circular convolution: Matrix Method

$$\begin{bmatrix} y(0) \\ y(1) \\ y(2) \\ y(3) \end{bmatrix} = \begin{bmatrix} 4 & 2 & 2 & 3 \\ 3 & 4 & 2 & 2 \\ 2 & 3 & 4 & 2 \\ 2 & 2 & 3 & 4 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 3 \\ 1 \end{bmatrix}$$

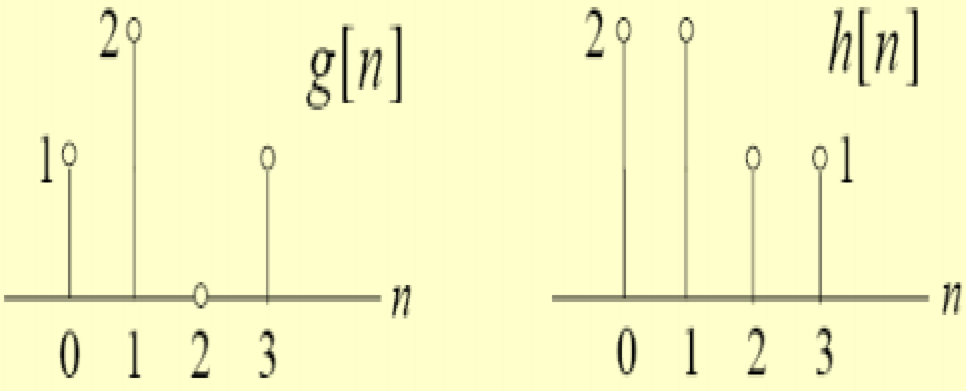
Circular Convolution: Matrix Method

$$\begin{bmatrix} y(0) \\ y(1) \\ y(2) \\ y(3) \end{bmatrix} = \begin{bmatrix} 4 & 2 & 2 & 3 \\ 3 & 4 & 2 & 2 \\ 2 & 3 & 4 & 2 \\ 2 & 2 & 3 & 4 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 3 \\ 1 \end{bmatrix} = \begin{bmatrix} 17 \\ 19 \\ 22 \\ 19 \end{bmatrix}$$

$y(m) = \{17, 19, 22, 19\}$ Ans.

Circular convolution using DFT

Example - Consider the two length-4 sequences repeated below for convenience:



Circular convolution using DFT

- The two 4-point DFTs can also be computed as

$$\begin{bmatrix} G[0] \\ G[1] \\ G[2] \\ G[3] \end{bmatrix} = \mathbf{D}_4 \begin{bmatrix} g[0] \\ g[1] \\ g[2] \\ g[3] \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 0 \\ 1 \end{bmatrix} =$$

$$\begin{bmatrix} H[0] \\ H[1] \\ H[2] \\ H[3] \end{bmatrix} = \mathbf{D}_4 \begin{bmatrix} h[0] \\ h[1] \\ h[2] \\ h[3] \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix} \begin{bmatrix} 2 \\ 2 \\ 1 \\ 1 \end{bmatrix} =$$

?

Circular convolution using DFT

The two 4-point DFTs can also be computed using the matrix relation given earlier

$$\begin{bmatrix} G[0] \\ G[1] \\ G[2] \\ G[3] \end{bmatrix} = \mathbf{D}_4 \begin{bmatrix} g[0] \\ g[1] \\ g[2] \\ g[3] \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 4 \\ 1-j \\ -2 \\ 1+j \end{bmatrix}$$

$$\begin{bmatrix} H[0] \\ H[1] \\ H[2] \\ H[3] \end{bmatrix} = \mathbf{D}_4 \begin{bmatrix} h[0] \\ h[1] \\ h[2] \\ h[3] \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix} \begin{bmatrix} 2 \\ 2 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 6 \\ 1-j \\ 0 \\ 1+j \end{bmatrix}$$

$$Y_C[k] = \begin{bmatrix} Y_C[0] \\ Y_C[1] \\ Y_C[2] \\ Y_C[3] \end{bmatrix} = \begin{bmatrix} G[0]H[0] \\ G[1]H[1] \\ G[2]H[2] \\ G[3]H[3] \end{bmatrix} = \quad ?$$

Circular convolution using DFT

$$\begin{bmatrix} y_C[0] \\ y_C[1] \\ y_C[2] \\ y_C[3] \end{bmatrix} = \frac{1}{4} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & j & -1 & -j \\ 1 & -1 & 1 & -1 \\ 1 & -j & -1 & j \end{bmatrix} \begin{bmatrix} 24 \\ -j2 \\ 0 \\ j2 \end{bmatrix} = \quad ?$$

Circular convolution using DFT

- A 4-point IDFT of $Y_C[k]$ yields

$$\begin{bmatrix} y_C[0] \\ y_C[1] \\ y_C[2] \\ y_C[3] \end{bmatrix} = = \frac{1}{4}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & j & -1 & -j \\ 1 & -1 & 1 & -1 \\ 1 & -j & -1 & j \end{bmatrix}\begin{bmatrix} 24 \\ -j2 \\ 0 \\ j2 \end{bmatrix} = \quad ?$$

Circular convolution using DFT

- A 4-point IDFT of $Y_C[k]$ yields

$$\begin{bmatrix} y_C[0] \\ y_C[1] \\ y_C[2] \\ y_C[3] \end{bmatrix} = = \frac{1}{4}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & j & -1 & -j \\ 1 & -1 & 1 & -1 \\ 1 & -j & -1 & j \end{bmatrix}\begin{bmatrix} 24 \\ -j2 \\ 0 \\ j2 \end{bmatrix} = \begin{bmatrix} 6 \\ 7 \\ 6 \\ 5 \end{bmatrix}$$

Linear convolution using Circular convolution

$$x(n) = (1, 2, 1), \; h(n) = (1, 2)$$

Here, length of $x(n) = L = 3$

and length of $h(n) = M = 2$

Therefore, $N = L + M - 1 = 3 + 2 - 1 = 4$

i.e., we have to calculate 4-point DFT, i.e., $N = 4$

Let us make length of x(n) and h(n) equal to 4 by adding zeros at end.

Hence, $x(n) = \{1, 2, 1, 0\}$

and $h(n) = \{1,2, 0,0\}$

We have, $X(k) = W_N \cdot x_N$

N-point circular convolution	$\sum_{m=0}^{N-1} g[m]h[\langle n-m\rangle_N]$	$G[k]H[k]$

Linear convolution using Circular convolution

$$\begin{bmatrix} X(0) \\ X(1) \\ X(2) \\ X(3) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 1 \\ 0 \end{bmatrix}$$

Linear convolution using Circular convolution

$$\begin{bmatrix} X(0) \\ X(1) \\ X(2) \\ X(3) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 4 \\ -2j \\ 0 \\ j \end{bmatrix}$$

Linear convolution using Circular convolution

$$H(k) = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 0 \\ 0 \end{bmatrix}$$

Linear convolution using Circular convolution

$$H(k) = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 3 \\ 1-2j \\ -1 \\ 1+2j \end{bmatrix}$$

$$H(k) = \{\underset{\uparrow}{3},\ 1-2j,\ -1,\ 1+2j\}$$

Linear convolution using Circular convolution

N-point circular convolution	$\sum_{m=0}^{N-1} g[m]h[\langle n-m\rangle_N]$	$G[k]H[k]$

$X(k) = \{4, -2j, 0, j\}$

$H(k) = \{3, 1-2j, -1, 1+2j\}$

$Y(k) = X(k)H(k)$

$Y(k) = \{4, -2j, 0, j\}.\{3, 1-2j, -1, 1+2j\}$

$Y(k) = \{12, -2j-4, 0, j-2\}$

Properties of DFT

Type of Property	Length-N Sequence	N-point DFT
	$g[n]$ $h[n]$	$G[k]$ $H[k]$
Linearity	$\alpha g[n] + \beta h[n]$	$\alpha G[k] + \beta H[k]$
Circular time-shifting	$g[\langle n - n_o \rangle_N]$	$W_N^{k n_o} G[k]$
Circular frequency-shifting	$W_N^{-k_o n} g[n]$	$G[\langle k - k_o \rangle_N]$
Duality	$G[n]$	$N g[\langle -k \rangle_N]$
N-point circular convolution	$\sum_{m=0}^{N-1} g[m] h[\langle n - m \rangle_N]$	$G[k] H[k]$
Modulation	$g[n] h[n]$	$\frac{1}{N} \sum_{m=0}^{N-1} G[m] H[\langle k - m \rangle_N]$