Name: **Zahra Soukhtedel**
Student Id: **98105138**

Assignment Number: **1**
Course: **Medical Imaging**

Problem 1

1.1

If $y[n] = x[n] * h[n]$, then $y[n-1] = x[n-1] * h[n-1]$?

When performing convolution, one sequence is shifted relative to the other. So, if $y[n]$ is the result of convolving $x[n]$ and $h[n]$, then to find $y[n-1]$, we would need to convolve $x[n-1]$ and $h[n-1]$. However, this is true only if $x[n]$ and $h[n]$ are "shift-invariant" or "time-invariant" sequences.

1.2

if $y(t) = x(t) * h(t)$, then $y(-t) = x(-t) * h(-t)$?

Let's analyze whether the equation $y(-t) = x(-t) * h(-t)$ holds true by calculating both sides. Let's calculate $y(-t)$ using the convolution formula:

$$y(-t) = \int_{-\infty}^{\infty} x(\tau) \cdot h(-t-\tau)\, d\tau$$

On the other hand, let's calculate $x(-t) * h(-t)$ using the convolution formula:

$$x(-t) * h(-t) = \int_{-\infty}^{\infty} x(-\tau) \cdot h(-t-\tau)\, d\tau$$

By comparing the two expressions, we observe that $y(-t)$ and $x(-t) * h(-t)$ have the same integrand, but the integration variable differs (τ in $y(-t)$ and $-\tau$ in $x(-t) * h(-t)$).
Therefore, in general, $y(-t)$ is not equal to $x(-t) * h(-t)$. The convolution operation is not symmetric with respect to time reversal. Shifting the input functions by t or $-t$ leads to different integrals, and as a result, the equality $y(-t) = x(-t) * h(-t)$ does not hold true for the convolution operation.

Problem2

Given two LTI systems. S_1 has input x and output w. S_2 has input w and output y. The systems are causal and described by the following difference equations:

$$S_1 : w[n] = \frac{1}{2}w[n-1] + x[n]$$

$$S_2 : y[n] = \alpha y[n-1] + \beta w[n]$$

Given:

$$y[n] = (-\frac{1}{8})y[n-2] + \frac{3}{4}y]n-1] + x[n]$$

2.1

find the values of α and β.

To find the values of α and β in the difference equation for S_2, we can compare it with the given equation for $y[n]$. We can observe that the coefficients of the terms involving $y[n]$ match. Therefore, we can equate the coefficients to find the values of α and β.
Coefficient of $y[n-1]$: Comparing with the equation for S_2, we have $\alpha = \frac{3}{4}$.
Coefficient of $w[n]$: Comparing with the equation for S_2, we have $\beta = 1$.

2.2

Find the cascade impulse response of two systems S_1 and S_2.

The cascade impulse response of two systems S_1 and S_2 is the convolution of their impulse responses. Let's find each system's impulse response by setting the input of each system to the Kronecker delta function $\delta[n]$ in the difference equation, therefor we have:

$$x[n] = \delta[n] \Rightarrow w[n] = \frac{1}{2}w[n-1] + \delta[n] \Rightarrow h_1[n] = \left(\frac{1}{2}\right)^n \delta[n]$$

$$w[n] = \delta[n] \Rightarrow y[n] = \alpha y[n-1] + \beta\delta[n] \Rightarrow h_2[n] = \alpha^n \beta\delta[n]$$

Now to find the cascade impulse response $h[n]$ of the two systems, we need to convolve $h_1[n]$ and $h_2[n]$:

$$h[n] = h_1[n] * h_2[n] = \left(\frac{1}{2}\right)^n \alpha^n \beta\delta[n] * \delta[n]$$

Since the convolution of two delta functions is a delta function, we can simplify further:

$$h[n] = \left(\frac{1}{2}\right)^n \alpha^n \beta\delta[n]$$

from the previous section, we conclude that $\alpha = \frac{3}{4}, \beta = 1$, therefor we have:

$$h[n] = \left(\frac{3}{8}\right)^n \alpha^n \beta\delta[n]$$

Problem 3

$x_1[n] = 1 + \sin(\frac{3\pi}{8}n + \frac{\pi}{4})$ $x_2[n] = \sum_{k=-\infty}^{+\infty}(\frac{1}{1})^{n-4k}u[n-4k]$ compute the output of two systems with inputs x_1 and x_2 on the filter below

$$h[w] = -u[w + \frac{\pi}{3}] + u[w + \frac{5\pi}{12}] + u[w - \frac{\pi}{3}] - u[w - \frac{5\pi}{12}]$$

To compute the output of the two systems with inputs $x_1[n]$ and $x_2[n]$ using the given filter $h[w]$, we need to convolve the input signals with the impulse response of the filter.
Let's start with the first input signal $x_1[n] = 1 + \sin(\frac{3\pi}{8}n + \frac{\pi}{4})$:

$$y_1[n] = x_1[n] * h[n]$$

where $*$ denotes the convolution operation.
Next, let's compute the convolution of $x_1[n]$ with $h[n]$:

$$y_1[n] = \sum_{k=-\infty}^{+\infty} x_1[k] \cdot h[n-k]$$

Substituting the values of $x_1[n]$ and $h[n]$ into the convolution equation, we get:

$$y_1[n] = \sum_{k=-\infty}^{+\infty} (1+\sin(\frac{3\pi}{8}k+\frac{\pi}{4})) \cdot (-u[n-k+\frac{\pi}{3}]+u[n-k+\frac{5\pi}{12}]+u[n-k-\frac{\pi}{3}]-u[n-k-\frac{5\pi}{12}])$$

Now, let's compute the second input signal $x_2[n] = \sum_{k=-\infty}^{+\infty}(\frac{1}{1})^{n-4k}u[n-4k]$:

$$y_2[n] = x_2[n] * h[n]$$

Again, we need to convolve $x_2[n]$ with $h[n]$:

$$y_2[n] = \sum_{k=-\infty}^{+\infty} x_2[k] \cdot h[n-k]$$

Substituting the values of $x_2[n]$ and $h[n]$ into the convolution equation, we get:

$$y_2[n] = \sum_{k=-\infty}^{+\infty} \left(\sum_{m=-\infty}^{+\infty} (\frac{1}{1})^{k-4m} u[k-4m] \right) \cdot (-u[n-k+\frac{\pi}{3}]+u[n-k+\frac{5\pi}{12}]+u[n-k-\frac{\pi}{3}]-u[n-k-\frac{5\pi}{12}])$$

Problem 4

Given the convolution of 2 pictures with size $N * N$ as follows:

$$(x * y)[m,n] = \sum_{m'=0}^{N-1} \sum_{n'=0}^{N-1} x[m',n'] \cdot y[m-m', n-n']$$

1. Find the order of time calculation of this value.
2. Introduction of intermediate calculations, suggest a solution that can be used to calculate the above answer in time order $O(N^2 \log(N))$ and prove the said time order for your solution (hint: use Fourier transform).

4.1

The time complexity of the given convolution operation is $O(N^4)$. This is because, for every pixel in the output image, we have to do a double summation over all the pixels in the input images, resulting in N^2 multiplications and additions. Since there are N^2 pixels in the output image, the total time complexity becomes $O(N^4)$.

4.2

Based on the Fourier transform, we know that convolution in the spatial domain is equivalent to multiplication in the frequency domain. So we perform the following steps:

1.Perform the Fourier transform on both images using the FFT algorithm. The FFT algorithm has a time complexity of $O(N^2 \log(N))$ for each image, so the time complexity for this part is $O(N^2 \log(N))$.
2.Multiply the transformed images element-wise. This process takes $O(N^2)$ time.
3.Perform the inverse Fourier transform on the resulting image using the IFFT algorithm. The IFFT algorithm also has a time complexity of $O(N^2 \log(N))$.
4.Normalize the resulting image by dividing each pixel by N^2. This process takes $O(N^2)$ time.

Therefore, the overall time complexity of this solution is $O(N^2 \log(N))$.

Problem 5

1. Describe some of the key components of the IO file. How do they help to comprehensively display medical imaging data? What is the use of the two tags rescae interept and resae sope?
2. Describe in detail the method of anonymizing information in IO. Also, suggest the pyio code for this, and tell its modules in order, and state their functionality.

5.1

The DICOM (Digital Imaging and Communications in Medicine) file format is specifically designed for storing and transmitting medical imaging data. It comprises several key components that contribute to the comprehensive display of medical imaging data:

1. File Meta Information: This section contains metadata about the DICOM file itself, including details such as the DICOM version, transfer syntax, and character set. This information is crucial for correctly interpreting the file.
2. Data Elements: These are the fundamental elements of the DICOM file. Each data element consists of a tag, value representation (VR), value length, and value. Tags uniquely identify each element and provide information about attributes such as patient information, imaging modality, image dimensions, pixel data, and more. Data elements can be organized into modules, which represent specific aspects of the medical imaging study, such as patient information, study details, image acquisition parameters, etc.
3. Pixel Data: This component stores the actual image data captured by the medical imaging device, such as X-ray, MRI, CT scan, etc. The pixel data is typically represented as a 2D or 3D array of intensity values. DICOM supports various data types for pixel data, including grayscale and color images. The pixel data component enables the comprehensive display of medical images, facilitating visualization and analysis.

The two tags, **rescale intercept** and **rescale slope**, play a crucial role in transforming the pixel values stored in the DICOM file to their real-world physical values. Medical imaging devices often capture images using a specific intensity range that may not directly correspond to the actual physical values being measured. The rescale intercept and rescale slope values allow for the conversion of pixel values to their true physical values using the formula:

Physical Value = Pixel Value * Rescale Slope + Rescale Intercept

The rescale intercept represents the offset or shift applied to the pixel values, while the rescale slope represents the scaling factor. By applying these transformations, the DICOM file accurately represents the measured physical quantities, enabling comprehensive display and analysis of medical imaging data. This ensures that the images are displayed with the correct intensity and accurately reflect the actual physical properties being measured.

5.2

Anonymizing information in DICOM (Digital Imaging and Communications in Medicine) involves removing or de-identifying patient-specific and sensitive information from the DICOM file to protect patient privacy. This process ensures compliance with privacy regulations and allows for the secure sharing and analysis of medical imaging data.
The method of anonymizing information in DICOM typically involves the following steps:

1. Patient Identification Removal: The patient's identifying information, such as name, ID, birthdate, and address, is removed or replaced with generic or anonymized values. This step ensures that the patient's identity cannot be easily determined from the DICOM file.
2. Study and Series Anonymization: Information related to the study and series, such as study description, study date, and series description, may also be anonymized to further protect patient privacy. This step helps prevent the identification of specific medical procedures or examinations.
3. Image Pixel Data Preservation: The pixel data, which represents the actual medical images, is typically left intact during the anonymization process. This ensures that the diagnostic quality of the images is not compromised.
4. Metadata Preservation: Certain metadata, such as modality, image orientation, and image position, are usually preserved to maintain the integrity and interpretability of the DICOM file.
5. Consistency Checks: After anonymization, the DICOM file is checked to ensure that it remains valid and conforms to the DICOM standard. This step helps prevent any unintended modifications or corruptions during the anonymization process.

It is important to note that the specific anonymization steps and techniques may vary depending on the requirements and regulations of the healthcare institution or jurisdiction. Additionally, it is crucial to consult legal and privacy experts to ensure compliance with applicable laws and regulations when anonymizing DICOM information.

Anonymizing information plays a crucial role in protecting patients' privacy when working with DICOM (*Digital Imaging and Communications in Medicine*) files. It involves removing or de-identifying sensitive personal information from the file while retaining essential clinical data. There are specific DICOM tags that contain identifying patient information, such as name, date of birth, patient ID, etc. Anonymization aims to remove or replace these tags with generic or non-identifying values. Here's a general method for anonymizing DICOM information:

1. Load the DICOM file: Read the DICOM file using a DICOM library or module. This will enable you to access and modify the DICOM tags.

2. Identify sensitive tags: Identify the DICOM tags that contain sensitive information. These tags are commonly known as private or confidential tags and vary depending on the specific DICOM dataset.

3. Anonymize personal information: Replace the sensitive tags with generic or random values. For example, you can replace the patient's name with a generic name like "ANONYMOUS" or generate a random patient ID.

4. Preserve clinical information: Ensure that essential clinical information, such as study description, study date, imaging modality, etc., remains intact. These details are necessary for accurate interpretation and analysis.

5. Save the anonymized file: Save the modified DICOM file with the anonymized information. This new file should no longer contain sensitive patient information, safeguarding patient privacy.

As for the PyDICOM code, PyDICOM is a widely used Python library for working with DICOM files. Here's an example code snippet to anonymize DICOM files using PyDICOM:

```
import pydicom

# Load the DICOM file
dataset = pydicom.dcmread('input.dcm')

# Identify and anonymize sensitive tags
dataset.PatientName = 'ANONYMOUS'
dataset.PatientID = '12345'

# Save the anonymized file
dataset.save_as('output.dcm')
```

The code snippet above demonstrates a basic approach to anonymize DICOM files using PyDICOM. However, note that the specific tags to anonymize may vary depending on your requirements and the DICOM dataset you are working with.
The essential module for anonymization using PyDICOM is `pydicom`. It provides functionalities for reading, modifying, and saving DICOM files. Other commonly used modules in DICOM processing include:

- `os`: This module is useful for file handling, such as reading and writing files.
- `glob`: The `glob` module is used for pattern matching and retrieving a list of files in a directory.
- `numpy`: The `numpy` module is often used for efficient array operations and manipulation in DICOM pixel data.

These modules, along with `pydicom`, work together to facilitate the anonymization and general processing of DICOM files.