Programming task 1: FIR filtering

1. Opening of the WAV

```
    #To open the wave file and get parameters

    file = filedialog.askopenfilename(title='Select File', filetype=(('WAV File (*.wav)', '*.wav'),))
    wavefile = wave.open(file, 'rb')
    parameters = wavefile.getparams()

    #Reading the data and header

    nchannels, samplewidth, framerate, nframes = parameters[:4]
    data = wavefile.readframes(nframes) #Reading the wave data

    fileforheader = open(file,'rb')
    buffer = fileforheader.read(44) #Reading the header
    header = np.frombuffer(buffer,dtype=np.int16) #Formatting the header
    wavdata = np.frombuffer(data, dtype=np.int16) #Formatting the data
```

2. The FIR filtering

```
   #Convolution - FIR Filter

    for n in range(N): #Comes from Y
        for k in range(N2): #Comes from H
            y[n] = y[n]+x[n-k]*h[k] #Convolution

    print('Output y =\n',y)
```

3. Saving of the new WAV

```
   #Save the wave file

    f = open("FIR Filtered.wav", "wb") #Create a new wave file
    f.write(np.hstack((header, finaly)).tobytes()) #Concatenate header and output
```

Programming task 2: IIR filtering

1. Opening of the WAV

```
    #To open the wave file and get parameters

    file = filedialog.askopenfilename(title='Select File', filetype=(('WAV
File (*.wav)', '*.wav'),)) #For Opening WAV file
    wavefile = wave.open(file, 'rb')
    parameters = wavefile.getparams()

    #Reading the data and header

    nchannels, samplewidth, framerate, nframes = parameters[:4]
    data = wavefile.readframes(nframes) #Reading the wave data

    fileforheader = open(file,'rb')
    buffer = fileforheader.read(44) #Reading the header
    header = np.frombuffer(buffer,dtype=np.int16) #Formatting the header
    wavdata = np.frombuffer(data, dtype=np.int16) #Formatting the data
```

2. The IIR filtering

```
   #Filtering with Difference Equation

   for m in range(len(a)): #Repeat until all coefficient values are
inserted

        e = a[m]*a[m] + b[m]*b[m] #Multiplying the conjugate for zeros
        f = c[m]*c[m] + d[m]*d[m] #Multiplying the conjugate for poles

        for n in range(N1-1): #Calculate Impulse Response
            y[n] = x[n] - (2*a[m]*(x[n-1])) + (e*(x[n-2])) + (2*c[m]*(y[n-
1])) - (f*y[n-2]) #Difference equation
        x = y
```

3. Saving of the new WAV

```
   #Save the wave file

    f = open("IIR Filtered.wav", "wb") #Create a new wave file
    f.write(np.hstack((header, finaly)).tobytes()) #Concatenate header and
output
```