Question 1 (CT representation of the composition)

The continuous time representation of the function of the composition shown in figure 1 is given below:

$$
\begin{aligned}
&s(t) =(0\times(u(t-0)-u(t-0.125)))+\sin (440\pi \times 2^{10/12}) \times (u(t-0.125)-u(t-0.25)))\\
&+(0 \times (u(t-0.25)-u(t-0.375)))+ \sin (440\pi \times 2^{10/12}) \times (u(t-0.375)-u(t-0.625)))\\
&+ (0 \times (u(t-0.625)-u(t-0.75)))+ \sin (440\pi \times 2^{6/12}) \times (u(t-0.75)-u(t-1.25)))\\
&+ (0\times (u(t-1.25)-u(t-1.375)))+ \sin (440\pi \times 2^{8/12}) \times (u(t-1.375)-u(t-1.5)))\\
&+ (0\times (u(t-1.5)-u(t-1.625)))+ \sin (440\pi \times 2^{8/12}) \times (u(t-1.625)-u(t-1.875)))\\
&+ (0\times (u(t-1.875)-u(t-2)))+ \sin (440\pi \times 2^{5/12}) \times (u(t-2)-u(t-2.5)))
\end{aligned}
\tag{1}
$$

Question 2 (DT representation of the composition)

Mathematically, the DT representation of an analog signal $x(t)$ is

$$x[n]=x(nT_s) \tag{2}$$

whereas $T_s = 1/f_s$

T_s = Sampling period
f_s = Sampling frequency

The sampling frequency for this particular composition was 8 KHz. To convert the CT function to a valid DT function, the following code was written in MATLAB:

```
>> clear all
>> n=1:1:8000*19*2/8;
>> t=n/8000;
>> s1=(0.*(u(t-0)-u(t-0.125)));
s2= ((sin ((2*pi*220*(2^ (10/12))).*t)).*(u(t-0.125)-u(t-
0.25)));
s3=(0.*(u(t-0.25)-u(t-0.375)));
s4=((sin ((2*pi*220*(2^ (10/12))).*t)).*(u(t-0.375)-u(t-
0.625)));
s5=(0.*(u(t-0.625)-u(t-0.75)));
s6=((sin ((2*pi*220*(2^ (6/12))).*t)).*(u(t-0.75)-u(t-
1.25)));
s7=(0.*(u(t-1.25)-u(t-1.375)));
s8=((sin ((2*pi*220*(2^ (8/12))).*t)).*(u(t-1.375)-u(t-
1.5)));
s9=(0.*(u(t-1.5)-u(t-1.625)));
s10=((sin ((2*pi*220*(2^ (8/12))).*t)).*(u(t-1.625)-u(t-
1.875)));
s11=(0.*(u(t-1.875)-u(t-2)));
s12=((sin ((2*pi*220*(2^ (5/12))).*t)).*(u(t-2)-u(t-2.5)));
```

```
>> s=s1+s2+s3+s4+s5+s6+s7+s8+s9+s10+s11+s12;
```

Question 3 (Playing the composition)

The ‘s’ in the command of Question 2 is a row vector and is a DT representation of the of the musical notes shown in Figure 1. To play the composition, the following command was written. The signal was scaled before it was saved as a sound.

```
>> wavwrite (s,'naheed');
Warning: Data clipped during write to file:naheed
> In wavwrite>PCM_Quantize at 247
  In wavwrite>write_wavedat at 267
  In wavwrite at 112
>> y=wavread ('nahid');
>> sound (y)
```

Question 4 (Decay of each musical notes over time)

To have the volume of each note decay over time, each musical note representation was multiplied with a shifted exponential function (e^{-t}). The vector representation for new composition is“se” and the composition was played to understand the difference between the two compositions.

```
>> se1= ((0.*(u(t-0)-u(t-0.125)))).*(exp(-t));
se2= (((sin ((2*pi*220*(2^ (10/12))).*t)).*(u(t-0.125)-u(t-
0.25)))) .*(exp(-(t-0.125)));
se3= ((0.*(u(t-0.25)-u(t-0.375)))) .*(exp(-(t-0.25)));
se4= (((sin ((2*pi*220*(2^ (10/12))).*t)).*(u(t-0.375)-u(t-
0.625)))) .*(exp(-(t-0.375)));
se5= ((0.*(u(t-0.625)-u(t-0.75)))) .*(exp(-(t-0.625)));
se6= (((sin ((2*pi*220*(2^ (6/12))).*t)).*(u(t-0.75)-u(t-
1.25)))) .*(exp(-(t-0.75)));
se7=((0.*(u(t-1.25)-u(t-1.375)))) .*(exp(-(t-1.25)));
se8=(((sin ((2*pi*220*(2^ (8/12))).*t)).*(u(t-1.375)-u(t-
1.5)))) .*(exp(-(t-1.375)));
se9=((0.*(u(t-1.5)-u(t-1.625)))) .*(exp(-(t-1.5)));
se10=(((sin ((2*pi*220*(2^ (8/12))).*t)).*(u(t-1.625)-u(t-
1.875)))) .*(exp(-(t-1.625)));
se11=((0.*(u(t-1.875)-u(t-2)))) .*(exp(-(t-1.875)));
se12=(((sin ((2*pi*220*(2^ (5/12))).*t)).*(u(t-2)-u(t-
2.5)))) .*(exp(-(t-2)));
>> se=se1+se2+se3+se4+se5+se6+se7+se8+se9+se10+se11+se12;

>> wavwrite (se,'mahmud');
Warning: Data clipped during write to file:mahmud
> In wavwrite>PCM_Quantize at 247
```

```
  In wavwrite>write_wavedat at 267
  In wavwrite at 112
>> ye=wavread ('mahmud');
>> sound (ye)
```

Question 5 (Original Signal and the Echo)

A simple analytical model of the echo process is

$$s_e (t) = \alpha (t-T) . s(t-T) \tag{3}$$

Whereas $s_e (t)$ = echo
$\alpha (t)$ = function that attenuates the volume over time, $\alpha (t) \in [0,1)$
T= time delay introduced by reflection, $T \geq 0$

The equation also implies that echo lags the original signal and is weaker than the original signal.

For this project, $\alpha (t)$ was assumed to be a constant and it was equal to 0.1 second. The time delay was multiplied by the sampling frequency.

The sound that is heard is represented by the following equation:

$$r(t) = s(t) + s_e (t) \tag{4}$$

whereas $s(t)$ = original signal
$s_e (t)= \alpha (t-T) . s(t-T)$ = echo

The process could be illustrated through the following block diagram shown in Figure 2*:*

Figure 2: Model of the echo process

In Matlab, a zero vector was created and the elements of this vector were placed at the end and beginning of the vectors representing original signal (se) and the echo repectively. The original signal is twice the amplitude of the echo.

The matlab command is given below.

```
>> Delay=0.1;
>> fxDelay=0.1*8000

fxDelay =

   800
```

```
>> Z= zeros (1,800);
>> original=1000.*[se Z];
>> echo=(0.1*500).*[Z se];
>> x=original+echo;
```

Question 6 (Reverb Effects)

The following command was written to play the vector [x].

```
>> wavwrite (x,'nahidmahmud');
Warning: Data clipped during write to file:nahidmahmud
> In wavwrite>PCM_Quantize at 247
  In wavwrite>write_wavedat at 267
  In wavwrite at 112
>> r=wavread ('nahid');
>> sound (r)
```

Question 7 (Spectrogram)

The following command was run in MATLAB:

spectrogram(x,'yaxis');

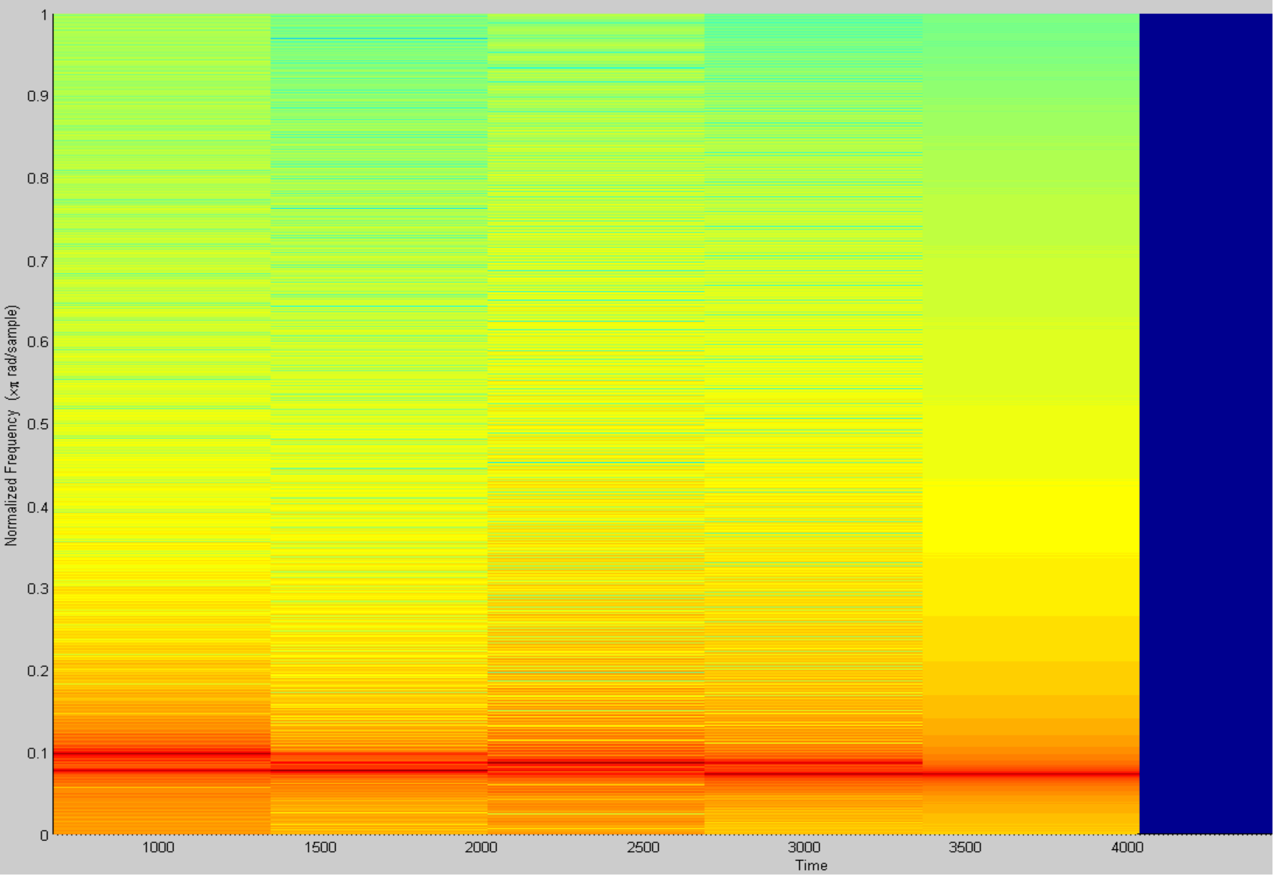

Figure 3: Spectrogram