Digital signal processing homework №1

Homework №1

Shaposhnikov Vladimir

Link to the Jupyter notebook - https://github.com/Volodimirich/DSP (https://github.com/Volodimirich/DSP)

Problem №1

$$a. \quad f(x) = x * e^{-\alpha|x|}, \quad \alpha > 0$$

$$b. \quad f(x) = e^{-a^2x^2} cos(bx)$$

Solution 1a

$$\int_{-\infty}^{\infty} xe^{-\alpha|x|}e^{-\omega xi}dx = \int_{-\infty}^{0} xe^{\alpha x}e^{-\omega xi}dx + \int_{0}^{+\infty} xe^{-\alpha x}e^{-wxi}dx$$

$$x < 0:$$

$$\int_{-\infty}^{0} xe^{(\alpha-\omega i)x}dx = \int_{-\infty}^{0} \frac{x}{\alpha - wi}de^{(\alpha-wi)x} = \frac{1}{\alpha - wi}\left(xe^{(\alpha-wi)x}\Big|_{-\infty}^{0} - \int_{-\infty}^{0} e^{(\alpha-wi)x}dx\right) =$$

$$= \frac{1}{\alpha - wi}\left(xe^{(\alpha-wi)x}\Big|_{-\infty}^{0} - \frac{1}{\alpha - wi}e^{(\alpha-wi)x}\Big|_{-\infty}^{0}\right)$$

Let's analyse both parts:

$$xe^{(\alpha-\omega i)x} \quad \text{in } 0 \Rightarrow 0 \cdot e^0 = 0$$

$$\lim_{x\to-\infty} xe^{(\alpha-\omega i)x} = \lim_{x\to+\infty} \frac{-x}{e^{(\alpha-\omega i)x}} = \langle \text{With using L'Hôpital's rule} \rangle = \lim_{x\to-\infty} \frac{1}{(\alpha - \omega i)e^{(\alpha-\omega i)x}} = 0$$

$$e^{(\alpha-wi)x} \text{ in } 0 = e^0 = 1$$

$$\lim_{x\to-\infty} e^{(\alpha-\omega i)x} = \lim_{x\to+\infty} \frac{1}{e^{(\alpha-\omega i)x}} = 0$$

Then the result is

$$-\frac{1}{\alpha - \omega i} \cdot \frac{0 + (1 - 0)}{\alpha - \omega i} = -\frac{1}{(\alpha - \omega i)^2}$$

$$x > 0:$$

$$\int_{0}^{+\infty} xe^{-(\alpha+\omega i)x}dx = -\int_{0}^{+\infty} \frac{x}{\alpha + \omega i}de^{-(\alpha+\omega i)x} = -\frac{1}{\alpha + wi}\left(xe^{-(\alpha+wi)x}\Big|_{0}^{+\infty} - \int_{0}^{+\infty} e^{-(\alpha+\omega i)x}dx\right) =$$

$$-\frac{1}{\alpha + wi}\left(xe^{-(\alpha+wi)x}\Big|_{0}^{+\infty} + \frac{1}{\alpha + wi}e^{-(\alpha+wi)x}\Big|_{0}^{+\infty}\right)$$

Let's analyse both parts:

$$xe^{-(\alpha+\omega i)x} \quad \text{in } 0 \Rightarrow 0 \cdot e^0 = 0$$

$$\lim_{x\to+\infty} xe^{-(\alpha+\omega i)x} = \lim_{x\to+\infty} \frac{x}{e^{(\alpha+\omega i)x}} = \langle \text{With using L'Hôpital's rule} \rangle = \lim_{x\to+\infty} \frac{1}{(\alpha + \omega i)e^{(\alpha+\omega i)x}} = 0$$

$$e^{-(\alpha+wi)x} \text{ in } 0 = e^0 = 1$$

$$\lim_{x\to+\infty} e^{-(\alpha+\omega i)x} = \lim_{x\to+\infty} \frac{1}{e^{(\alpha+\omega i)x}} = 0$$

Then the result is

$$-\frac{1}{\alpha + \omega i} \cdot \frac{0 + (0 - 1)}{\alpha + \omega i} = \frac{1}{(\alpha + \omega i)^2}$$

Final result is:

$$-\frac{1}{(\alpha - wi)^2} + \frac{1}{(\alpha + wi)^2} = -\frac{\alpha^2 + 2w\alpha i - w^2 - \alpha^2 + 2w\alpha i + w^2}{\left(\alpha^2 + w^2\right)^2} = -\frac{4\alpha wi}{\left(\alpha^2 + w^2\right)^2}$$

Solution 1b

$$e^{-a^2x^2}\cos(bx) = e^{-a^2x^2}\left(\frac{e^{bx}+e^{-bx}}{2}\right) = \frac{e^{-a^2x^2}}{2}e^{bxi} + \frac{e^{-a^2x^2}}{2}e^{-bxi}$$
$$Lets\ use\ fourier\ shift\ property:$$
$$x(t)\cdot e^{w_o it} = X((w-w_o)i)$$

Let's calculate:

$$\frac{1}{2}\int_{-\infty}^{\infty} e^{-a^2x^2-wix+z-z}dx$$

$$-a^2x^2 - i\omega x - z \Leftrightarrow -(\alpha x+\beta)^2$$
$$-a^2x^2 - wix - z \Leftrightarrow -\alpha^2x^2 - 2\alpha\beta x - \beta^2$$
$$\begin{cases} a^2 = \alpha^2 \\ wi = 2\alpha\beta \\ z = \beta^2 \end{cases} \Rightarrow \begin{cases} \alpha = |a| \\ \beta = \frac{wi}{2|a|} \\ z = \frac{-w^2}{4a^2} \end{cases}$$
$$\frac{1}{2}\int_{-\infty}^{+\infty} e^{-a^2x^2-wix}\cdot e^{\frac{w^2}{4a^2}-\frac{w^2}{4a^2}}dx =$$
$$= \frac{1}{2|a|}e^{\frac{-w^2}{4a^2}}\int_{-\infty}^{+\infty} e^{-\left(|a|x+\frac{\omega i}{2|a|}\right)^2} d\left(|a|x+\frac{\omega i}{2|a|}\right) = \frac{\sqrt{\pi}}{2|a|}\cdot e^{-\frac{w^2}{4a^2}}$$

With using euler integral euler-poisson integral:

$$\int_{-\infty}^{+\infty} e^{-x^2}dx = \sqrt{\pi}$$

With using Fourier Shift property:

$$X(wi) = \frac{\sqrt{\pi}}{2|a|}\cdot e^{-\frac{w^2}{4a^2}}$$
$$\frac{1}{2}e^{-a^2x^2}\cdot e^{bxi} + \frac{1}{2}e^{-a^2x^2}\cdot e^{-bxi} = X((w-b)i) + X((w+b)i) =$$
$$\frac{\sqrt{\pi}}{2|a|}\cdot e^{-\frac{(w-b)^2}{4a^2}} + \frac{\sqrt{\pi}}{2|a|}\cdot e^{-\frac{(w+b)^2}{4a^2}}$$
$$.$$

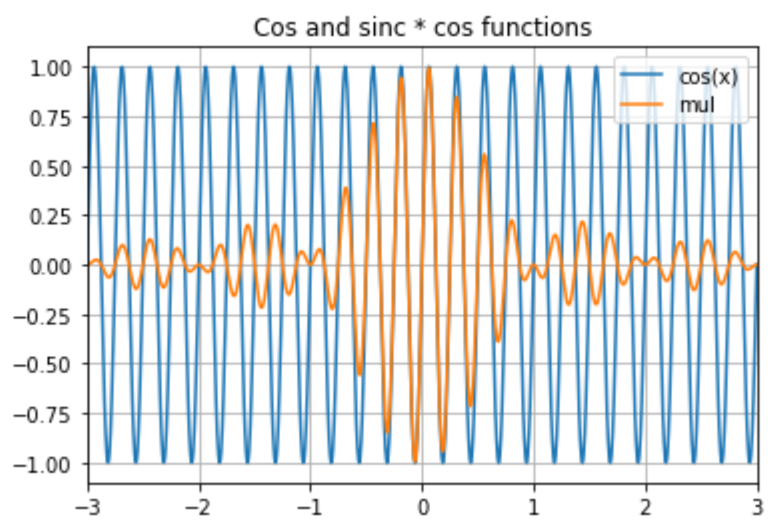
Problem №2

Compute the spectrum of the cosine-filled sinc function:

In [2]:
```
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline

```

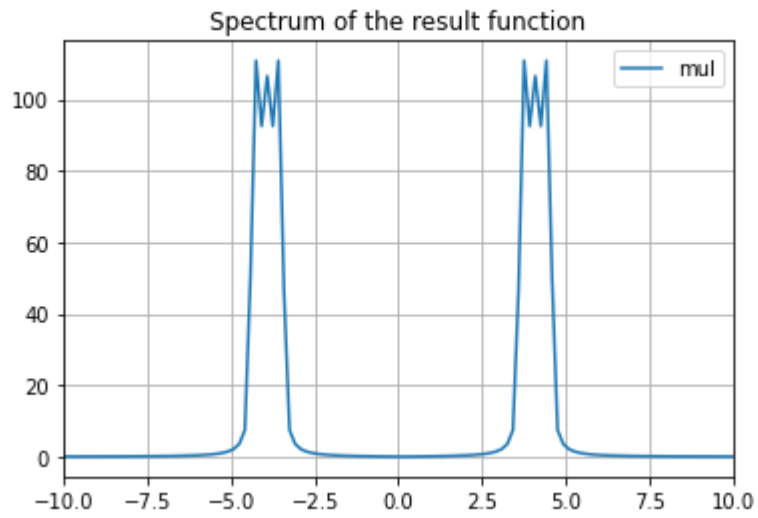
In [3]:
```
sample_freq = 200
interval = (-3, 3)

x = np.linspace(interval[0], interval[1], (interval[1] - interval[0]) * sample_freq)
y1 = np.sinc(x)
y2 = np.cos(8*np.pi*x - np.pi/2) #According picture in the homework

fig, ax = plt.subplots()
ax.plot(x, y2, label = 'cos(x)')
ax.plot(x, y1*y2, label = 'mul')
ax.set_xlim([-3, 3])
ax.set_title('Cos and sinc * cos functions')
ax.grid()
plt.legend(loc = 'upper right')



```

Out[3]: <matplotlib.legend.Legend at 0x7f2347fcfd30>



In [4]:
```
y_f_mul = np.fft.fftshift(np.fft.fft(y1*y2))
y_f_cos = np.fft.fftshift(np.fft.fft(y2))

x_f = np.linspace(-sample_freq/2, sample_freq/2, len(x))

fig, bx = plt.subplots()
bx.plot(x_f, abs(y_f_mul), label = 'mul')
bx.set_xlim([-10, 10])
bx.set_title('Spectrum of the result function')

bx.grid()
bx.legend()


```

Out[4]: <matplotlib.legend.Legend at 0x7f2344692f70>

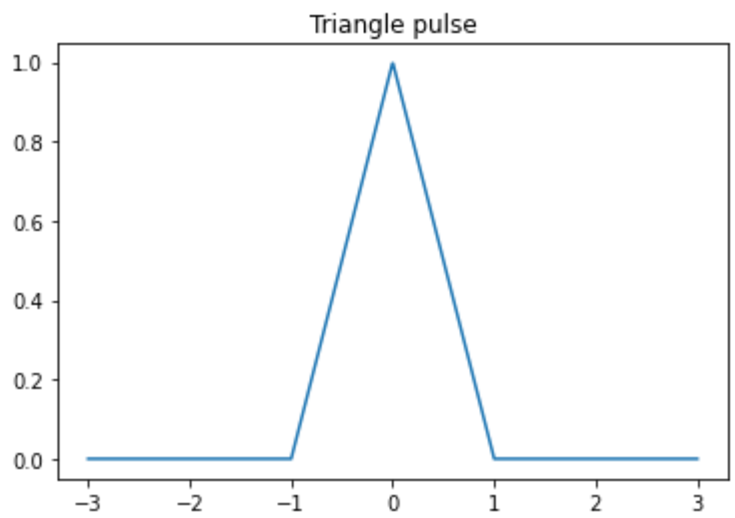


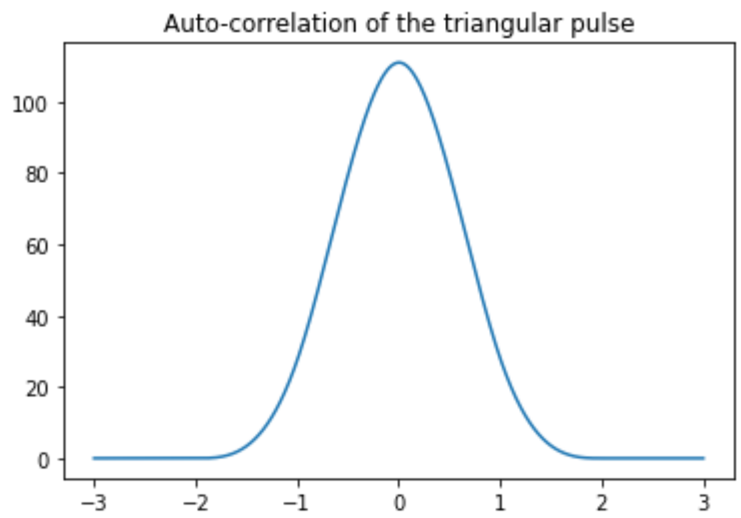
Problem №3

Compute the auto-correlation of the triangular pulse

In [4]:

```python
def triangle_wave(x = 0, c = 1, hc = 1, k = 1): #  with amplitude hc, width c, and slope hc/2c
    if abs(x) >= c/k:
        return 0
    else:
        sign = 1 if x < 0 else -1
        return  hc + (k * sign * x * hc )/ c

x=np.linspace(-3,3,1000)
y=np.array([triangle_wave(t) for t in x])
plt.plot(x, y)
plt.title('Triangle pulse')
plt.show()

act = np.correlate(y, y, mode='same')

plt.plot(x, act)
plt.title('Auto-correlation of the triangular pulse')
plt.show()


```





Problem №4

Compute (by hands) the convolution of the following signals:

$$\begin{array}{ll} a. & h = [2, 3, 6, 8], \quad x = [1, 2, 10, 1] \\ b. & h = [5, 1, 3, 10], \quad x = [9, 6, 10, 1] \end{array}$$

Solution 4a

h = [2,3,6,8], x = [1,2,10,1]
h_rev = [8,6,3,2], $x = [1, 2, 10, 1]$

```
             8   6   3   2
                         |               => 1 * 2 = 2
                         1   2   10   1


             8   6   3   2
                     |   |               => 1 * 3 + 2 * 2 = 7
                     1   2   10   1


             8   6   3   2
                 |   |   |               => 6 * 1 + 3 * 2 + 2 * 10 = 32
                 1   2   10   1


             8   6   3   2
             |   |   |   |               => 8 * 1 + 6 * 2 + 3 * 10 + 2 * 1 = 52
```

```
             1   2   10  1


             8   6   3   2
             |   |   |                    => 8 * 2 + 6 * 10 + 3 * 1 = 79
         1   2   10  1


             8   6   3   2
             |   |                        => 8 * 10 + 6 * 1 = 86
     1   2   10  1


             8   6   3   2
             |                         => 8 * 1 = 8
1   2   10   1

res_conv = [2 7 32 52 79 86 8]

```

In [5]:
```python
np.convolve([2,3,6,8], [1,2,10,1])
```

Out[5]: `array([ 2,  7, 32, 52, 79, 86,  8])`

Solution 4b

h = [5,1,3,10], x = [9,6,10,1]
h_rev = [10,3,1,5], x = [9,6,10,1]

```
             10  3   1   5
                         |                => 5 * 9 = 45
                         9   6   10   1

             10  3   1   5
                     |   |                => 1 * 9 + 5 * 6 = 39
                     9   6   10   1

             10  3   1   5
                 |   |   |                => 3 * 9 + 1 * 6 + 5 * 10 = 83
                 9   6   10   1

             10  3   1   5
             |   |   |   |                => 10 * 9 + 3 * 6 + 1 * 10 + 5 * 1 = 123
             9   6   10  1

             10  3   1   5
             |   |   |                    => 10 * 6 + 3 * 10 + 1 * 1 = 91
         9   6   10  1

             10  3   1   5
             |   |                         => 10 * 10 + 3 * 1 = 103
     9   6   10  1

             10  3   1   5
             |                            => 10 * 1 = 10
9   6   10   1

res_conv = [45 39 83 123 91 103 10]

```

In [4]:
```python
np.convolve([5,1,3,10], [9,6,10,1])
```

Out[4]: `array([ 45,  39,  83, 123,  91, 103,  10])`

Problem №5

The sinusoidal signal with the frequency 6kHz is sampled with frequency 10 kHz. Compute the apparent frequency after the signal reconstruction

Solution 5a

```
In this case we have sample frequence 10kHz. According to Kotelnikov theorem we can reconstruct without losing only if the sampling rate > the double highest signal frequency. In our case this equality is not satisfied.
But we can use Dirac comb property:

```

$$\int_{-\infty}^{+\infty} x(t) \cdot \delta(t - nt_0)dt = X(nt_0)$$

```
With using this property we can add multiplie copies of the original spectrum. With using shifted copy of the spectrum we can compute frequency after the sighal reconstruction - 4kHz
```

title

Solution 5b

```
In this problem, Kotelnikov's theorem is fulfilled. We will be able to restore the sampled signal. As you can see on the graph, the recovery is not perfect, but the frequency and amplitude are close to the original.
```

In [11]:

```python
interval = [0, 8]
sample_freq = 6000



x = np.linspace(interval[0], interval[1], (interval[1] - interval[0]) * sample_freq)
sin_orig = np.sin(2*np.pi*x)
downsampling_rate = len(x)//(8 * 2.5)


idx = np.arange(0, len(x), downsampling_rate + 100).astype(int)
z_downsampled = np.zeros_like(sin_orig)
z_downsampled[idx] = sin_orig[idx]


# Plot original and downsampled versions of our signal
plt.figure()
plt.plot(x, sin_orig)  # From the previous cell
plt.plot(x, z_downsampled) # All except z_downsampled[idx] are zeros

# Now, restore the signal using sinc functions
def sinc(x):
    return np.sinc(x / downsampling_rate)


z_restored = np.zeros_like(z_downsampled)
z_samples = z_downsampled[idx]
for i in range(len(x)):
    z_restored[i] += np.sum(z_samples * sinc(idx - i))

plt.plot(x, z_restored)
plt.xlim([0, 4]) # Zoom the X-axis in
```

Out[11]: (0.0, 4.0)