In [1]:
```
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
```

Task 1.a

Compute the Fourier transform for the function: $f(x) = xe^{-\alpha|x|}, \alpha > 0$

$$F(\omega) = \int_{-\infty}^{+\infty} f(x)e^{-j\omega x}dx = \int_{-\infty}^{0} xe^{\alpha x}e^{-j\omega x}dx + \int_{0}^{+\infty} xe^{-\alpha x}e^{-j\omega x}dx = I_1 + I_2$$

$$\begin{aligned} I_1 &= \int_{-\infty}^{0} xe^{(\alpha-j\omega)x}dx = \frac{1}{\alpha - j\omega}\int_{-\infty}^{0} xde^{(\alpha-j\omega)x} = \\ &= \frac{1}{\alpha-j\omega}\left[xe^{(\alpha-j\omega)x}|_{-\infty}^{0} - \frac{1}{\alpha-j\omega}\int_{-\infty}^{0} e^{(\alpha-j\omega)x}d((\alpha-j\omega)x)\right] = \\ &= \frac{1}{\alpha-j\omega}\left[0 - \frac{1}{\alpha-j\omega}xe^{(\alpha-j\omega)x}|_{-\infty}^{0}\right] = \\ &= \frac{1}{\alpha-j\omega}\left[-\frac{1}{\alpha-j\omega}(1-0)\right] = \frac{-1}{(\alpha-j\omega)^2} \end{aligned}$$

Note:

$$\begin{aligned} xe^{(\alpha-j\omega)x}\Big|_{-\infty}^{0} &= 0 - \lim_{x\to-\infty} xe^{(\alpha-j\omega)x} = \\ = -\lim_{x\to-\infty}\frac{x}{e^{-(\alpha-j\omega)x}} &= [\frac{\infty}{\infty}] = by\ L'Hopital's\ rule = \\ &= -\lim_{x\to-\infty}\frac{1}{-(\alpha-j\omega)e^{-(\alpha-j\omega)x}} = 0 \end{aligned}$$

$$\begin{aligned} I_2 &= \int_{0}^{+\infty} xe^{-(\alpha+j\omega)x}dx = -\frac{1}{\alpha+j\omega}\int_{0}^{+\infty} xde^{-(\alpha+j\omega)x} = \\ &= -\frac{1}{\alpha+j\omega}\left[xe^{-(\alpha+j\omega)}|_{0}^{+\infty} + \frac{1}{\alpha+j\omega}\int_{0}^{+\infty} e^{-(\alpha+j\omega)x}d(-(\alpha+j\omega)x)\right] = \\ &= -\frac{1}{(\alpha+j\omega)^2}(0-1) = \frac{1}{(\alpha+j\omega)^2} \end{aligned}$$

Note:

$$\begin{aligned} xe^{-(\alpha+j\omega)x}\Big|_{0}^{+\infty} &= \lim_{x\to+\infty} xe^{-(\alpha+j\omega)x} - 0 = \\ = \lim_{x\to+\infty}\frac{x}{e^{(\alpha+j\omega)x}} &= [\frac{\infty}{\infty}] = by\ L'Hopital's\ rule = \\ &= \lim_{x\to+\infty}\frac{1}{(\alpha+j\omega)e^{(\alpha+j\omega)x}} = 0 \end{aligned}$$

Finally:

$$I = I_2 + I_1 = \frac{1}{(\alpha + j\omega)^2} + \frac{-1}{(\alpha - j\omega)^2} =$$
$$= \frac{\alpha^2 - 2j\omega\alpha - \omega^2 - \alpha^2 - 2j\omega + \omega^2}{(\alpha + j\omega)(\alpha + j\omega)(\alpha - j\omega)(\alpha - j\omega)} = \frac{-4j\omega\alpha}{(\alpha^2 + \omega^2)^2}$$

Answer:

$$F(\omega) = \frac{-4j\omega\alpha}{(\alpha^2 + \omega^2)^2}$$

Task 1.b

Compute the Fourier transform for the function: $f(x) = e^{-a^2x^2}cos(bx)$

Here I will use the shifting property of FT. For this reason I will start my calculations from computing the FT of function: $f(x) = e^{-a^2x^2}$ and then will apply shift to the result.

$$F_1(\omega) = \int_{-\infty}^{+\infty} e^{-a^2x^2} e^{-j\omega x} dx =$$
$$= \int_{-\infty}^{+\infty} e^{-(a^2x^2 + j\omega x - \frac{\omega^2}{4a^2} + \frac{\omega^2}{4a^2})} dx =$$
$$= e^{-\frac{\omega^2}{4a^2}} \int_{-\infty}^{+\infty} e^{-\left(|a|x + \frac{j\omega}{2|a|}\right)^2} dx =$$
$$= e^{-\frac{\omega^2}{4a^2}} \frac{1}{|a|} \int_{-\infty}^{+\infty} e^{-\left(|a|x + \frac{j\omega}{2|a|}\right)^2} d\left(|a|x + \frac{j\omega}{2|a|}\right) = \frac{\sqrt{\pi}}{|a|} e^{-\frac{\omega^2}{4a^2}}$$

Applying shifting property:

$$F(\omega) = \int_{-\infty}^{+\infty} e^{-\alpha^2x^2} \cos(bx) e^{-j\omega x} dx = \frac{1}{2} \int_{-\infty}^{+\infty} e^{-a^2x^2} \left(e^{jbx} + e^{-jbx}\right) e^{-j\omega x} dx =$$
$$= \frac{1}{2} \left(\int_{-\infty}^{+\infty} e^{-a^2x^2} e^{-j(\omega - b)x} dx + \int_{-\infty}^{+\infty} e^{-a^2x^2} e^{-j(\omega + b)x} dx \right) =$$
$$= \frac{1}{2} \left(\frac{\sqrt{\pi}}{|a|} e^{-\frac{(\omega - b)^2}{4a^2}} + \frac{\sqrt{\pi}}{|a|} e^{-\frac{(\omega + b)^2}{4a^2}} \right) = \frac{\sqrt{\pi}}{2|a|} \left(e^{-\frac{(\omega - b)^2}{4a^2}} + e^{-\frac{(\omega + b)^2}{4a^2}} \right)$$

Answer:

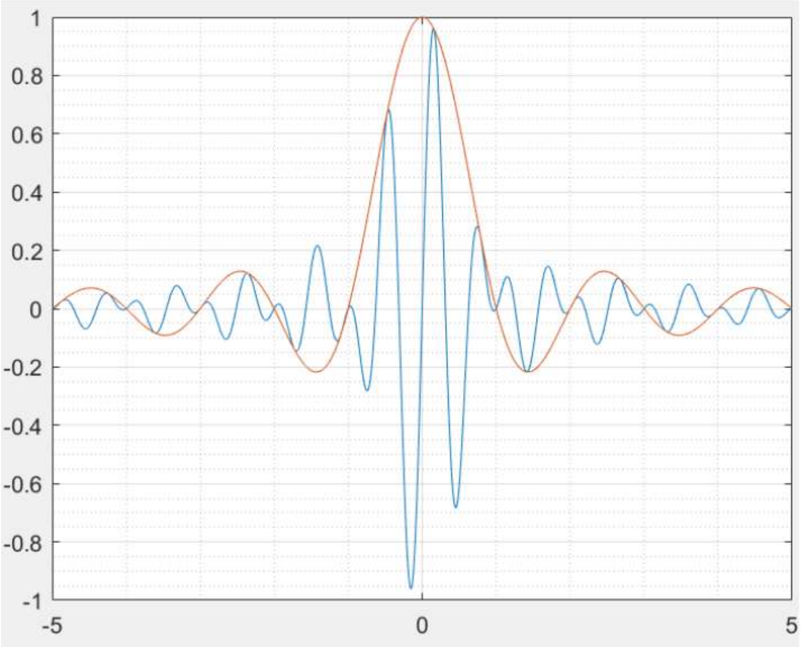
$$F(\omega) = \frac{\sqrt{\pi}}{2|a|} \left(e^{-\frac{(\omega - b)^2}{4a^2}} + e^{-\frac{(\omega + b)^2}{4a^2}} \right)$$

Task 2

Compute the spectrum of the cosine-filled sinc function:

In [149…

```
#plot picture from HW file
from IPython.display import Image
display(Image(filename='sinc.png', width = 400))
```



First of all, I defined the cosine-filled sinc function: $f(x) = \frac{\sin(\pi x)}{\pi x}\cos(\omega_0 x + \varphi_0)$. I write ω_0, φ_0 because I want to derive the representation of cosine-filled sinc function in the frequency domain in general case.
By the way, it is obvious that for our case $\varphi_0 = -\frac{\pi}{2}$ (because cosine is delayed ($f(0) = 0$)) and $\omega_0 = 3\pi$
Proof: $f(x) = \frac{\sin(\pi x)}{\pi x}\cos(\omega_0 x - \frac{\pi}{2}) = 0$,
then

$$\left[\begin{array}{c} sinc(x) = 0 \\ cos(\omega_0 x - \dfrac{\pi}{2}) = 0 \end{array}\right.$$

from the second equation we get

$$\omega_0 x - \frac{\pi}{2} = \frac{\pi}{2} + \pi k, k \in N$$
$$\omega_0 x = \pi + \pi k$$

Then according to the graph above I can say that when $x = \frac{1}{3}$ cosine-filled function equals to zero, so the first root (when $k = 0$) of the equation above can be find in this way:

$$\frac{\omega_0}{3} = \pi$$
$$\omega_0 = 3\pi$$

$$F(\omega) = \int_{-\infty}^{+\infty} sinc(x)\cos(\omega_0 x + \varphi_0) e^{-j\omega x} dx =$$
$$= \int_{-\infty}^{+\infty} sinc(x)\left(\frac{e^{j(\omega_0 x + \varphi_0)} + e^{-j(\omega_0 x + \varphi_0)}}{2}\right) \cdot e^{-j\omega x} dx =$$
$$= \frac{1}{2}\left(\int_{-\infty}^{+\infty} e^{j\varphi_0} sinc(x) e^{-j(\omega - \omega_0)x} dx + \int_{-\infty}^{+\infty} e^{-j\varphi_0} sinc(x) e^{-j(\omega + \omega_0)x} dx\right) = I_1 + I_2$$

$$I_1 = \frac{1}{2}e^{j\varphi_0}\int_{-\infty}^{+\infty} sinc(x)e^{-j(\omega-\omega_0)x}dx = \begin{cases} \frac{1}{2}e^{j\varphi_0}, |\omega-\omega_0| \leqslant \pi \\ 0, \text{otherwise} \end{cases}$$

$$I_2 = \frac{1}{2}e^{-j\varphi_0}\int_{-\infty}^{+\infty} sinc(x)e^{-j(\omega+\omega_0)x}dx = \begin{cases} \frac{1}{2}e^{-j\varphi_0}, |\omega+\omega_0| \leqslant \pi \\ 0, \text{otherwise} \end{cases}$$

For our case $\omega_0 = 3\pi, \varphi_0 = -\frac{\pi}{2}$ we get:

$$I_1 = \begin{cases} \frac{j}{2}, 2\pi \leqslant \omega \leqslant 4\pi \\ 0, \text{otherwise} \end{cases}$$

$$I_2 = \begin{cases} -\frac{j}{2}, -4\pi \leqslant \omega \leqslant -2\pi \\ 0, \text{otherwise} \end{cases}$$

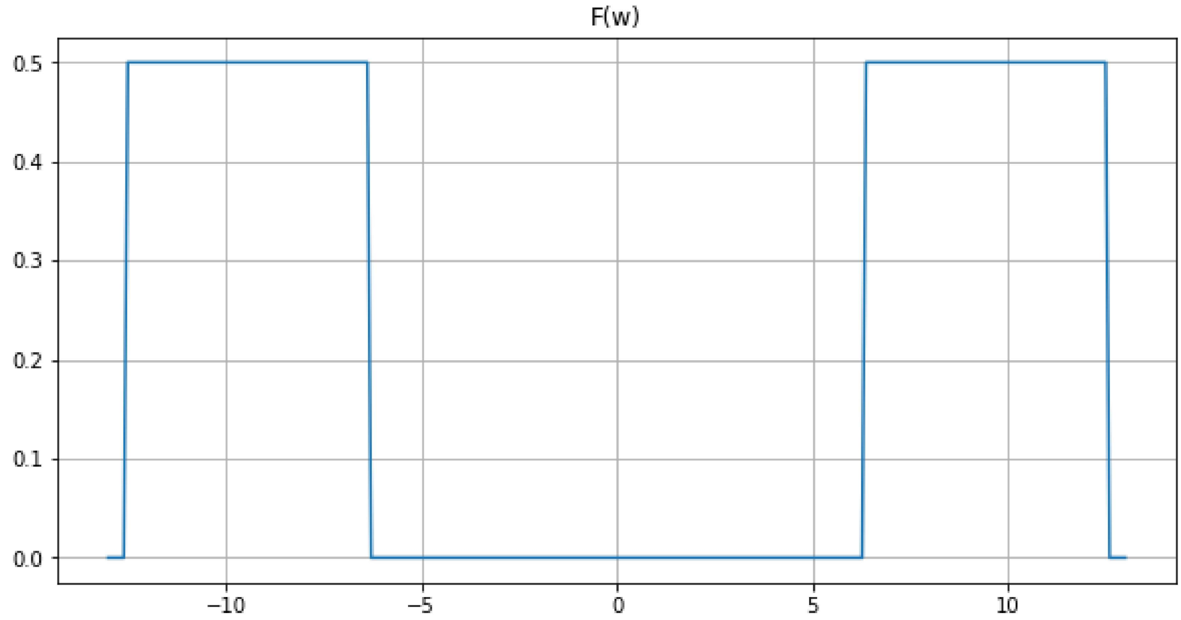
Finally:

$$I = \begin{cases} -\frac{j}{2}, -4\pi \leqslant \omega \leqslant -2\pi \\ \frac{j}{2}, 2\pi \leqslant \omega \leqslant 4\pi \\ 0, \text{otherwise} \end{cases}$$

Now let's plot the graph. Firstly, we need to find the magnitude of the signal in frequency domain. In our case it is obvious that $A = \frac{1}{2}$ (by definition $A(\omega) = \sqrt{F(\omega)F^*(\omega)}$)

In [3]:
```
x = np.linspace(-13, 13, 260)
y = []
for point in x:
    if (-4 * np.pi <= point <= -2 * np.pi):
        y.append(0.5)
    elif (2 * np.pi <= point <= 4 * np.pi): # of course, I can fill y
                                            # list in one if clause
        y.append(0.5)                       #but I did it in this
                                            #way for the better understanding
    else:
        y.append(0)
```

In [4]:
```
fig, ax = plt.subplots(1, 1, figsize=(10,5))
ax.plot(x, y)
ax.grid()
ax.set_title('F(w)');
```



We recieved the rectangular window in frequency domain shifted by $\omega_0 = 3\pi$ with two times decreased magnitude.

Answer:

$$I = \begin{cases} -\frac{j}{2}, -4\pi \leqslant \omega \leqslant -2\pi \\ \frac{j}{2}, 2\pi \leqslant \omega \leqslant 4\pi \\ 0, \text{otherwise} \end{cases}$$

Task 3

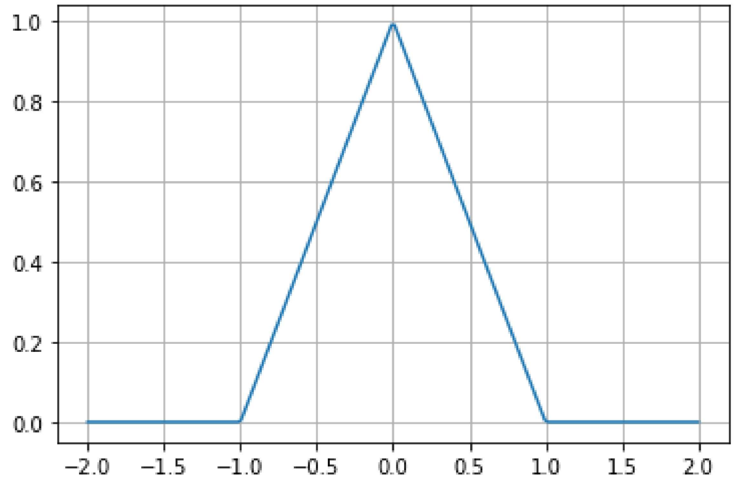
Compute the auto-correlation of the triangle pulse:

In [5]:

```
x = np.linspace(-2, 2, 200)
y = []
for point in x:
    if (-1 <= point <= 0):
        y.append(point + 1)
    elif (0 < point <= 1):
        y.append(-point + 1)
    else:
        y.append(0)
```

In [6]:

```
plt.plot(x, y) # just check that everything is ok
plt.grid()
```



In [7]:

```
autocorr = np.correlate(y, y, 'full') # calculate an autocorrelation of the given fu
```

In [8]:

```
print(f'length x {len(x)}')
print(f'length autocorr {len(autocorr)}')
```

```
length x 200
length autocorr 399
```

In [9]:

```
ac = autocorr[100:300] # cut zeros from both sides
#(because the lentgth is 2 * len(x) - 1 and I want len(autocorr) = len(x))
```

In [10]:

```
len(ac)
```

Out[10]:

```
200
```

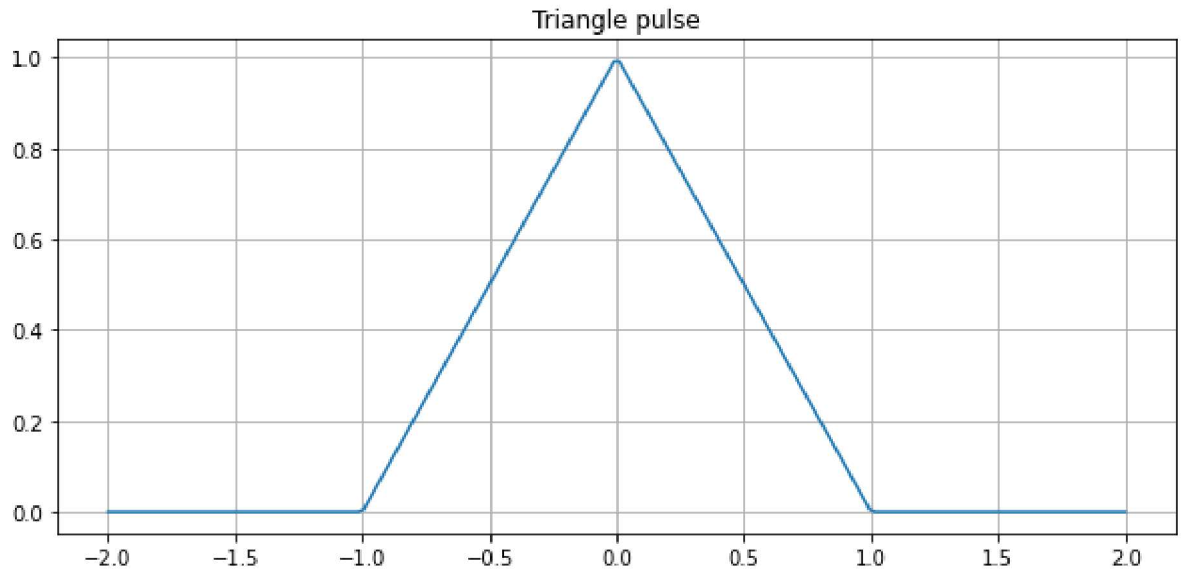
In [11]:

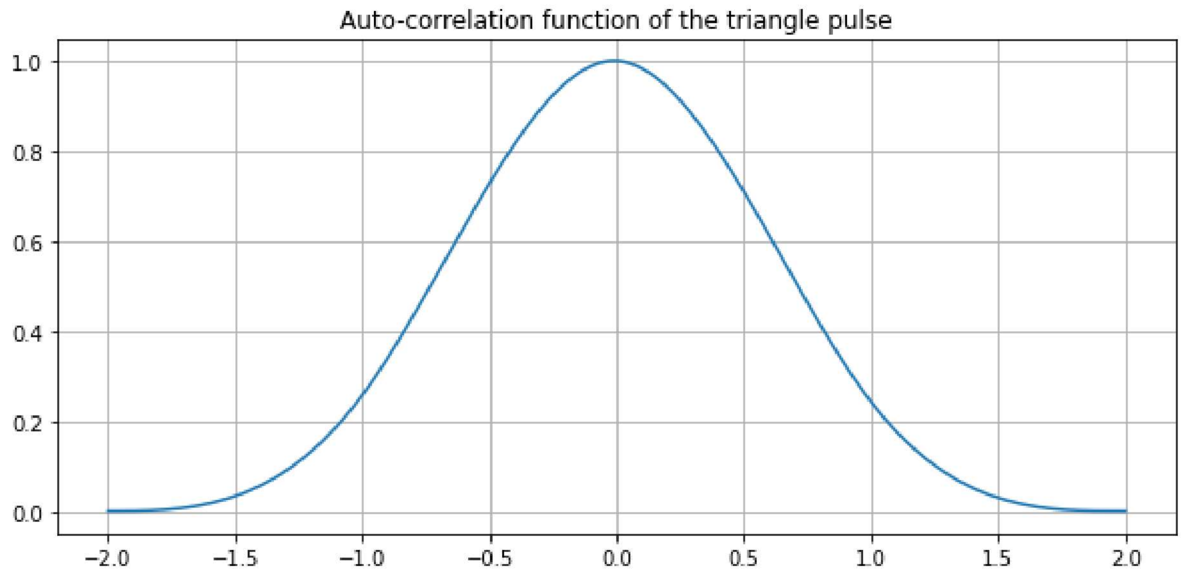
```
ac_norm = ac / max(ac) # normlize the autocorrelation
```

Answer

In [12]:

```
fig, ax = plt.subplots(2, 1, figsize=(10,10))
ax[0].plot(x, y)
ax[0].grid(True)
ax[0].set_title('Triangle pulse')
ax[1].plot(x, ac_norm)
ax[1].grid(True)
ax[1].set_title('Auto-correlation function of the triangle pulse');
```





In [ ]:

In [ ]:

Task 4.a

Compute (by hands) the convolution of the following signals:
h = [2, 3, 6, 8], x =[1, 2, 10, 1]

By definition of the discrete convolution:

$$(f * g)[n] = \sum_{m=-M}^{M} f[n-m]g[m]$$

Firstly I flip h signal:

$$h_{flipped} = [8, 6, 3, 2]$$

Then calculate the convolution:

1) Step 1:

$$\begin{array}{c} 8\;6\;3\;2 \\ \quad\quad\quad 1\;2\;10\;1 \\ 2\cdot 1 = 2 \end{array}$$

2) Step 2:

$$\begin{array}{c} 8\;6\;3\;2 \\ \quad\quad 1\;2\;10\;1 \\ 3\cdot 1 + 2\cdot 2 = 3 + 4 = 7 \end{array}$$

3) Step 3:

$$\begin{array}{c} 8\;6\;3\;\;2 \\ \quad 1\;2\;10\;1 \\ 6\cdot 1 + 3\cdot 2 + 2\cdot 10 = 6 + 6 + 20 = 32 \end{array}$$

4) Step 4:

$$\begin{array}{c} 8\;6\;\;3\;\;2 \\ 1\;2\;10\;1 \\ 8\cdot 1 + 6\cdot 2 + 3\cdot 10 + 2\cdot 1 = 8 + 12 + 30 + 2 = 52 \end{array}$$

5) Step 5:

$$\begin{array}{c} \quad 8\;\;6\;\;3\;\;2 \\ 1\;2\;10\;1 \\ 8\cdot 2 + 6\cdot 10 + 3\cdot 1 = 16 + 60 + 3 = 79 \end{array}$$

6) Step 6:

$$\begin{array}{c} \quad\quad 8\;\;6\;\;3\;\;2 \\ 1\;2\;10\;1 \\ 8\cdot 10 + 6\cdot 11 + 3\cdot 1 = 80 + 6 = 86 \end{array}$$

7) Step 7:

$$\begin{array}{c} \quad\quad\quad 8\;\;6\;\;3\;\;2 \\ 1\;2\;10\;1 \\ 8\cdot 1 = 8 \end{array}$$

Answer: $conv = [2, 7, 32, 52, 79, 86, 8]$

Task 4.b

Compute (by hands) the convolution of the following signals:
h = [5, 1, 3, 10], x =[9, 6, 10, 1]

By definition of the discrete convolution:

$$(f * g)[n] = \sum_{m=-M}^{M} f[n-m]g[m]$$

Firstly I flip h signal:

$$h_{flipped} = [10, 3, 1, 5]$$

Then calculate the convolution:

1) Step 1:

$$10\ 3\ 1\ 5$$
$$9\ 6\ 10\ 1$$
$$5 \cdot 9 = 45$$

2) Step 2:

$$10\ 3\ 1\ 5$$
$$9\ 6\ 10\ 1$$
$$5 \cdot 6 + 1 \cdot 9 = 39$$

3) Step 3:

$$10\ 3\ 1\ 5$$
$$9\ 6\ 10\ 1$$
$$3 \cdot 9 + 1 \cdot 6 + 5 \cdot 10 = 27 + 6 + 50 = 83$$

4) Step 4:

$$10\ 3\ 1\ 5$$
$$9\ 6\ 10\ 1$$
$$10 \cdot 9 + 3 \cdot 6 + 1 \cdot 10 + 5 \cdot 1 = 90 + 18 + 10 + 5 = 123$$

5) Step 5:

$$10\ 3\ 1\ 5$$
$$9\ 6\ 10\ 1$$
$$10 \cdot 6 + 3 \cdot 10 + 1 \cdot 1 = 60 + 30 + 1 = 91$$

6) Step 6:

$$10\ 3\ 1\ 5$$
$$9\ 6\ 10\ 1$$
$$10 \cdot 10 + 3 \cdot 1 = 100 + 3 = 103$$

7) Step 7:

$$10\ 3\ 1\ 5$$
$$9\ 6\ 10\ 1$$
$$10 \cdot 1 = 10$$

Answer: $conv = [45, 39, 83, 123, 91, 103, 10]$

Task 5.a

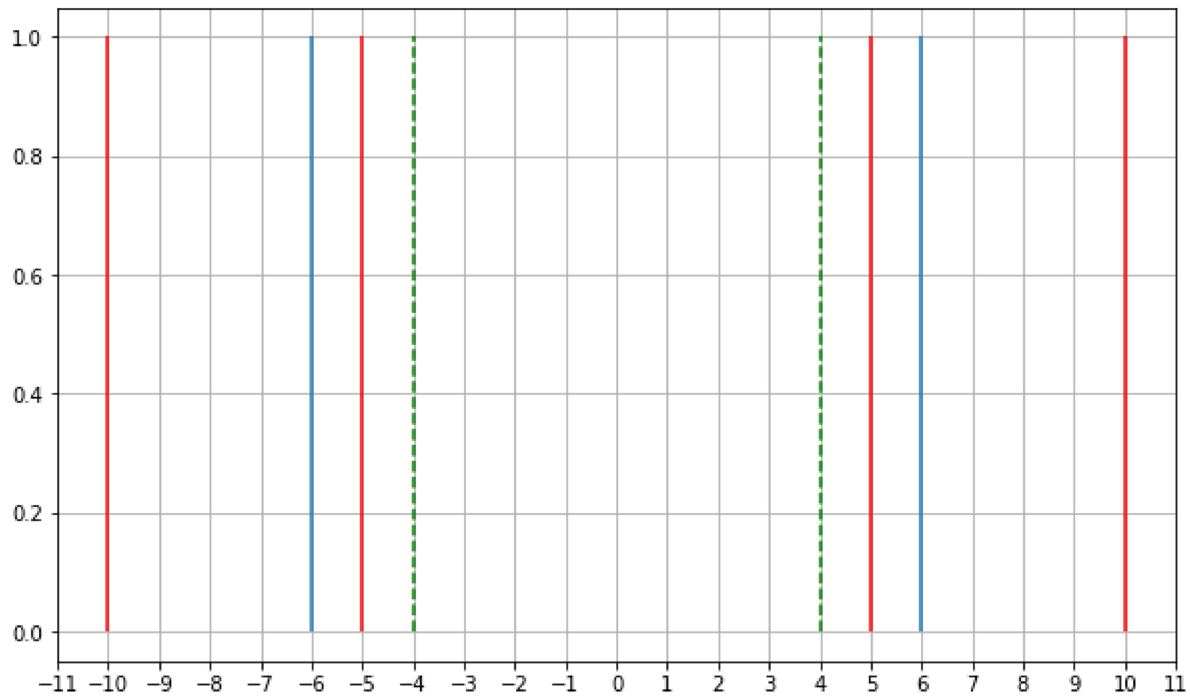
The sinusoidal signal with the frequency $f = 6kHz$ is sampled with frequency $f_s = 10kHz$. Compute the apparent frequency after the signal reconstruction.

For simplicity assume that the magnitude of signal is 1, the phase is 0. Then $f(x) = sin(2 \cdot 6 \cdot 10^3 \pi x)$

The FT of this function will give two delta functions in frequency domain located on $f = \pm 6kHx$. Let's plot it.

In [28]:
```
x = list(range(-12, 12 ,1))
y = []
for point in x:
    if point == -6:
        y.append(1)
    elif point == 6:
        y.append(1)
    else:
        y.append(0)
```

In [46]:
```
# red corresponds to the sampling frequency
# blue corresponds to the initial signal frequency
# gren dash corresponds to the aliasing frequency
fig, ax = plt.subplots(figsize=(10,6))
major_ticks = list(range(-16, 16, 1))
ax.vlines(6, 0, 1)
ax.vlines(-6, 0, 1)
ax.grid()
ax.set_xticks(major_ticks);
# Then I plot our sampling frequence
ax.vlines(5, 0, 1, color = 'r')
ax.vlines(-5, 0, 1, color = 'r')
# It is obvious that frequency of the initial signal lies
#outside the fungamental frequency range (|F|<= Fs/2) so we will achieve
#aliasing here.
# Then I will plot the copies of initial signal near the multiplies of Fs
ax.vlines(10, 0, 1, color = 'r')
ax.vlines(-10, 0, 1, color = 'r')
ax.vlines(4, 0, 1, color = 'g', linestyle = '--')
ax.vlines(-4, 0, 1, color = 'g',linestyle = '--');
# As you can see I obtain frequency = +-4 kHz inside the fundamental frequency range
# It is the aliasing frequency
```



Answer: after the signal reconstruction we will get the sinusoidal signal with the frequency $f = 4kHz$

Task 5.b

The sinusoidal signal with the frequency $f = 6kHz$ is represented by 4 signals periods sampled with the sample clock frequency $f_s = 15kHz$. Are these samples enough to reconstruct the initial signal correctly.

I think that we can not reconstruct signal correctly based on these samples. Because when we sample function at frequence f_s the sinusoidal functions yelds the same set of samples $\{sin(2\pi(f + Nf_s)x + \varphi, N = 0, \pm1, \pm2, \ldots\}$ So without additional information we can't distinguish the frequency of the original function. I will try to give an example.

In [142…

```
# define the frequency of the initial signal and sampling freauency
f = 6
f_s = 15
T = 1 / f
T_s = 1 / f_s
```

In [143…

```
x = np.arange(0, 4 * T, T_s)
```

In [144…

```
y = np.sin(2 * np.pi * f * x) # initial signal
```

In [145…

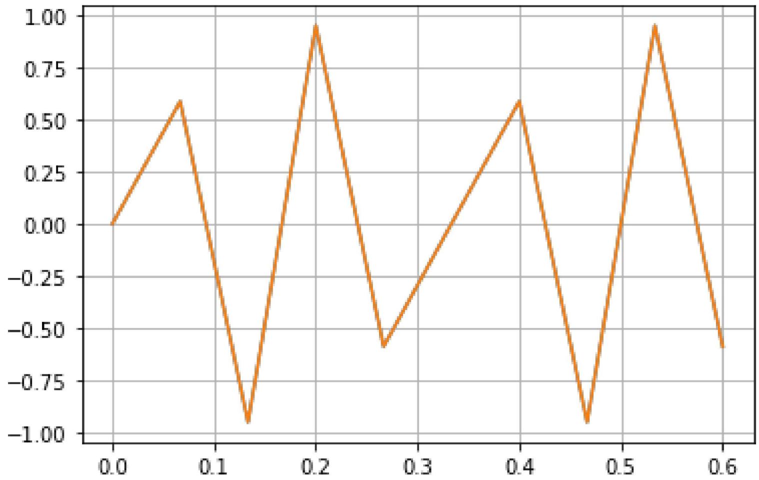
```
N = 1
```

In [147…

```
y_2 = np.sin(2 * np.pi * (f + N * f_s)* x) # aliased sinusoidal signal
```

In [148…

```
plt.plot(x, y)
plt.plot(x, y_2);
plt.grid()
```



According to the graph above, both sinusoudal functions with different carrier frequencies has the same set of samples, that's why their graphs coincides.
Answer: we can't reconstruct the initial signal correctly without additional information about this signal.

You can download this notebook and run locally:
https://github.com/dmasny99/DSP_skoltech/blob/main/HW1_Masnyi_Dmitrii.ipynb