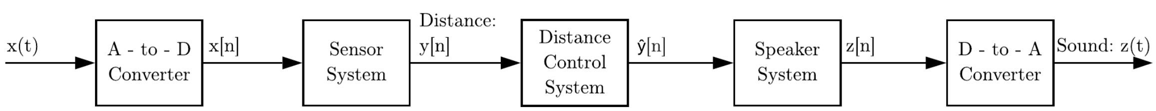
Assignment 1

Batuhan Faik Derinbay derinbay18@itu.edu.tr
150180705

Chapter 1: Introduction

Question 1

(a) Block Diagram of Parking Sensor System



(b) **1D Signal:** Music
2D Signal: Static Image
3D Signal: Video
4D Signal: Magnetic Resonance Images

Chapter 2: Sinusoids

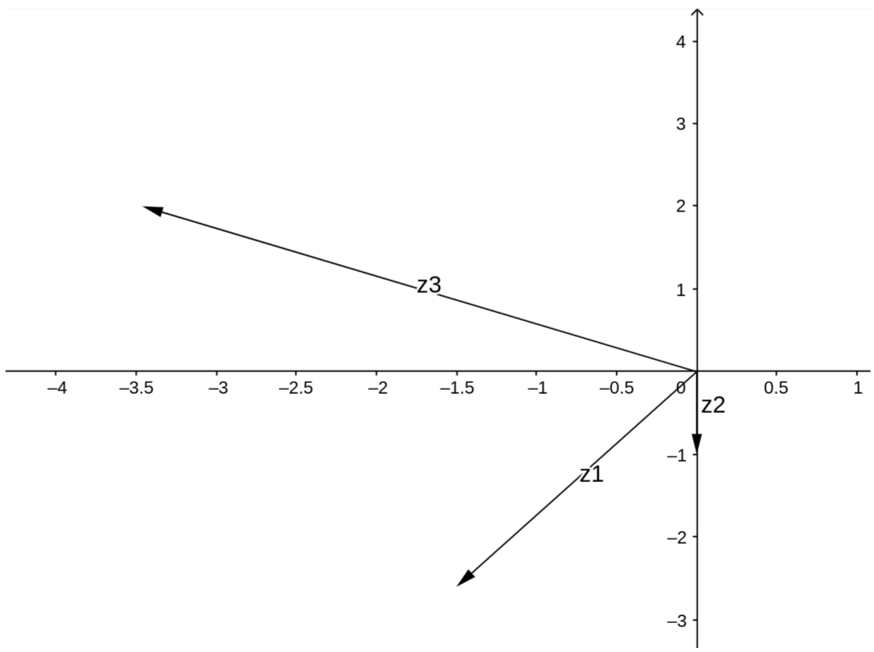
Question 2

(a) Expression of $x(t)$ in the form of $x(t) = A\cos(wt + \phi)$

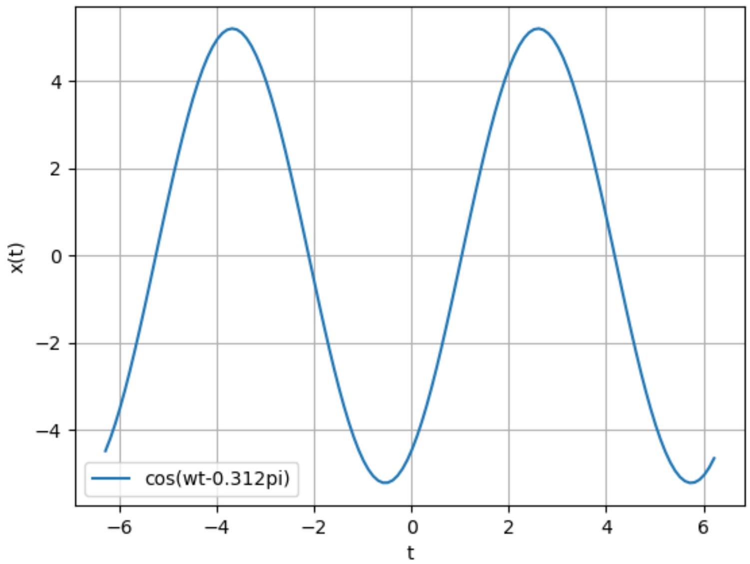
$$
\begin{aligned}
z_1(t) &= Re\{3e^{j(wt-\frac{2}{3}\pi)}\} = Re\{3e^{jwt}e^{-j\frac{2}{3}\pi}\} \Rightarrow X_1 = 3e^{-j\frac{2}{3}\pi} \\
z_2(t) &= \frac{e^{j(wt-\frac{1}{2}\pi)} + e^{-j(wt-\frac{1}{2}\pi))}}{4} = \frac{1}{2}\cos(wt - \frac{\pi}{2}) \Rightarrow X_2 = \frac{1}{2}e^{-j\frac{\pi}{2}} \\
z_3(t) &= 4\sin(wt - \frac{2\pi}{3}) = 4(\frac{1}{2j}e^{jwt}e^{-j\frac{2}{3}\pi} - \frac{1}{2j}e^{-jwt}e^{j\frac{2}{3}\pi}) \\
&= 4(\frac{1}{2}e^{jwt}e^{-j\frac{2}{3}\pi}e^{-j\frac{\pi}{2}} - \frac{1}{2}e^{-jwt}e^{j\frac{2}{3}\pi}e^{-j\frac{\pi}{2}}) = 2e^{jwt}(e^{j\frac{5\pi}{6}} + e^{-j\frac{5\pi}{6}}) \\
&= 4\cos(wt + \frac{5\pi}{6}) \Rightarrow X_3 = 4e^{j\frac{5\pi}{6}}
\end{aligned}
$$

$$X = X_1 + 2X_2 + X_3 = 3e^{-j\frac{2\pi}{3}} + e^{-j\frac{\pi}{2}} + 4e^{j\frac{5\pi}{6}}$$
$$X_{i_{rectangular}} = r_i \cos(\phi_i) + jr_i \sin(\phi_i) \Rightarrow$$
$$X_{1_{rec}} = -1.5 - 2.6j \wedge X_{2_{rec}} = 0 - j \wedge X_{3_{rec}} = -3.46 - 2j$$
$$X_{rec} = -1.5 - 2.6j - 0 - j - 3.46 + 2j = -4.96 - 1.6j$$
$$X_A = \sqrt{(-4.92)^2 + (-1.6)^2} = 5.2 \wedge X_\phi = \pi - \arctan(\frac{-1.6}{-4.96}) = 2.829\pi$$
$$x(t) = 5.2\cos(wt - 2.829\pi)$$

(b) Phasors of $x(t)$ in the complex plane



(c) Plot of $x(t)$ from -2π to 2π with $w = 1$



Python Snippet to Draw the Signal

```
import matplotlib.pyplot as plt
import numpy as np

w = 1
t = np.arange(-2 * np.pi, 2 * np.pi, 0.1)  # start,stop,step
x_t = 5.21*np.cos(w*t - 2.829 * np.pi)
plt.plot(t, x_t)
plt.grid()
plt.xlabel('t')
plt.ylabel('x(t)')
plt.legend(['cos(wt-2.829pi)'])
plt.show()
```

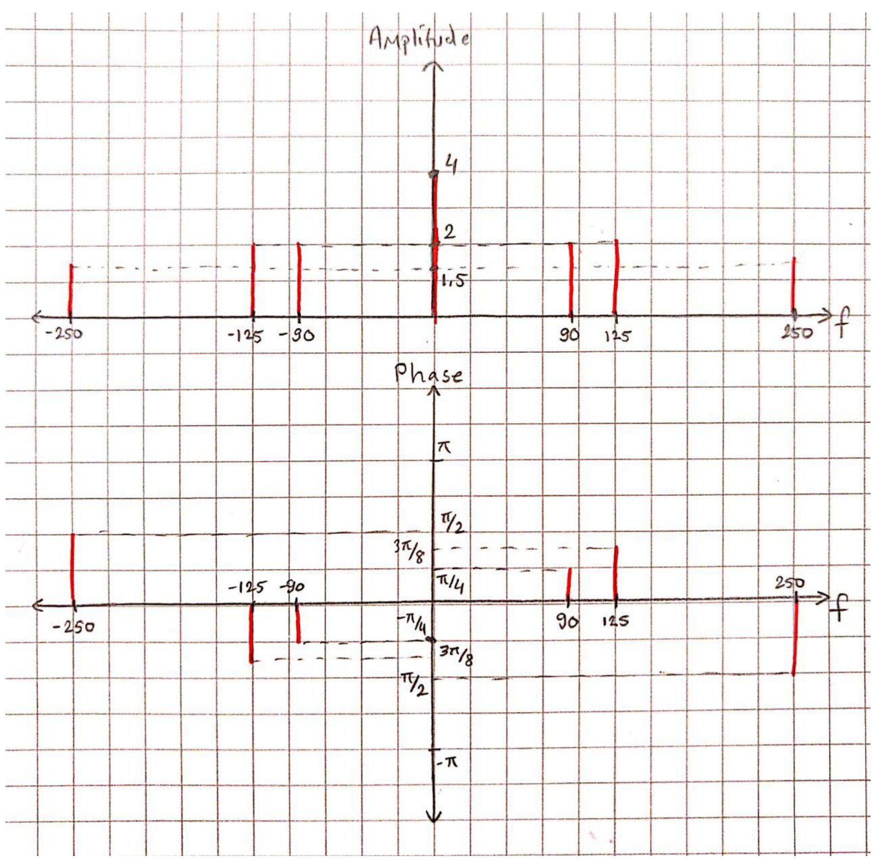
Chapter 3: Spectrum Representation

Question 3

(a) Calculating and sketching the $x(t) = 4z_1(t) - 4z_2(t) + z_3(t) + 2z_4(t)$

$$z_1 = j(2\cos(\frac{11\pi}{12}))(2j\sin(\frac{11\pi}{12})) = -4\cos(\frac{11\pi}{12})\sin(\frac{\pi}{12}) = -2(\sin(\pi) - \sin(\frac{5\pi}{6}))$$
$$= 2\sin(\frac{5\pi}{6}) = 1$$
$$z_2 = -\cos(180\pi t + \frac{\pi}{4})$$
$$z_3 = 3\cos(500\pi t - \frac{\pi}{2}) = 3\sin(500\pi t)$$
$$z_4 = 2\cos(250\pi t + \frac{3\pi}{8})$$

$$x(t) = 4 + 4\cos(180\pi t + \frac{\pi}{4}) + 3\sin(500\pi t) + 4\cos(250\pi t + \frac{3\pi}{8})$$



(b) Yes, $x(t)$ is periodic. The period $T = \frac{1}{f_0} = \frac{1}{5} = 0.2s$ because greatest common divisor of the frequencies $f_{z_2} = 90\text{Hz}, f_{z_3} = 125\text{Hz}$ and $f_{z_4} = 250\text{Hz}$ is $f_0 = 5\text{Hz}$.

(c) Fundamental frequency $f_0 = 5Hz$. There exists three following harmonics: 18^{th}, 25^{th} and 50^{th}.

Question 4

$$\int_0^{T_0} v_k(t)v_l^*(t)\mathrm{d}t = \begin{cases} 0, & k \neq l \\ T_0, & k = l \end{cases} \tag{1}$$

The orthogonality property is defined as (1). By using the property A can be calculated as follows:

$$\begin{aligned}
\int_{-\infty}^{\infty} x_1(t)x_2(t)\mathrm{d}t &= \int_{-\infty}^{0} e^{3t}(1 - Ae^{9t})\mathrm{d}t + \int_0^{\infty} e^{-3t}(1 - Ae^{-9t})\mathrm{d}t \\
&= \int_{-\infty}^{0} e^{3t} - Ae^{12t}\mathrm{d}t + \int_0^{\infty} e^{-3t} - Ae^{-12t}\mathrm{d}t \\
&= \frac{e^{3t}}{3}\bigg|_{-\infty}^{0} - \frac{e^{-3t}}{3}\bigg|_{0}^{\infty} - A\frac{e^{12t}}{12}\bigg|_{-\infty}^{0} + A\frac{e^{-12t}}{12}\bigg|_{0}^{\infty} \\
&= \frac{2}{3} - \frac{A}{6} = \frac{4 - A}{6} \Rightarrow 4 - A = 0 \Rightarrow A = 4
\end{aligned}$$

As the result, $A = 4$ is obtained.

Question 5

(a) i) Fourier series coefficients for the first signal

$$x(t) = \begin{cases} 0.5, & -5 < t < 5 \\ 0, & 5 < t < 15 \end{cases} \wedge T_0 = 20 \Rightarrow$$

$$\begin{aligned}
a_k &= \frac{1}{20}\int_{-5}^{5} 0.5e^{-j2\pi k0.05t}\mathrm{d}t + \frac{1}{20}\int_5^{15} e^{-j2\pi k0.05t}\mathrm{d}t \\
&= \frac{-1}{40jk\pi0.1}e^{-jk\pi0.1t}\bigg|_{-5}^{5} + \frac{-1}{20jk\pi0.1}e^{-jk\pi0.1t}\bigg|_{5}^{15} \\
&= \frac{-1}{4jk\pi}(e^{-jk0.5\pi} - e^{jk0.5\pi}) + \frac{-1}{2jk\pi}(e^{-jk0.5\pi}e^{-jk\pi} - e^{-jk0.5\pi}) \\
&= \frac{-\sin(\frac{\pi k}{2})}{2\pi k} - \frac{e^{-jk0.5\pi}(e^{-jk\pi} - 1)}{2\pi k e^{\frac{j\pi}{2}}} \\
&= \frac{-\sin(\frac{\pi k}{2} + e^{\frac{-j\pi(k+1)}{2}}(e^{-jk\pi} - 1))}{2\pi k} \\
a_0 &= \frac{1}{20}\int_{-5}^{5} 0.5\mathrm{d}t + \frac{1}{20}\int_5^{15} 1\mathrm{d}t = \frac{1}{20}((2.5 + 2.5) + (15 - 5)) = 0.75
\end{aligned}$$

ii) Fourier series coefficients for the second signal

$$x(t) = \begin{cases} 2+t, & -2 < t < 0 \\ 2-2t, & 0 < t < 1 \end{cases} \wedge T_0 = 3 \Rightarrow$$

$$a_k = \frac{1}{3}\int_{-2}^{0}(2+t)e^{-j2\pi k\frac{1}{3}t}\mathrm{d}t + \frac{1}{3}\int_{0}^{1}(2-2t)e^{-j2\pi k\frac{1}{3}t}\mathrm{d}t\mathrm{d}t$$

$$= 2\int_{-2}^{0} e^{\frac{-2}{3}j\pi kt}\mathrm{d}t + \int_{-2}^{0} te^{\frac{-2}{3}j\pi kt}\mathrm{d}t - 2\int_{0}^{1} te^{\frac{-2}{3}j\pi kt}\mathrm{d}t + 2\int_{0}^{1} e^{\frac{-2}{3}j\pi kt}\mathrm{d}t$$

Integrating by parts using the rule of product for differentiation

$$\int u\mathrm{d}v = uv - \int v\mathrm{d}u$$

$$a_k = \frac{2jte^{\frac{-2}{3}j\pi kt}}{\pi k} + (2+\frac{2j}{\pi k})\int_{-2}^{0} e^{\frac{-2}{3}j\pi kt}\mathrm{d}t - \frac{3jte^{\frac{-2}{3}j\pi kt}}{\pi k} + (2+\frac{3j}{\pi k})\int_{0}^{1} e^{\frac{-2}{3}j\pi kt}\mathrm{d}t$$

$$= \left.\frac{e^{\frac{-2}{3}j\pi kt}(-3-4j\pi k(t-1))}{3\pi^2k^2}\right|_{-2}^{0} - \left.\frac{e^{\frac{-2}{3}j\pi kt}(-9-6j\pi k(t-1))}{2\pi^2k^2}\right|_{0}^{1}$$

$$= \frac{4j\pi k - 3e^{j\frac{4}{3}\pi k} + 3}{4\pi^2k^2} - \frac{6j\pi k - 9e^{j\frac{2}{3}\pi k} - 9}{6\pi^2k^2}$$

$$a_0 = \frac{1}{3}\int_{-2}^{0}(2+t)\mathrm{d}t + \frac{1}{3}\int_{0}^{1}(2-2t)\mathrm{d}t = \left.\frac{2t}{3}\right|_{-2}^{0} + \left.\frac{t^2}{6}\right|_{-2}^{0} + \left.\frac{2t}{3}\right|_{0}^{1} - \left.\frac{t^2}{6}\right|_{0}^{1}$$

$$= \frac{4-1}{3} = 1$$

iii) Fourier series coefficients for the third signal

$$x(t) = \begin{cases} 2-0.5\sin(\frac{\pi t}{2}), & 0 < t < 2 \\ 0, & 2 < t < 4 \end{cases} \wedge T_0 = 4 \Rightarrow$$

$$a_k = \frac{1}{4}\int_{0}^{2} 2-0.5\sin(\frac{\pi t}{2})e^{-j2\pi k\frac{1}{4}t}\mathrm{d}t = -\frac{1}{8}\int_{0}^{2} e^{-j\frac{\pi kt}{2}}\sin(\frac{\pi t}{2})\mathrm{d}t + \frac{1}{2}\int_{0}^{2} e^{-j\frac{\pi kt}{2}}\mathrm{d}t$$

Integrating by parts using the rule of product for differentiation

$$\int u\mathrm{d}v = uv - \int v\mathrm{d}u$$

$$a_k = \left.\left(-\frac{1}{8}\frac{e^{j\frac{\pi}{2}kt}(-\frac{1}{2}\cos(\frac{\pi t}{2}) - j\frac{\pi}{2}k\sin(\frac{\pi t}{2}))}{\frac{\pi^2}{4} - 2.47k^2} - \frac{e^{-j\frac{\pi}{2}kt}}{j\pi k}\right)\right|_{0}^{2}$$

$$= \frac{e^{-j\pi k}}{4(k^2-1)}(0.32(e^{j\pi k}+1)) + \frac{1}{2k}(0.64(e^{-j\pi k}-1))$$

$$a_0 = \frac{1}{4}\int_{0}^{2} 2-0.5\sin(\frac{\pi t}{2})\mathrm{d}t = \frac{1}{4}\left(2t + 0.32\cos(\frac{\pi t}{2})\Big|_{0}^{1}\right) = 0.84$$

iv) Fourier series coefficients for the forth signal

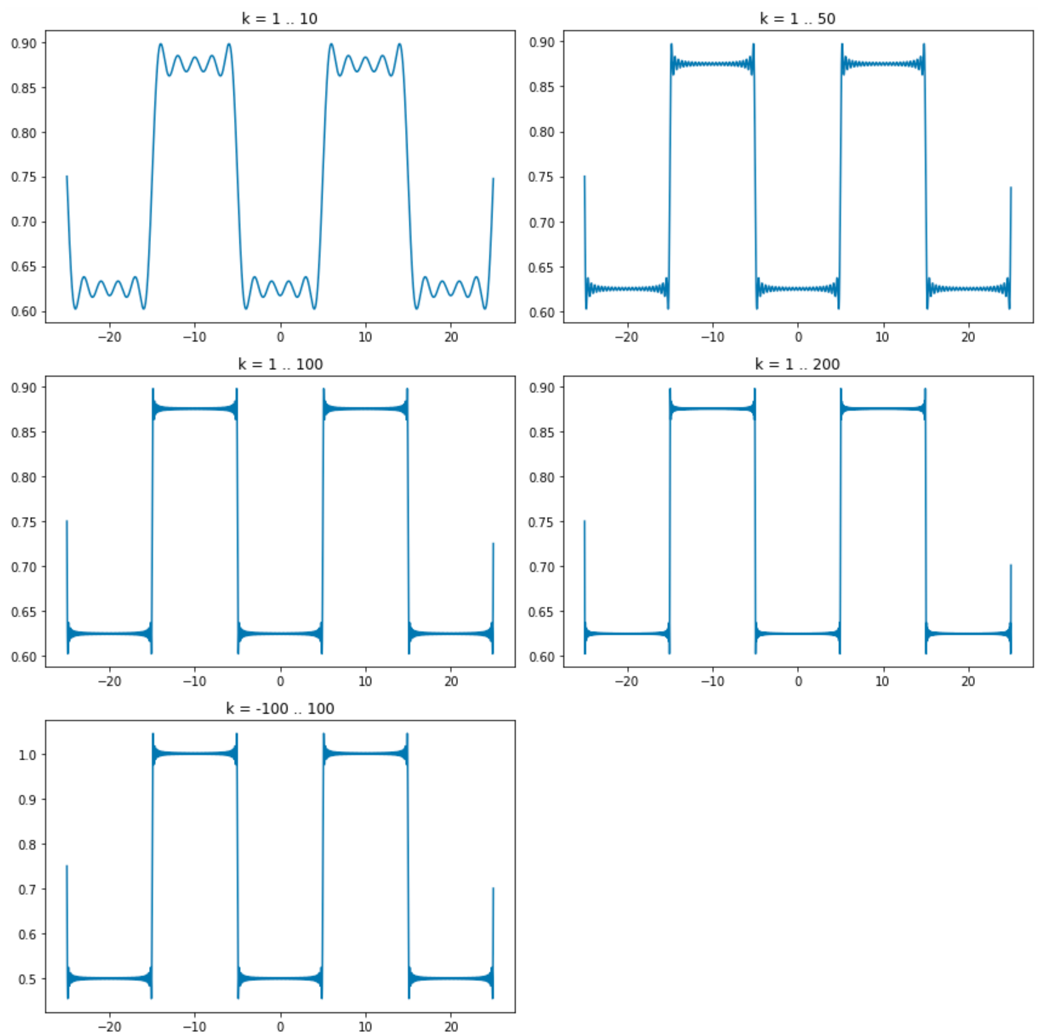
$$x(t) = \begin{cases} e^{-t}, & 0 < t < 1 \\ 0, & 1 < t < 2 \end{cases} \wedge T_0 = 2 \Rightarrow$$

$$a_k = \frac{1}{2}\int_0^1 e^{-t}e^{-j\pi kt}\mathrm{d}t = \frac{-1}{2jk\pi + 1}e^{-t(jk\pi+1)}\Big|_0^1 = \frac{1 - e^{-j\pi k - 1}}{2(j\pi k + 1)}$$

$$a_0 = \frac{1}{2}\int_0^1 e^{-t}\mathrm{d}t = \frac{-1}{2}e^{-t}\Big|_0^1 = -\frac{1}{2}(\frac{1}{e} - 1) = 0.316$$

(b) Graphical results of the signals plotted using a_0, a_k and T_0 variables obtained in Question 5(a) using the synthesis function $x(t) = \sum_{k=-\infty}^{\infty} a_k e^{j2\pi kt/T_0}$, and code snippets that plot the function are given below.

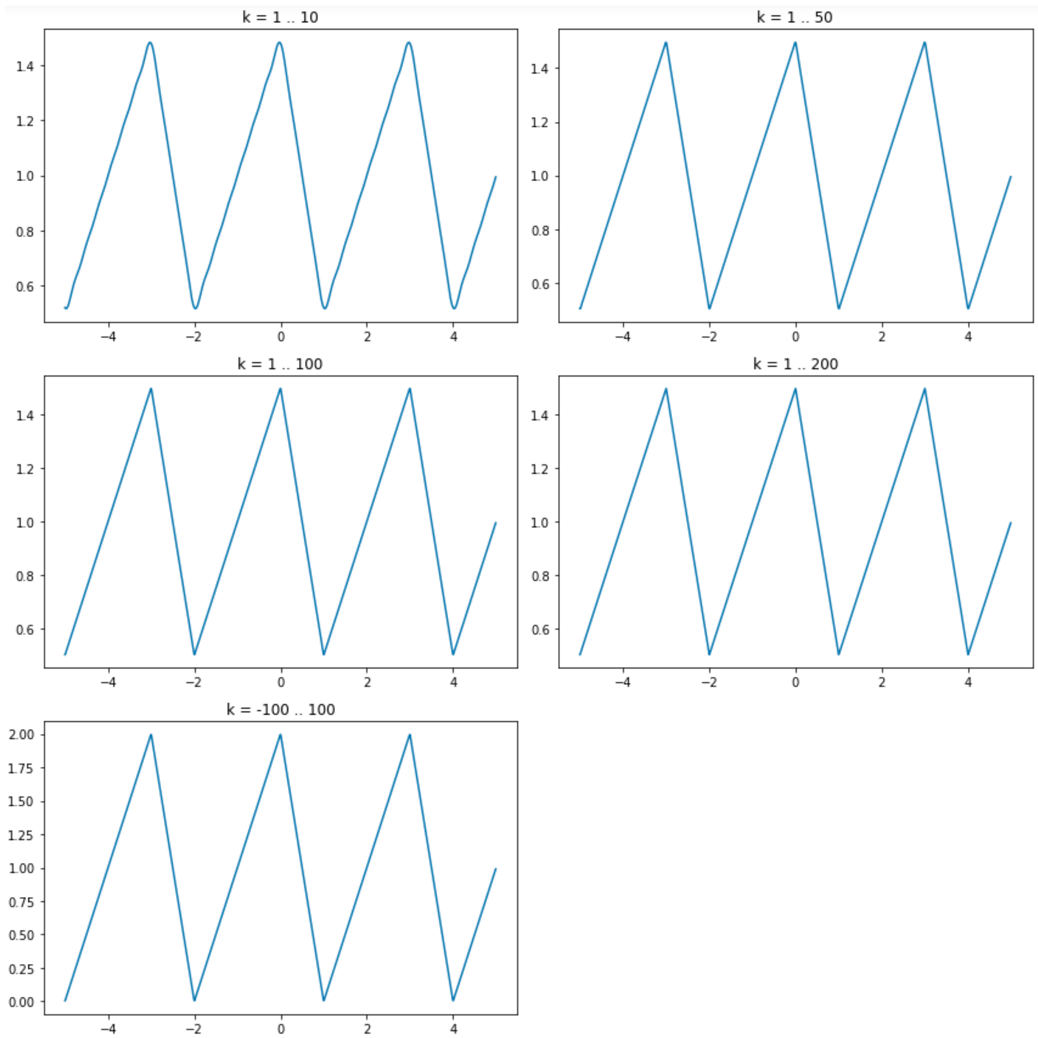
i) Plots and the code snippet for the first signal with varying values for the variable k



Python Snippet to Draw the First Signal

```
k_list = [(1,10),(1,50),(1,100),(1,200),(-100,100)]

def a_k_1(k):
    return (-1 / (6 * np.pi * k)) * (np.sin(np.pi * k / 2) + np.exp(-1j * np.pi / 2 * (k + 1)) * (np.exp(-1j * np.pi * k) - 1))

a_0_1 = 0.75
T_0_1 = 20

f_0_1 = 1 / T_0_1
t_1 = np.arange(-25, 25, 0.01)

fig, axes = plt.subplots(nrows=3, ncols=2, figsize=(12, 12))

for index, k_tuple in enumerate(k_list):
    k_min = k_tuple[0]
    k_max = k_tuple[1]
    x_t_1 = a_0_1
    for k in range(k_min, k_max + 1):
        if not k == 0:
            x_t_1 += a_k_1(k) * np.exp(1j * 2 * np.pi * k * f_0_1 * t_1)
    axes_index = (index // 2, index % 2)
    axes[axes_index].plot(t_1, x_t_1)
    title = "k = " + str(k_min) + " .. " + str(k_max)
    axes[axes_index].set_title(title)

axes[2, 1].remove()

fig.tight_layout()
plt.show()
```

ii) Plots and the code snippet for the second signal with varying values for the variable k

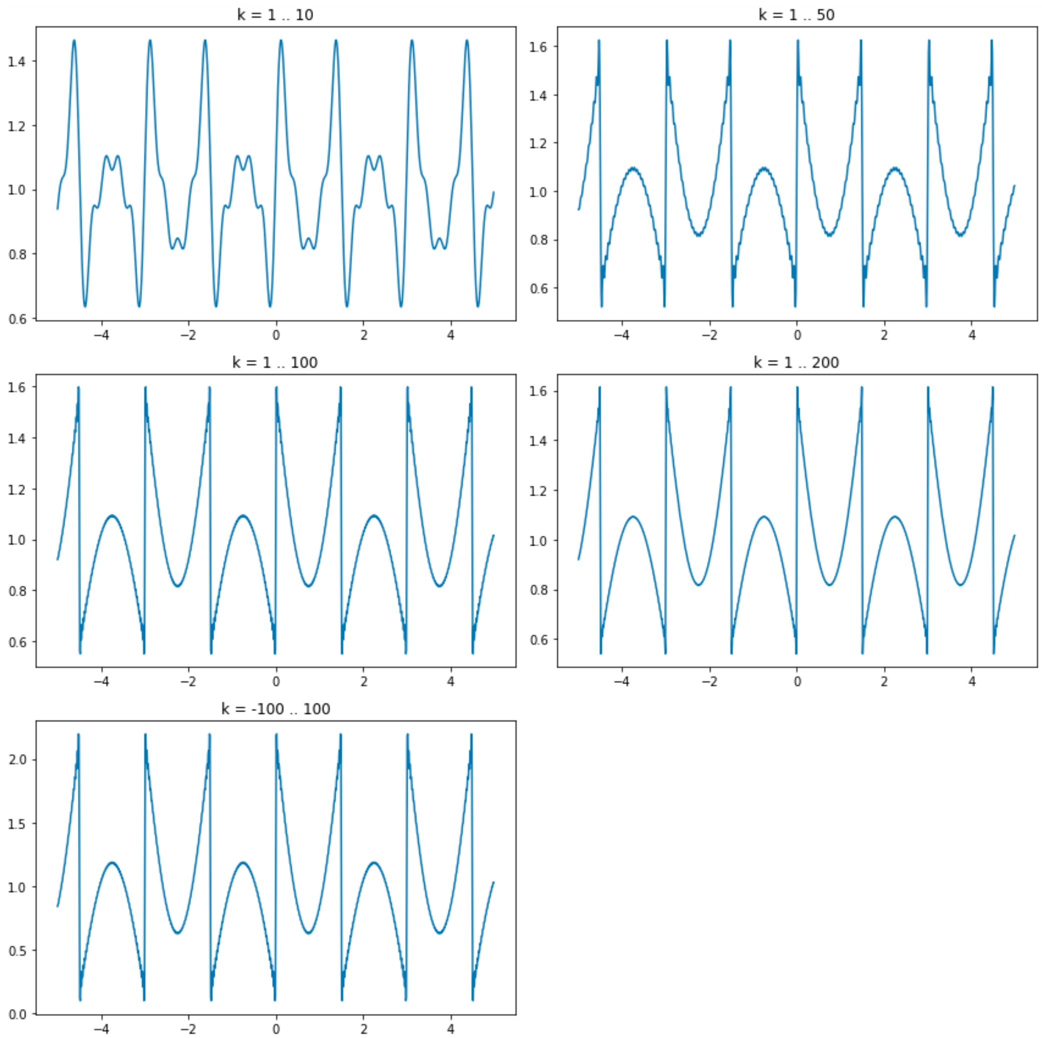


Python Snippet to Draw the Second Signal

```python
k_list = [(1,10),(1,50),(1,100),(1,200),(-100,100)]

def a_k_2(k):
    return (4j * np.pi * k - 3 * np.exp(1j * 4/3 * np.pi * k) + 3)/(2 * np.pi ** 2 * k ** 2) - (6j * np.pi * k + 9 * np.exp(-1j * 2/3 * np.pi * k) - 9)/(3 * np.pi ** 2 * k ** 2)

a_0_2 = 1
T_0_2 = 3

```

```
    f_0_2 = 1 / T_0_2
    t_2 = np.arange(-5, 5, 0.01)

    fig, axes = plt.subplots(nrows=3, ncols=2, figsize=(12, 12))

    for index, k_tuple in enumerate(k_list):
        k_min = k_tuple[0]
        k_max = k_tuple[1]
        x_t_2 = a_0_2
        for k in range(k_min, k_max + 1):
            if not k == 0:
                x_t_2 += a_k_2(k) * np.exp(1j * 2 * np.pi * k * f_0_2 * t_2)
        axes_index = (index // 2, index % 2)
        axes[axes_index].plot(t_2, x_t_2)
        title = "k = " + str(k_min) + " .. " + str(k_max)
        axes[axes_index].set_title(title)

    axes[2, 1].remove()

    fig.tight_layout()
    plt.show()
```

iii) Plots and the code snippet for the third signal with varying values for the variable k

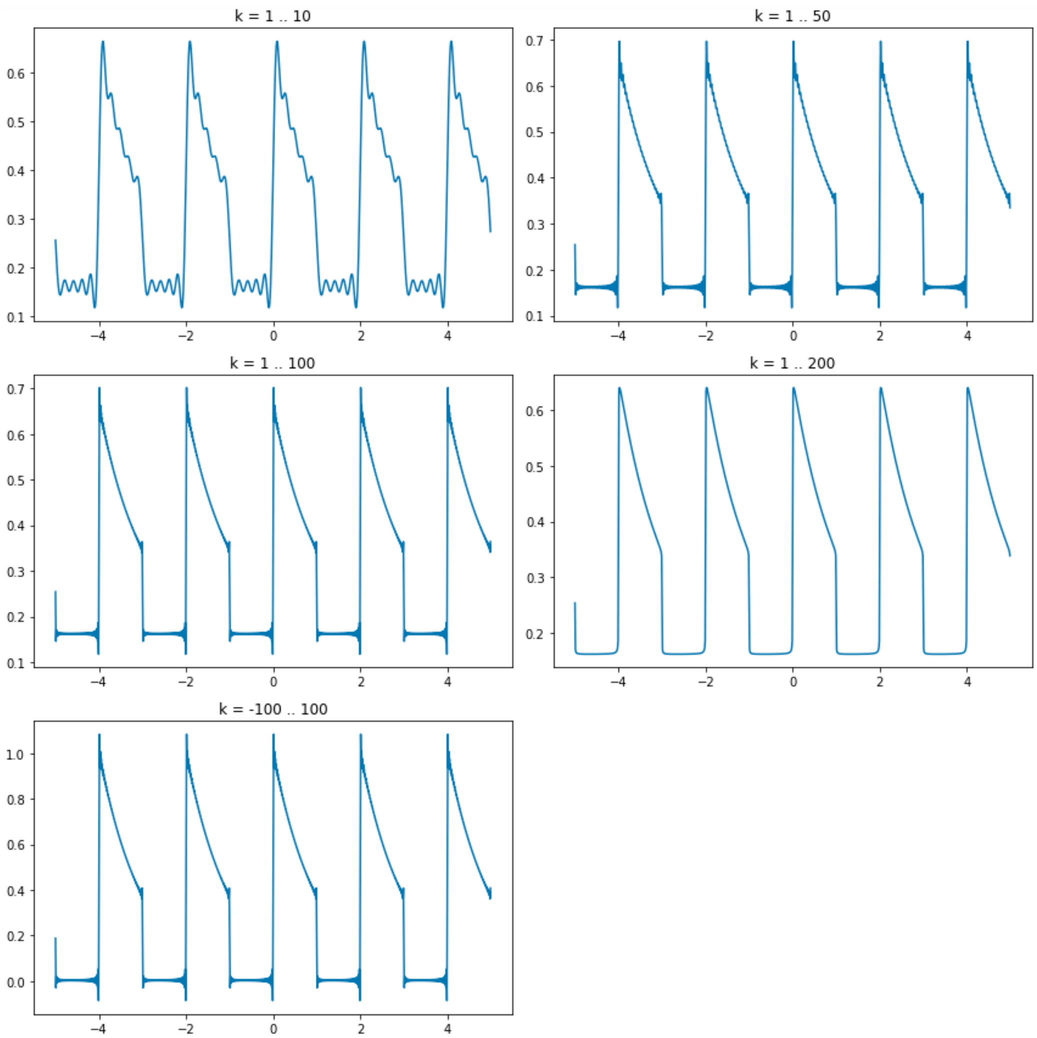


Python Snippet to Draw the Third Signal

```python
k_list = [(1,10),(1,50),(1,100),(1,200),(-100,100)]

def a_k_3(k):
    return 0.25 * (np.exp(-1j * np.pi * k)/(k**2 - 1) * (0.32 * (np.exp(1j * np.pi * k) + 1)) + (2 / k) * (0.64 * 1j * (np.exp(-1j * np.pi * k) - 1)))

a_0_3 = 1
T_0_3 = 3

f_0_3 = 1 / T_0_3
```

```python
    t_3 = np.arange(-5, 5, 0.01)

    fig, axes = plt.subplots(nrows=3, ncols=2, figsize=(12, ↩
        12))

    for index, k_tuple in enumerate(k_list):
        k_min = k_tuple[0]
        k_max = k_tuple[1]
        x_t_3 = a_0_3
        for k in range(k_min, k_max + 1):
            if k != 0 and k * k != 1:
                x_t_3 += a_k_3(k) * np.exp(1j * 2 * np.pi * k↩
                    * f_0_3 * t_3)
        axes_index = (index // 2, index % 2)
        axes[axes_index].plot(t_3, x_t_3)
        title = "k = " + str(k_min) + " .. " + str(k_max)
        axes[axes_index].set_title(title)

    axes[2, 1].remove()

    fig.tight_layout()
    plt.show()
```

iv) Plots and the code snippet for the forth signal with varying values for the variable k



Python Snippet to Draw the Forth Signal

```
k_list = [(1,10),(1,50),(1,100),(1,200),(-100,100)]

def a_k_4(k):
    return 0.5 * ((1 - np.exp(-1j * np.pi * k - 1)) / (1j * np.pi * k + 1))

a_0_4 = 0.32
T_0_4 = 2

f_0_4 = 1 / T_0_4
t_4 = np.arange(-5, 5, 0.01)
```

```

    fig, axes = plt.subplots(nrows=3, ncols=2, figsize=(12, 12))

    for index, k_tuple in enumerate(k_list):
        k_min = k_tuple[0]
        k_max = k_tuple[1]
        x_t_4 = a_0_4
        for k in range(k_min, k_max + 1):
            if k != 0:
                x_t_4 += a_k_4(k) * np.exp(1j * 2 * np.pi * k * f_0_4 * t_4)
        axes_index = (index // 2, index % 2)
        axes[axes_index].plot(t_4, x_t_4)
        title = "k = " + str(k_min) + " .. " + str(k_max)
        axes[axes_index].set_title(title)

    axes[2, 1].remove()

    fig.tight_layout()
    plt.show()
```

Graphical results are obtained as close as possible to the signals given in the Question 5. However, due to the computational limitations and the range given for the variable k in Question 5(b)i-v. plotted signals are not the exact representations of the ones given in the Question 5. Nevertheless, they are close enough approximations so much so that conclusions driven from them are on point.

In the value range 1-10 for k, taken sinusoidal harmonics are observable and therefore not yet enough for a close approximating signal. Same story holds true for the range 1-50. On the other hand, ranges of k, 1-100 and 1-200 are very close approximations of the original signal. Therefore they are distinctly more suitable as higher resolution syntheses.

It should be mentioned that none of the plots scale correctly for the ranges of k 1-10, 1-50, 1-100 and 1-200. This is due to negative k are not being rendered, resulting in inadequate representations. In order to achieve the correct results, k should be given a value range from negative to positive just like in Question 5(b)v (-100 to 100). Notice once a truthful range is chosen for k, $x(t)$ scales correctly.

Moreover, the results obtained for Question 5(b)(iii) seem to not represent the given signal properly. Such fallacies can be caused by either wrong derivations of a_ks or an error in the given question.

Question 6

$$x(t) = \sum_{k=-\infty}^{\infty} a_k e^{j2\pi k f_0 t} \text{ where } f = \frac{1}{T} = \frac{1}{4} \wedge a_k = \begin{cases} jk, & |k| < 3 \\ 0, & \text{otherwise} \end{cases}$$

$$\text{Then, } x(t) = \sum_{k=-2}^{2} jke^{j\pi k 0.5t} = -2je^{-j\pi t} - je^{-j\pi 0.5t} + je^{j\pi 0.5t} + -2je^{j\pi t}$$
$$= -2sin(0.5\pi t) - 4sin(\pi t)$$