Homework Signal 1

Week 1

6733172621 Patthadon Phengpinij

Collaborators. ChatGPT (for LaTeX styling and grammar checking)

1 Representing Signals

Problem 1. Sketch the following signals

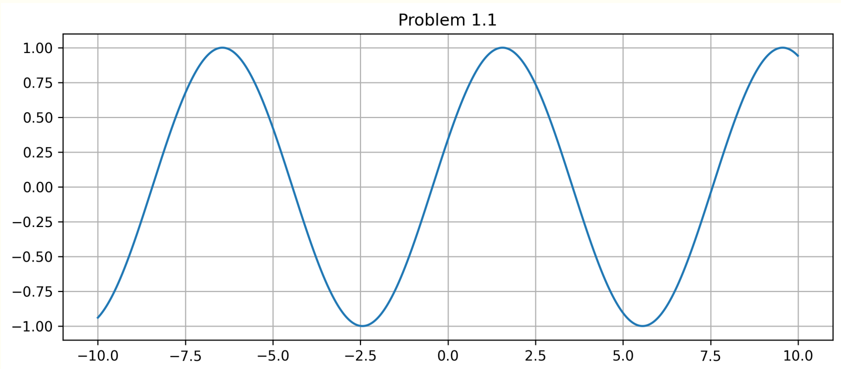
TO SUBMIT

a) $x(t) = \sin \frac{\pi}{4}t + 20°$

Solution. Using Python and Matplotlib to plot the signal $x(t) = \sin \frac{\pi}{4}t + 20°$:

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

fig = plt.figure()

t = np.arange(-10, 10, 0.01)
x = np.sin(np.pi/4 * t + np.pi/9)

plt.title("Problem 1.1")
plt.plot(t, x)
plt.grid(True)
plt.show()
```

The plot of the signal is shown below:

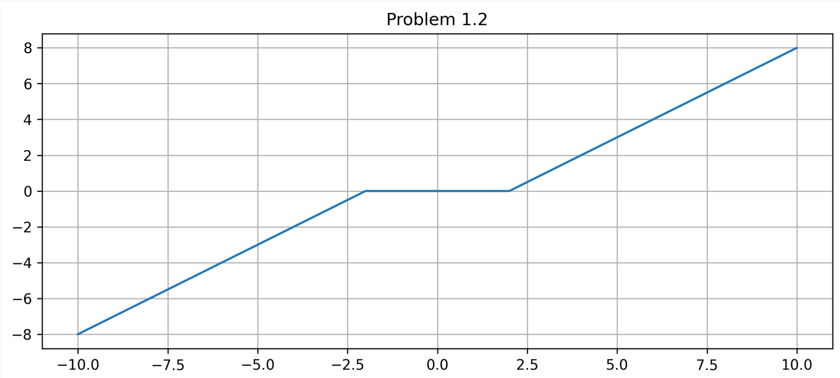


b) $x(t) = \begin{cases} t+2, & t \le 2 \\ 0, & -2 \le t \le 2 \\ t-2, & t \ge 2 \end{cases}$

Solution. Using Python and Matplotlib to plot the piecewise signal $x(t)$:

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

fig = plt.figure(figsize=(10, 4))

t = np.arange(-10, 10, 0.01)
x = np.piecewise(t, [t < -2, (t >= -2) & (t < 2), t >=
    2], [lambda t: t + 2, 0, lambda t: t - 2])

plt.title("Problem 1.2")
plt.plot(t, x)
plt.grid(True)
plt.show()
```

The plot of the signal is shown below:



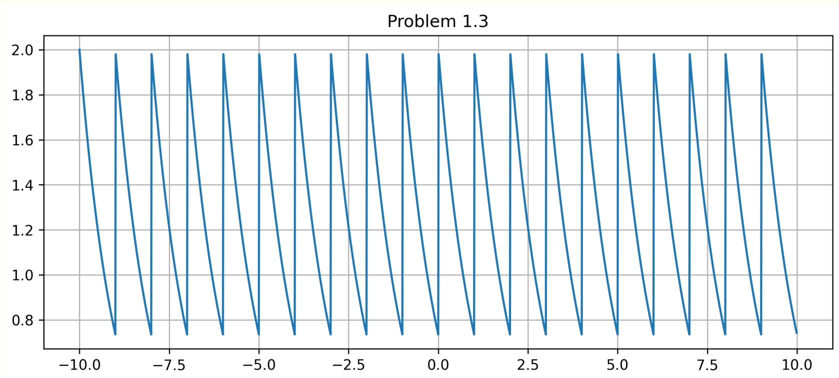
TO SUBMIT

c) $x(t) = 2e^{-t}, 0 \leq t < 1$ and $x(t+1) = x(t), \forall t$

Solution. Using Python and Matplotlib to plot the piecewise signal $x(t) = 2e^{-t}, 0 \leq t < 1$ and $x(t+1) = x(t), \forall t$:

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

def x3(t):
    if t >= 1:
        return x3(t - 1)
    if t < 0:
        return x3(t + 1)
    return 2 * (np.e ** (-t))

fig = plt.figure(figsize=(10, 4))

t = np.arange(-10, 10, 0.01)
x3_vectorize = np.vectorize(x3)
x = x3_vectorize(t)

plt.title("Problem 1.3")
plt.plot(t, x)
plt.grid(True)
plt.show()
```

The plot of the signal is shown below:

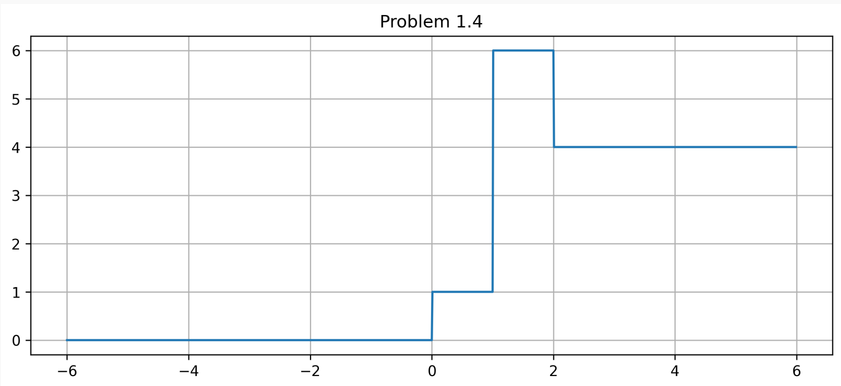


d) $x(t) = u(t) + 5u(t-1) + 2u(t-2)$

Solution. Using Python and Matplotlib to plot the piecewise signal $x(t) = u(t) + 5u(t-1) + 2u(t-2)$:

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

def unit_signal(t):
    return 1.0 if t >= 0 else 0.0

unit_signal_vectorize = np.vectorize(unit_signal)

fig = plt.figure(figsize=(10, 4))

t = np.arange(-6, 6, 0.01)

u1 = unit_signal_vectorize(t)
u2 = unit_signal_vectorize(t - 1)
u3 = unit_signal_vectorize(t - 2)

x = u1 + 5 * u2 - 2 * u3

plt.title("Problem 1.4")
plt.plot(t, x)
plt.grid(True)
plt.show()
```

The plot of the signal is shown below:



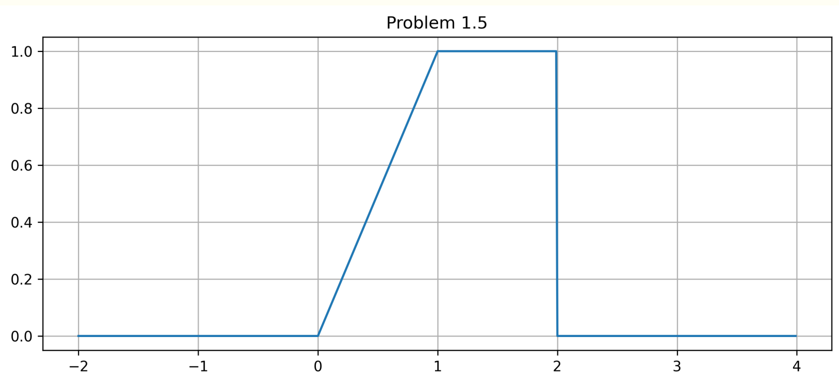
TO SUBMIT

e) $x(t) = r(t) - r(t-1) - u(t-2)$

Solution. Using Python and Matplotlib to plot the piecewise signal $x(t) = r(t) - r(t-1) - u(t-2)$:

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

def unit_signal(t):
    return 1.0 if t >= 0 else 0.0

def ramp_signal(t):
    return t * unit_signal(t)

unit_signal_vectorize = np.vectorize(unit_signal)
ramp_signal_vectorize = np.vectorize(ramp_signal)

fig = plt.figure(figsize=(10, 4))

t = np.arange(-2, 4, 0.01)

r1 = ramp_signal_vectorize(t)
r2 = ramp_signal_vectorize(t - 1)
u1 = unit_signal_vectorize(t - 2)

x = r1 - r2 - u1

plt.title("Problem 1.5")
plt.plot(t, x)
plt.grid(True)
plt.show()
```

The plot of the signal is shown below:



Problem 2. Determine whether each of following signals is periodic, and if so, find its period.

a) $x(t) = \sin\left(\frac{\pi}{3}t\right) + \cos\left(\frac{8\pi}{3}t\right)$

Solution. Consider each part of the signal separately:

$$\sin\left(\frac{\pi}{3}t\right) \text{ has a period of } T_1 = \frac{2\pi}{\frac{\pi}{3}} = 6$$

$$\cos\left(\frac{8\pi}{3}t\right) \text{ has a period of } T_2 = \frac{2\pi}{\frac{8\pi}{3}} = \frac{3}{4}$$

Considering the least common multiple of the two periods:

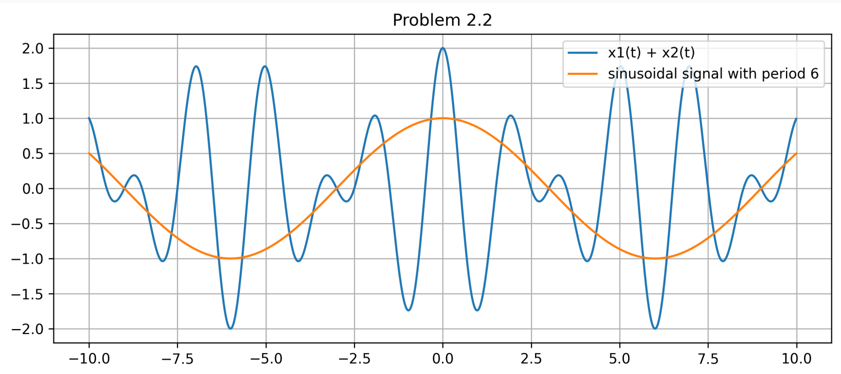
$$T = \operatorname{lcm}(T_1, T_2) = \operatorname{lcm}(6, \frac{3}{4}) = 6$$

Thus, the signal $x(t) = \sin\left(\frac{\pi}{3}t\right) + \cos\left(\frac{8\pi}{3}t\right)$ $\boxed{\textbf{is periodic}}$ with a period of $\boxed{T = 6}$.

By using Python and Matplotlib, we can visualize the periodicity of the signal:

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

fig = plt.figure(figsize=(10, 4))

t = np.arange(-10, 10, 0.01)

x1 = np.sin(np.pi/3 * t)
x2 = np.cos(8*np.pi/3 * t)

x = np.sin(np.pi/3 * t)

plt.title("Problem 2.1")
plt.plot(t, x1 + x2, label="x1(t) + x2(t)")
plt.plot(t, x, label="sinusoidal signal with period 6")
plt.grid(True)
plt.legend(loc="upper right")
plt.show()
```

The plot of the signal is shown below:



b) $x(t) = \exp\left(j\frac{7\pi}{6}t\right) + \exp\left(j\frac{5\pi}{6}t\right)$

Solution. Consider each part of the signal separately:

$$\exp\left(j\frac{7\pi}{6}t\right) \text{ has a period of } T_1 = \frac{2\pi}{\frac{7\pi}{6}} = \frac{12}{7}$$

$$\exp\left(j\frac{5\pi}{6}t\right) \text{ has a period of } T_2 = \frac{2\pi}{\frac{5\pi}{6}} = \frac{12}{5}$$

Considering the least common multiple of the two periods:

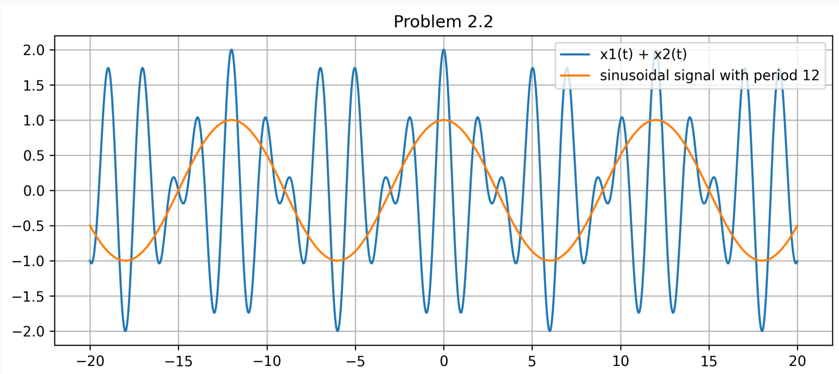
$$T = \text{lcm}(T_1, T_2) = \text{lcm}\left(\frac{12}{7}, \frac{12}{5}\right) = \frac{12}{1} = 12$$

Thus, the signal $x(t) = \exp\left(j\frac{7\pi}{6}t\right) + \exp\left(j\frac{5\pi}{6}t\right)$ $\boxed{\textbf{is periodic}}$ with a period of $\boxed{T = 12}$.

By using Python and Matplotlib, we can visualize the periodicity of the signal:

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

fig = plt.figure(figsize=(10, 4))

t = np.arange(-20, 20, 0.01)

x1 = np.exp(1j * 7*np.pi/6 * t)
x2 = np.exp(1j * 5*np.pi/6 * t)

x = np.exp(1j * np.pi/6 * t)

plt.title("Problem 2.2")
plt.plot(t, x1 + x2, label="x1(t) + x2(t)")
plt.plot(t, x, label="sinusoidal signal with period 12")
plt.grid(True)
plt.legend(loc="upper right")
plt.show()
```

The plot of the signal is shown below:



c) $x(t) = \exp\left(j\frac{7\pi}{6}t\right) + \exp\left(\frac{5\pi}{6}t\right)$

Solution. Consider each part of the signal separately:

$$\exp\left(j\frac{7\pi}{6}t\right) \text{ has a period of } T_1 = \frac{2\pi}{\frac{7\pi}{6}} = \frac{12}{7}$$

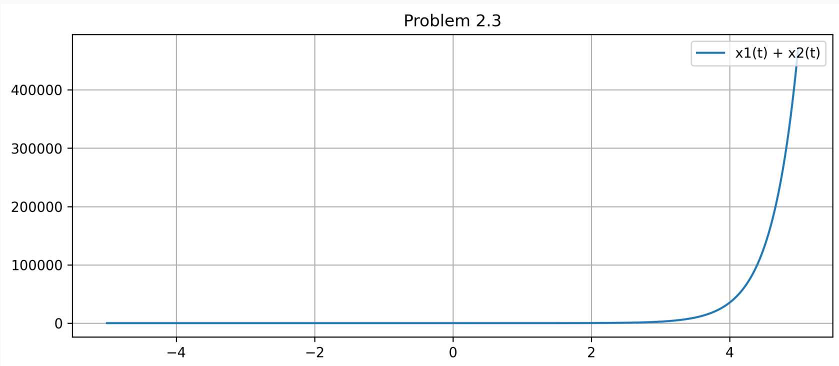
$$\exp\left(\frac{5\pi}{6}t\right) \text{ has no period since it is not a sinusoidal function. (non-periodic signal)}$$

Thus, the signal $x(t) = \exp\left(j\frac{7\pi}{6}t\right) + \exp\left(\frac{5\pi}{6}t\right)$ $\boxed{\textbf{is non-periodic}}$ since one part of the signal is non-periodic.

By using Python and Matplotlib, we can visualize the signal:

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

fig = plt.figure(figsize=(10, 4))

t = np.arange(-5, 5, 0.01)

x1 = np.exp(1j * 7*np.pi/6 * t)
x2 = np.exp(5*np.pi/6 * t)

plt.title("Problem 2.3")
plt.plot(t, x1 + x2, label="x1(t) + x2(t)")
plt.grid(True)
plt.legend(loc="upper right")
plt.show()
```

The plot of the signal is shown below:



Problem 3. Determine whether the following signals are power or energy signals or neither. Justify your answers

a) $x(t) = A\sin(t), -\infty < t < \infty$

Solution. Consider the energy of the signal:

$$
\begin{aligned}
E &= \lim_{N\to\infty} \int_{-N}^{N} |x(t)|^2 dt \\
&= \lim_{N\to\infty} \int_{-N}^{N} |A\sin(t)|^2 dt \\
&= \lim_{N\to\infty} A^2 \int_{-N}^{N} \sin^2(t) dt \\
&= \lim_{N\to\infty} A^2 \int_{-N}^{N} \frac{1-\cos(2t)}{2} dt \\
&= \lim_{N\to\infty} \frac{A^2}{2} \left[t - \frac{\sin(2t)}{2} \right]_{-N}^{N} \\
&= \lim_{N\to\infty} \frac{A^2}{2} (N - (-N)) \\
&= \lim_{N\to\infty} A^2 N \\
E &= \infty
\end{aligned}
$$

The integral diverges, so the energy is infinite.

Now, consider the power of the signal:

$$
\begin{aligned}
P &= \lim_{N\to\infty} \frac{1}{2N} \int_{-N}^{N} |x(t)|^2 dt \\
&= \lim_{N\to\infty} \frac{1}{2N} A^2 N \\
P &= \frac{A^2}{2}
\end{aligned}
$$

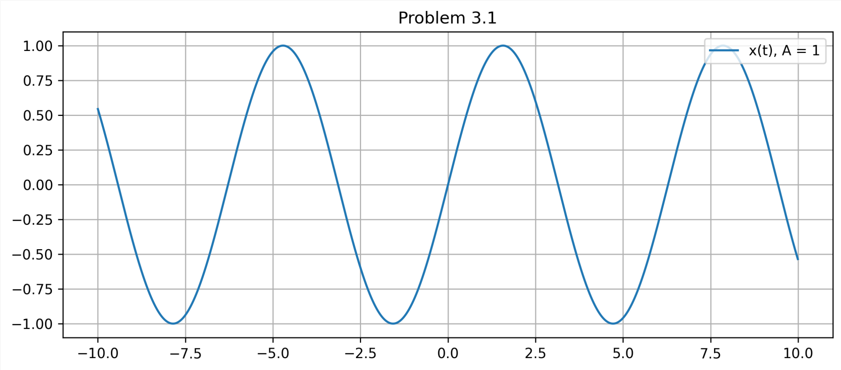
The integral converges to a finite value, so the power is finite.

Thus, the signal $x(t) = A\sin(t)$ is $\boxed{\textbf{a power signal}}$ with power $\boxed{P = \frac{A^2}{2}}$.

By using Python and Matplotlib, we can visualize the signal:

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

fig = plt.figure(figsize=(10, 4))

t = np.arange(-10, 10, 0.01)
x = np.sin(t)

plt.title("Problem 3.1")
plt.plot(t, x, label="x(t), A = 1")
plt.grid(True)
plt.legend(loc="upper right")
plt.show()
```

The plot of the signal is shown below:



b) $x(t) = A(u(t-a) - u(t+a))$, $a > 0$

Solution. Consider the energy of the signal:

$$\begin{aligned}
E &= \lim_{N\to\infty} \int_{-N}^{N} |x(t)|^2 dt \\
&= \lim_{N\to\infty} \int_{-N}^{N} |A(u(t-a) - u(t+a))|^2 dt \\
&= \lim_{N\to\infty} A^2 \int_{-N}^{N} |u(t-a) - u(t+a)|^2 dt \\
&= \lim_{N\to\infty} A^2 \int_{-N}^{N} (u(t-a) - u(t+a))^2 dt \\
&= \lim_{N\to\infty} A^2 \int_{-N}^{N} (u(t+a) - u(t-a)) dt \\
&= \lim_{N\to\infty} A^2 \int_{-a}^{a} 1 dt \\
&= \lim_{N\to\infty} A^2 (a - (-a)) \\
E &= 2aA^2
\end{aligned}$$

The integral converges to a finite value, so the energy is finite.

Now, consider the power of the signal:

$$\begin{aligned}
P &= \lim_{N\to\infty} \frac{1}{2N} \int_{-N}^{N} |x(t)|^2 dt \\
&= \lim_{N\to\infty} \frac{1}{2N} 2aA^2 \\
P &= 0
\end{aligned}$$

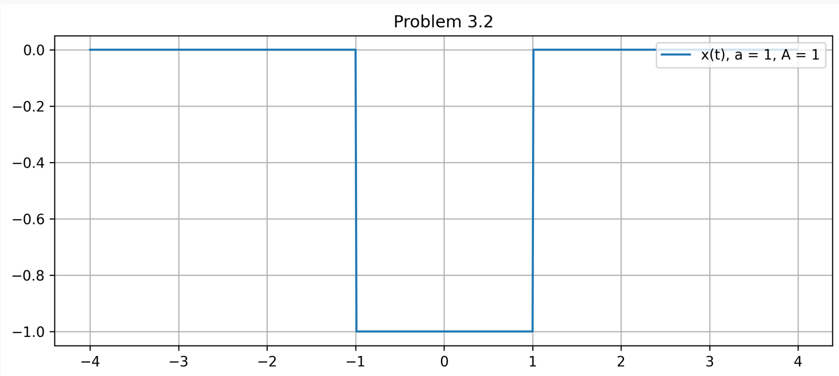
The integral converges to 0, so the power is 0.

Thus, the signal $x(t) = A(u(t-a) - u(t+a))$ is $\boxed{\textbf{a energy signal}}$ with energy $\boxed{E = 2aA^2}$.

By using Python and Matplotlib, we can visualize the signal:

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

def unit_signal(t):
    return 1.0 if t >= 0 else 0.0

unit_signal_vectorize = np.vectorize(unit_signal)

fig = plt.figure(figsize=(10, 4))

t = np.arange(-4, 4, 0.01)
x = unit_signal_vectorize(t - 1) - unit_signal_vectorize(
    t + 1)

plt.title("Problem 3.2")
plt.plot(t, x, label="x(t), a = 1, A = 1")
plt.grid(True)
plt.legend(loc="upper right")
plt.show()
```

The plot of the signal is shown below:



c) $x(t) = \exp(-at)u(t)$, $a > 0$

Solution. Consider the energy of the signal:

$$
\begin{aligned}
E &= \lim_{N\to\infty} \int_{-N}^{N} |x(t)|^2 dt \\
&= \lim_{N\to\infty} \int_{-N}^{N} |\exp(-at)u(t)|^2 dt \\
&= \lim_{N\to\infty} \int_{0}^{N} |\exp(-at)|^2 dt \\
&= \lim_{N\to\infty} \int_{0}^{N} \exp(-2at) dt \\
&= \lim_{N\to\infty} \left[-\frac{1}{2a} \exp(-2at) \right]_0^N \\
&= \lim_{N\to\infty} \left(-\frac{1}{2a} \exp(-2aN) + \frac{1}{2a} \right)
\end{aligned}
$$

$$E = \frac{1}{2a}$$

The integral converges to a finite value, so the energy is finite.

Now, consider the power of the signal:

$$\begin{aligned} P &= \lim_{N\to\infty} \frac{1}{2N} \int_{-N}^{N} |x(t)|^2 dt \\ &= \lim_{N\to\infty} \frac{1}{2N} \frac{1}{2a} \\ P &= 0 \end{aligned}$$

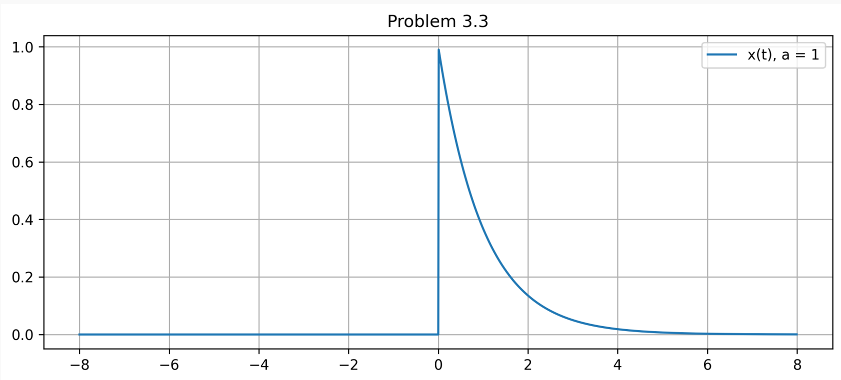
The integral converges to 0, so the power is 0.

Thus, the signal $x(t) = \exp(-at)u(t)$, $a > 0$ is $\boxed{\textbf{a energy signal}}$ with energy $\boxed{E = \frac{1}{2a}}$.

By using Python and Matplotlib, we can visualize the signal:

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

def unit_signal(t):
    return 1.0 if t >= 0 else 0.0

unit_signal_vectorize = np.vectorize(unit_signal)

fig = plt.figure(figsize=(10, 4))

t = np.arange(-8, 8, 0.01)
x = np.exp(-t) * unit_signal_vectorize(t)

plt.title("Problem 3.3")
plt.plot(t, x, label="x(t), a = 1")
plt.grid(True)
plt.legend(loc="upper right")
plt.show()
```

The plot of the signal is shown below:



d) $x(t) = A\exp(bt)u(t),\ b > 0$

Solution. Consider the energy of the signal:

$$
\begin{aligned}
E &= \lim_{N\to\infty}\int_{-N}^{N}|x(t)|^2dt \\
&= \lim_{N\to\infty}\int_{-N}^{N}|A\exp(bt)u(t)|^2dt \\
&= \lim_{N\to\infty}\int_{0}^{N}|A\exp(bt)|^2dt \\
&= \lim_{N\to\infty}\int_{0}^{N}A^2\exp(2bt)dt \\
&= \lim_{N\to\infty}A^2\left[\frac{1}{2b}\exp(2bt)\right]_0^N \\
&= \lim_{N\to\infty}A^2\left(\frac{1}{2b}\exp(2bN)-\frac{1}{2b}\right) \\
E &= \infty
\end{aligned}
$$

The integral diverges, so the energy is infinite.

Now, consider the power of the signal:

$$
\begin{aligned}
P &= \lim_{N\to\infty}\frac{1}{2N}\int_{-N}^{N}|x(t)|^2dt \\
&= \lim_{N\to\infty}\frac{1}{2N}A^2\left(\frac{1}{2b}\exp(2bN)-\frac{1}{2b}\right) \\
P &= \infty
\end{aligned}
$$

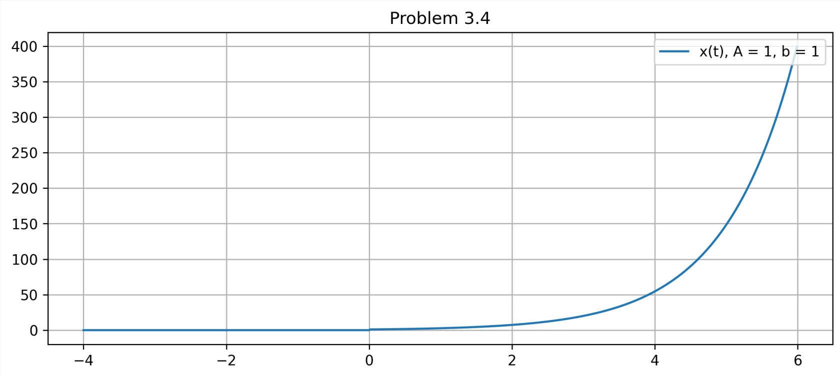
The integral diverges, so the power is infinite.

Thus, the signal $x(t) = A\exp(bt)u(t),\ b > 0$ is $\boxed{\textbf{neither a energy nor a power signal}}$.

By using Python and Matplotlib, we can visualize the signal:

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

def unit_signal(t):
    return 1.0 if t >= 0 else 0.0

unit_signal_vectorize = np.vectorize(unit_signal)

fig = plt.figure(figsize=(10, 4))

t = np.arange(-4, 6, 0.01)
x = np.exp(t) * unit_signal_vectorize(t)

plt.title("Problem 3.4")
plt.plot(t, x, label="x(t), A = 1, b = 1")
plt.grid(True)
plt.legend(loc="upper right")
plt.show()
```

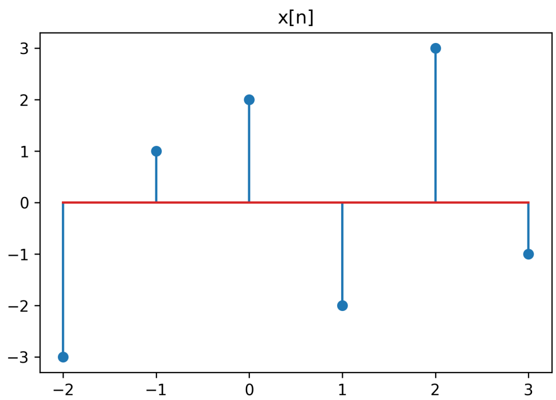
The plot of the signal is shown below:



Problem 4. For the discrete time signal $x[n]$ shown in Figure below, sketch each of the following

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

fig = plt.figure(figsize=(6, 4))

t = np.arange(-2, 4)
x_t = np.array([-3, 1, 2, -2, 3, -1])

plt.stem(t, x_t)
plt.title('x[n]')
plt.show()
```

With the resulting plot shown below:



Solution. By using Python, we can create a function to transform the signal based on the given transformation function:

```python
def transform_signal(x, n, f):
    """Return x[f(n)] for any discrete-time signal x[n]."""
    f_n = f(n)
    f_n_int = f_n[np.floor(f_n) == f_n]

    x_new = np.zeros(f_n_int.shape[0], dtype=float)

    idx = 0
    for i, val in enumerate(f_n):
        if int(val) == val:
            x_new[idx] = x[i]
            idx += 1

    return x_new, f_n_int
```

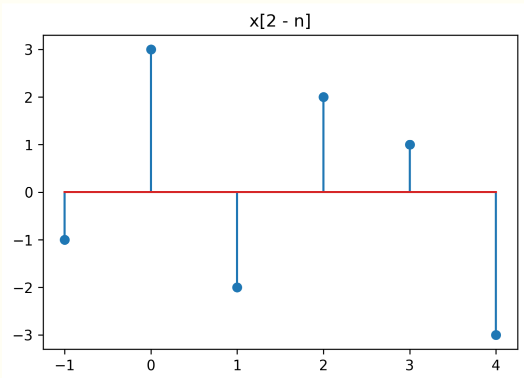
TO SUBMIT

a) $x[2-n]$

Solution. Using Python and Matplotlib to plot the signal $x[2-n]$:

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

fig = plt.figure(figsize=(6, 4))

t = np.arange(-2, 4)
x_t = np.array([-3, 1, 2, -2, 3, -1])

x_t, t = transform_signal(x_t, t, lambda x: 2 - x)

plt.stem(t, x_t)
plt.title("x[2 - n]")
plt.show()
```

With the resulting plot shown below:

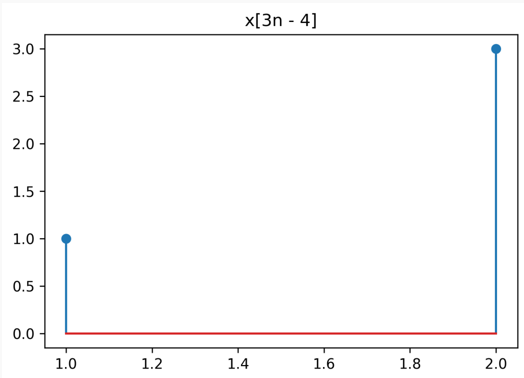


b) $x[3n-4]$

Solution. Using Python and Matplotlib to plot the signal $x[3n-4]$:

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

fig = plt.figure(figsize=(6, 4))

t = np.arange(-2, 4)
x_t = np.array([-3, 1, 2, -2, 3, -1])

x_t, t = transform_signal(x_t, t, lambda x: (x + 4) / 3)

plt.stem(t, x_t)
plt.title("x[3n - 4]")
plt.show()
```

With the resulting plot shown below:



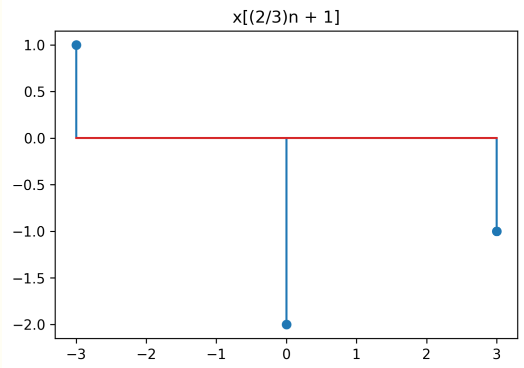
TO SUBMIT

c) $x\left[\frac{2}{3}n+1\right]$

Solution. Using Python and Matplotlib to plot the signal $x[\frac{2}{3}n+1]$:

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

fig = plt.figure(figsize=(6, 4))

t = np.arange(-2, 4)
x_t = np.array([-3, 1, 2, -2, 3, -1])

x_t, t = transform_signal(x_t, t, lambda x: (x - 1) * 3 /
    2)

plt.stem(t, x_t)
plt.title("x[(2/3)n + 1]")
plt.show()
```

With the resulting plot shown below:

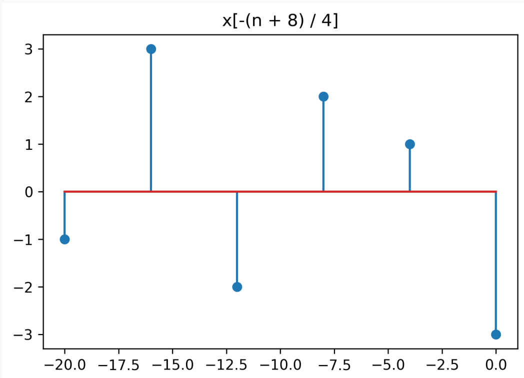


d) $x\left[-\frac{n+8}{4}\right]$

Solution. Using Python and Matplotlib to plot the signal $x\left[-\frac{n+8}{4}\right]$:

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

fig = plt.figure(figsize=(6, 4))

t = np.arange(-2, 4)
x_t = np.array([-3, 1, 2, -2, 3, -1])

x_t, t = transform_signal(x_t, t, lambda x: (-4 * x) - 8)

plt.stem(t, x_t)
plt.title("x[-(n + 8) / 4]")
plt.show()
```

With the resulting plot shown below:



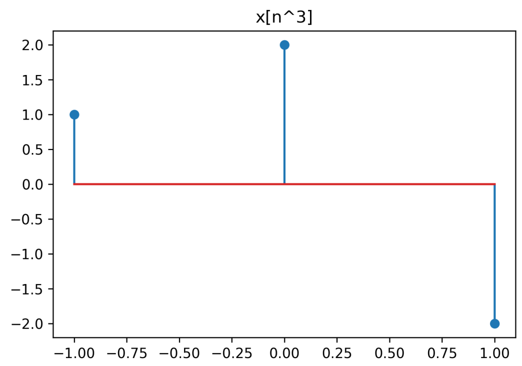
TO SUBMIT

e) $x[n^3]$

Solution. Using Python and Matplotlib to plot the signal $x[n^3]$:

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

fig = plt.figure(figsize=(6, 4))

t = np.arange(-2, 4)
x_t = np.array([-3, 1, 2, -2, 3, -1])

x_t, t = transform_signal(x_t, t, lambda x: np.cbrt(x))

plt.stem(t, x_t)
plt.title("x[n^3]")
plt.show()
```

With the resulting plot shown below:



f) $x[2-n] + x[3n-4]$

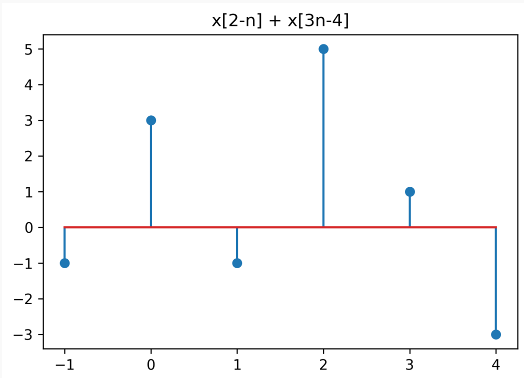
Solution. Introduce a helper function to add two signals:

```python
def add_discrete_signals(x1, t1, x2, t2):
    """Return x1[n] + x2[n] for any discrete-time signals
     x1[n] and x2[n]."""
    t = np.union1d(t1, t2)
    x = np.zeros(t.shape[0], dtype=float)

    for i, val in enumerate(t):
        if val in t1:
            x[i] += x1[np.where(t1 == val)[0][0]]
        if val in t2:
            x[i] += x2[np.where(t2 == val)[0][0]]

    return x, t
```

Using Python and Matplotlib to plot the signal $x[n^3]$:

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

fig = plt.figure(figsize=(6, 4))

t = np.arange(-2, 4)
x_t = np.array([-3, 1, 2, -2, 3, -1])

x_t_1, t_1 = transform_signal(x_t, t, lambda x: 2 - x)
x_t_2, t_2 = transform_signal(x_t, t, lambda x: (x + 4) /
    3)

x_t, t = add_discrete_signals(x_t_1, t_1, x_t_2, t_2)

plt.stem(t, x_t)
plt.title("x[2-n] + x[3n-4]")
plt.show()
```

With the resulting plot shown below:



Problem 5. Determine whether each of following signals is periodic, and if so, find its period.

a) $x[n] = \sin\left(\frac{\pi n}{4} + \frac{\pi}{8}\right)$

Solution. Consider the signal:

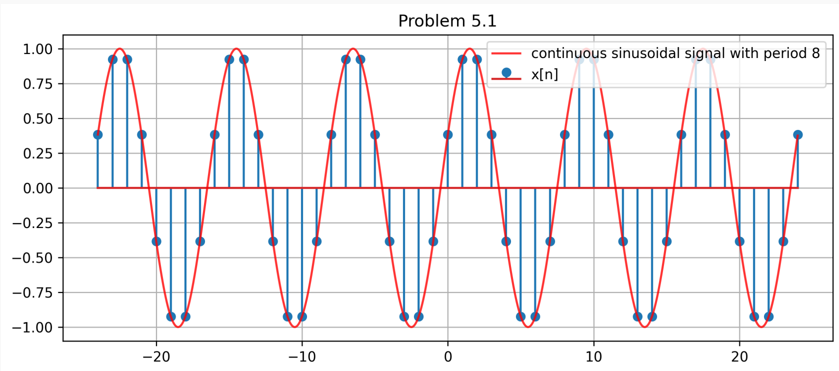
$$\sin\left(\frac{\pi n}{4} + \frac{\pi}{8}\right) \text{ has a period of } N = \frac{2\pi}{\frac{\pi}{4}} = 8$$

Thus, the signal $x[n] = \sin\left(\frac{\pi n}{4} + \frac{\pi}{8}\right)$ $\boxed{\textbf{is periodic}}$ with a period of $\boxed{N = 8}$.

By using Python and Matplotlib, we can visualize the periodicity of the signal:

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

fig = plt.figure(figsize=(10, 4))

t = np.arange(-24, 25, 1)
x = np.sin(np.pi/4 * t + np.pi/8)

tc = np.arange(-24, 24, 0.01)
xc = np.sin(np.pi/4 * tc + np.pi/8)

plt.title("Problem 5.1")
plt.stem(t, x, label="x[n]")
plt.plot(tc, xc, "r", alpha=0.8, label="continuous
    sinusoidal signal with period 8")
plt.grid(True)
plt.legend(loc="upper right")
plt.show()
```

The plot of the signal is shown below:



b) $x[n] = \sin\left(\frac{3\pi n}{4}\right) + \sin\left(\frac{\pi}{3}n\right)$

Solution. Consider the signal:

$$\sin\left(\frac{3\pi n}{4}\right) \text{ has a period of } N_1 = \frac{2\pi}{\frac{3\pi}{4}} = \frac{8}{3}$$

$$\sin\left(\frac{\pi n}{3}\right) \text{ has a period of } N_2 = \frac{2\pi}{\frac{\pi}{3}} = 6$$

Considering the least common multiple of the two periods:

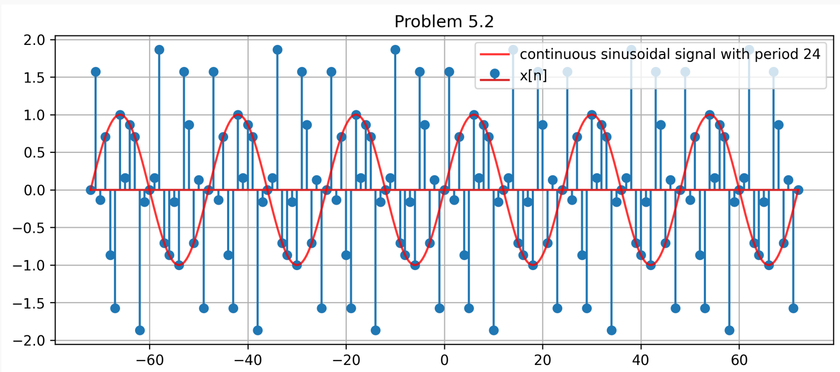
$$N = \operatorname{lcm}(N_1, N_2) = \operatorname{lcm}\left(\frac{8}{3}, 6\right) = \operatorname{lcm}\left(\frac{8}{3}, \frac{18}{3}\right) = \frac{72}{3} = 24$$

Thus, the signal $x[n] = \sin\left(\frac{3\pi n}{4}\right) + \sin\left(\frac{\pi}{3}n\right)$ $\boxed{\textbf{is periodic}}$ with a period of $\boxed{N = 24}$.

By using Python and Matplotlib, we can visualize the periodicity of the signal:

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

fig = plt.figure(figsize=(10, 4))

t = np.arange(-72, 73, 1)
x = np.sin(3 * np.pi/4 * t) + np.sin(np.pi/3 * t)

tc = np.arange(-72, 72, 0.01)
xc = np.sin(np.pi/12 * tc)

plt.title("Problem 5.2")
plt.stem(t, x, label="x[n]")
plt.plot(tc, xc, "r", alpha=0.8, label="continuous sinusoidal signal with period 24")
plt.grid(True)
plt.legend(loc="upper right")
plt.show()
```

The plot of the signal is shown below:



c) $x[n] = \sin\left(\frac{3\pi n}{4}\right)\sin\left(\frac{\pi}{3}n\right)$

Solution. Using the product-to-sum identities, we can rewrite the signal as:

$$x[n] = \sin\left(\frac{3\pi n}{4}\right)\sin\left(\frac{\pi}{3}n\right) = \frac{1}{2}\left[\cos\left(\frac{3\pi n}{4} - \frac{\pi n}{3}\right) - \cos\left(\frac{3\pi n}{4} + \frac{\pi n}{3}\right)\right]$$

Consider the signal:

$$\cos\left(\frac{3\pi n}{4} - \frac{\pi n}{3}\right) = \cos\left(\frac{5\pi n}{12}\right) \text{ has a period of } N_1 = \frac{2\pi}{\frac{5\pi}{12}} = \frac{24}{5}$$

$$\cos\left(\frac{3\pi n}{4} + \frac{\pi n}{3}\right) = \cos\left(\frac{13\pi n}{12}\right) \text{ has a period of } N_2 = \frac{2\pi}{\frac{13\pi}{12}} = \frac{24}{13}$$

Considering the least common multiple of the two periods:

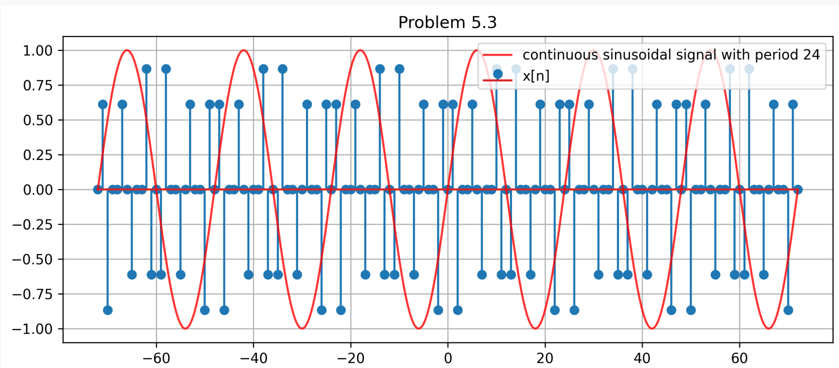
$$N = \operatorname{lcm}(N_1, N_2) = \operatorname{lcm}\left(\frac{24}{5}, \frac{24}{13}\right) = \frac{24}{\gcd(5, 13)} = 24$$

Thus, the signal $x[n] = \sin\left(\frac{3\pi n}{4}\right)\sin\left(\frac{\pi}{3}n\right)$ $\boxed{\textbf{is periodic}}$ with a period of $\boxed{N = 24}$.

By using Python and Matplotlib, we can visualize the periodicity of the signal:

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

fig = plt.figure(figsize=(10, 4))

t = np.arange(-72, 73, 1)
x = np.sin(3 * np.pi/4 * t) * np.sin(np.pi/3 * t)

tc = np.arange(-72, 72, 0.01)
xc = np.sin(np.pi/12 * tc)

plt.title("Problem 5.3")
plt.stem(t, x, label="x[n]")
plt.plot(tc, xc, "r", alpha=0.8, label="continuous
    sinusoidal signal with period 24")
plt.grid(True)
plt.legend(loc="upper right")
plt.show()
```

The plot of the signal is shown below:



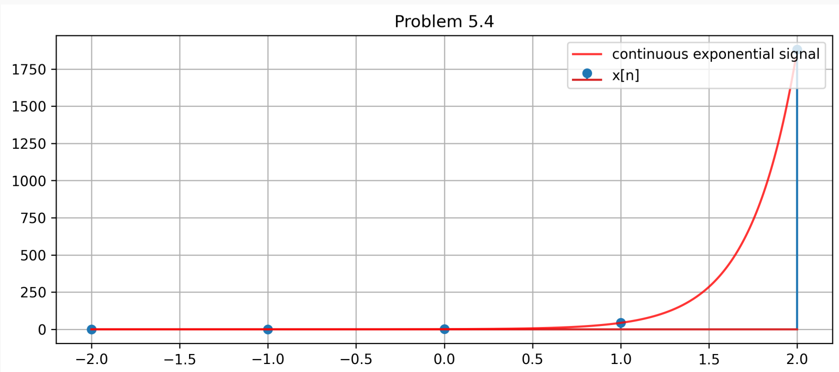
d) $x[n] = \exp\left(\frac{6\pi}{5}n\right)$

Solution.
Consider the signal, this is a exponential signal, which have not imaginary part, thus it $\boxed{\textbf{is not periodic}}$.

By using Python and Matplotlib, we can visualize the periodicity of the signal:

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

fig = plt.figure(figsize=(10, 4))

t = np.arange(-2, 3, 1)
x = np.exp(6 * np.pi/5 * t)

tc = np.arange(-2, 2, 0.01)
xc = np.exp(6 * np.pi/5 * tc)

plt.title("Problem 5.4")
plt.stem(t, x, label="x[n]")
plt.plot(tc, xc, "r", alpha=0.8, label="continuous
    exponential signal")
plt.grid(True)
plt.legend(loc="upper right")
plt.show()
```

The plot of the signal is shown below:



e) $x[n] = \exp\left(j\frac{5\pi}{6}n\right)$

Solution. Consider the signal:

$$\exp\left(j\frac{5\pi}{6}n\right) \text{ has a period of } N = \frac{2\pi}{\frac{5\pi}{6}} = \frac{12}{5}$$

We need to make the period an integer, thus we can multiply the period by 5:

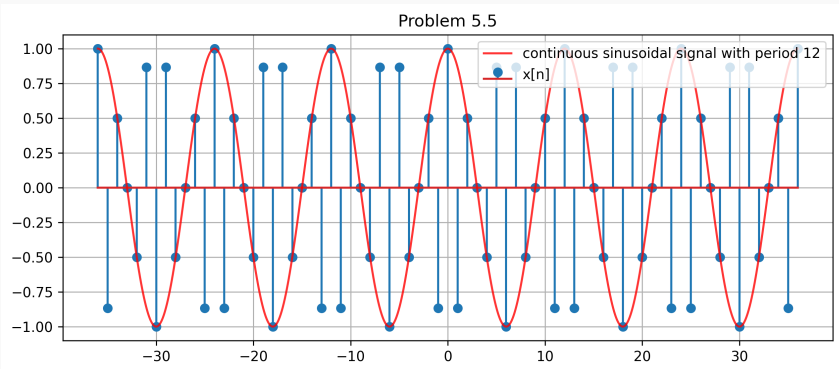
$$N = 5 \cdot \frac{12}{5} = 12$$

Thus, the signal $x[n] = \exp\left(j\frac{5\pi}{6}n\right)$ $\boxed{\textbf{is periodic}}$ with a period of $\boxed{N = 12}$.

By using Python and Matplotlib, we can visualize the periodicity of the signal:

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

fig = plt.figure(figsize=(10, 4))

t = np.arange(-36, 37, 1)
x = np.exp(1j * 5 * np.pi/6 * t)

tc = np.arange(-36, 36, 0.01)
xc = np.exp(1j * np.pi/6 * tc)

plt.title("Problem 5.5")
plt.stem(t, x, label="x[n]")
plt.plot(tc, xc, "r", alpha=0.8, label="continuous
    sinusoidal signal with period 12")
plt.grid(True)
plt.legend(loc="upper right")
plt.show()
```

The plot of the signal is shown below:



f) $x[n] = \sum_{m=-\infty}^{\infty} [\delta[n-2m] + 2\delta[n-3m]]$

Solution. Consider the signal, using the properties of the delta function:

1. The first term $\sum_{m=-\infty}^{\infty} \delta[n-2m]$ is a periodic signal with period of 2, since:

$$\sum_{m=-\infty}^{\infty} \delta[n-2m] = \sum_{m=-\infty}^{\infty} \delta[n+2-2m]$$

2. The second term $2\sum_{m=-\infty}^{\infty} \delta[n-3m]$ is a periodic signal with period of 3, since:

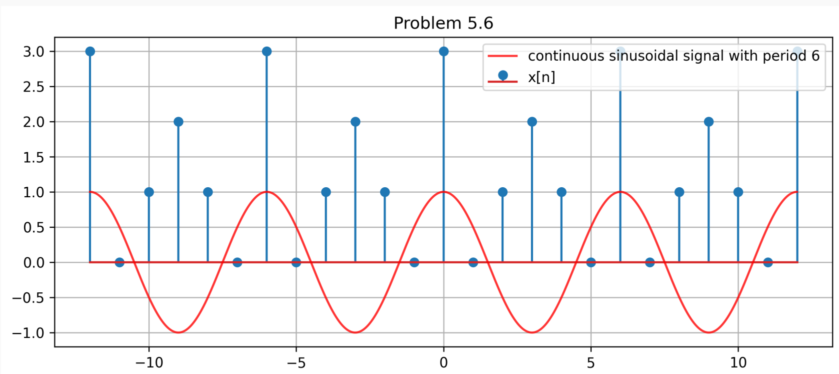
$$2\sum_{m=-\infty}^{\infty} \delta[n-3m] = 2\sum_{m=-\infty}^{\infty} \delta[n+3-3m]$$

Thus, the overall signal $x[n]$ $\boxed{\textbf{is periodic}}$ with a fundamental period of $\text{lcm}(2,3) = \boxed{6}$.

By using Python and Matplotlib, we can visualize the periodicity of the signal:

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

fig = plt.figure(figsize=(10, 4))

t = np.arange(-12, 13, 1)
x1 = np.isin(t \% 2, [0]).astype(float) # delta[n-2k]
x2 = np.isin(t \% 3, [0]).astype(float) # delta[n-3k]

x = x1 + 2 * x2

tc = np.arange(-12, 12, 0.01)
xc = np.exp(1j * np.pi/3 * tc)

plt.title("Problem 5.6")
plt.stem(t, x, label="x[n]")
plt.plot(tc, xc, "r", alpha=0.8, label="continuous
    sinusoidal signal with period 6")
plt.grid(True)
plt.legend(loc="upper right")
plt.show()
```

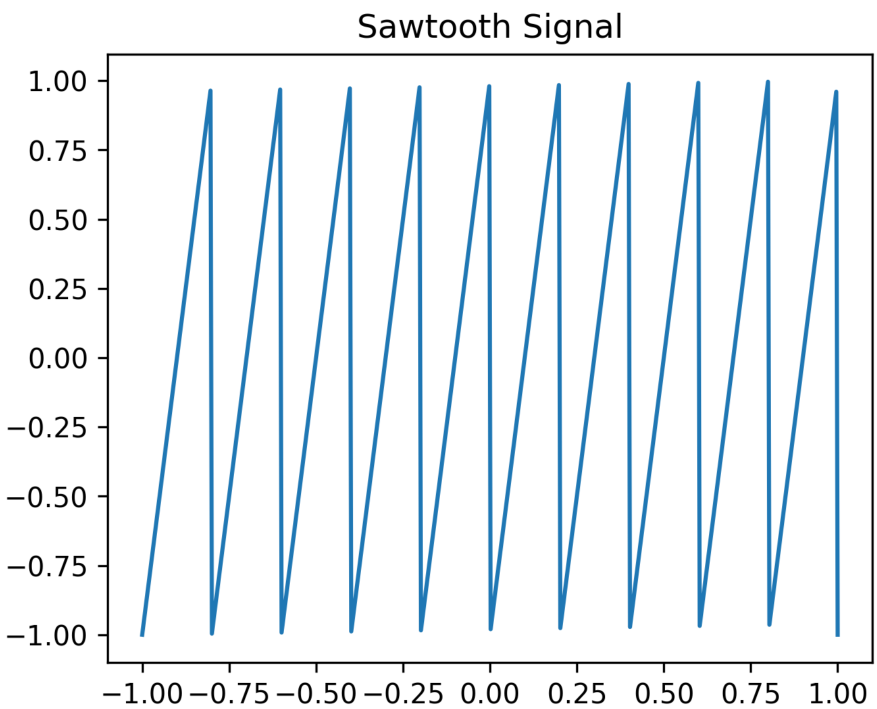
The plot of the signal is shown below:



Problem 6. Signal transformations: Study the sawtooth function in the figure below. Apply reflection, scaling, shifting operations to the signal and plot the transformed signals compared with the original sawtooth signal.

```python
import numpy as np
from scipy import signal

fig = plt.figure(figsize=(5, 4))

t = np.linspace(-1, 1, 500)
sawtooth = signal.sawtooth(2 * np.pi * 5 * t)

plt.title("Sawtooth Signal")
plt.plot(t, sawtooth)
plt.show()
```

The plot of the signal is shown below:

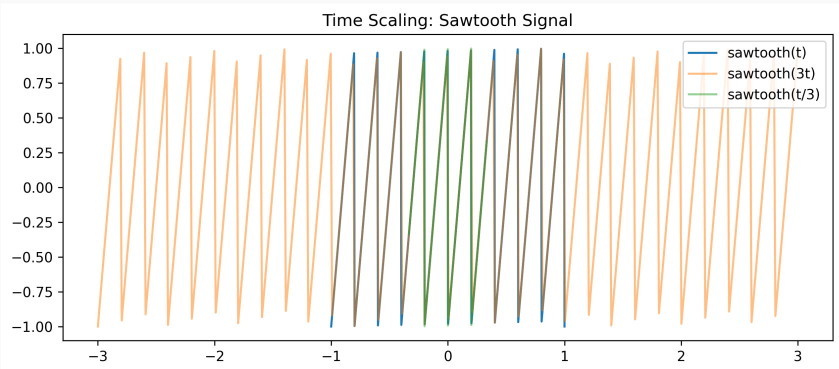


a) time scaling: scaling factor = 3 and 1/3

Solution. Using Python and Matplotlib to plot the time-scaled signals:

```python
import matplotlib.pyplot as plt
import numpy as np
from scipy import signal

fig = plt.figure(figsize=(10, 4))

t = np.linspace(-1, 1, 500)
sawtooth = signal.sawtooth(2 * np.pi * 5 * t)

# scaling factor = 3 and 1/3
## TODO : writing code for time scaling

t1 = t * 3
sawtooth1 = signal.sawtooth(2 * np.pi * 5 * t1)

t2 = t * (1/3)
sawtooth2 = signal.sawtooth(2 * np.pi * 5 * t2)

plt.title("Time Scaling: Sawtooth Signal")
plt.plot(t, sawtooth, label="sawtooth(t)", alpha=1)
plt.plot(t1, sawtooth1, label="sawtooth(3t)", alpha=0.5)
plt.plot(t2, sawtooth2, label="sawtooth(t/3)", alpha=0.5)
plt.legend(loc="upper right")
plt.show()
```

With the resulting plot shown below:

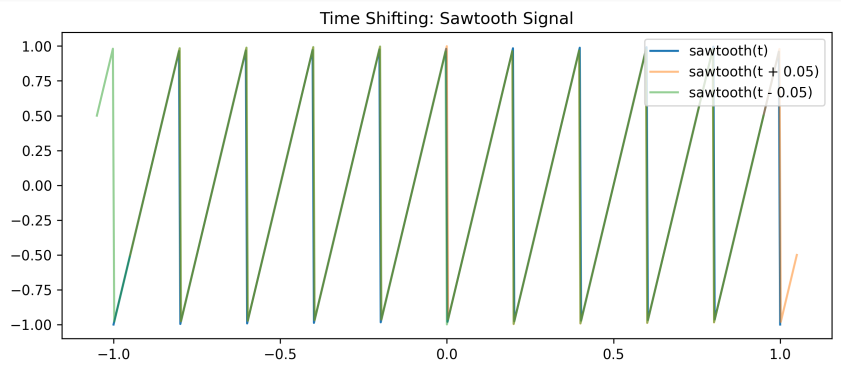


b) time shifting: shifting amount = ± 0.05

Solution. Using Python and Matplotlib to plot the time-shifted signals:

```python
import matplotlib.pyplot as plt
import numpy as np
from scipy import signal

fig = plt.figure(figsize=(10, 4))

t = np.linspace(-1, 1, 500)
sawtooth = signal.sawtooth(2 * np.pi * 5 * t)

# scaling factor = 3 and 1/3
## TODO : writing code for time scaling

t1 = t + 0.05
sawtooth1 = signal.sawtooth(2 * np.pi * 5 * t1)

t2 = t - 0.05
sawtooth2 = signal.sawtooth(2 * np.pi * 5 * t2)

plt.title("Time Shifting: Sawtooth Signal")
plt.plot(t, sawtooth, label="sawtooth(t)", alpha=1)
plt.plot(t1, sawtooth1, label="sawtooth(t + 0.05)", alpha
    =0.5)
plt.plot(t2, sawtooth2, label="sawtooth(t - 0.05)", alpha
    =0.5)
plt.legend(loc="upper right")
plt.show()
```

With the resulting plot shown below:

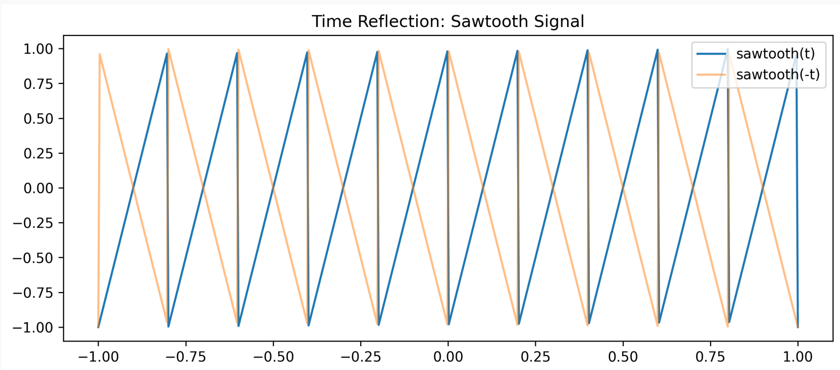


c) time reflection: reflecting over the y-axis

Solution. Using Python and Matplotlib to plot the time-reflected signals:

```python
import matplotlib.pyplot as plt
import numpy as np
from scipy import signal

fig = plt.figure(figsize=(10, 4))

t = np.linspace(-1, 1, 500)
sawtooth = signal.sawtooth(2 * np.pi * 5 * t)

## TODO : writing code for time Reflection

t1 = t
sawtooth1 = signal.sawtooth(2 * np.pi * 5 * (-1 * t))

plt.title("Time Reflection: Sawtooth Signal")
plt.plot(t, sawtooth, label="sawtooth(t)", alpha=1)
plt.plot(t1, sawtooth1, label="sawtooth(-t)", alpha=0.5)
plt.legend(loc="upper right")

plt.savefig("../images/problem_6_3.png", dpi=300,
    bbox_inches="tight")

plt.show()
```

With the resulting plot shown below:



Problem 9. Evaluate the following integrals

a) $\int_{-\infty}^{\infty} \left(\frac{2}{3}t - \frac{3}{2}\right) \delta(t-1)\, dt$

TO SUBMIT

Solution. Using the sifting property of the delta function, we have:

$$\begin{aligned}\int_{-\infty}^{\infty} \left(\frac{2}{3}t - \frac{3}{2}\right) \delta(t-1)\, dt &= \left(\frac{2}{3}(1) - \frac{3}{2}\right)\\ &= \frac{2}{3} - \frac{3}{2}\\ \int_{-\infty}^{\infty} \left(\frac{2}{3}t - \frac{3}{2}\right) \delta(t-1)\, dt &= \boxed{-\frac{5}{6}}\end{aligned}$$

b) $\int_{-\infty}^{\infty} (t-1)\delta\left(\frac{2}{3}t - \frac{3}{2}\right)\, dt$

Solution. Using the sifting property of the delta function, we have:

$$\begin{aligned}\int_{-\infty}^{\infty} (t-1)\delta\left(\frac{2}{3}t - \frac{3}{2}\right)\, dt &= \int_{-\infty}^{\infty} (t-1)\delta\left(t - \frac{9}{4}\right) \cdot \left|\frac{d}{dt}\left(\frac{2}{3}t - \frac{3}{2}\right)\right|^{-1}\, dt\\ &= \int_{-\infty}^{\infty} (t-1)\delta\left(t - \frac{9}{4}\right) \cdot \frac{3}{2} dt\\ &= (\frac{9}{4} - 1) \cdot \frac{3}{2}\\ &= \frac{5}{4} \cdot \frac{3}{2} = \frac{15}{8}\\ \int_{-\infty}^{\infty} (t-1)\delta\left(\frac{2}{3}t - \frac{3}{2}\right)\, dt &= \boxed{\frac{15}{8}}\end{aligned}$$

c) $\int_{-3}^{-2} \left[e^{(-t+1)} + \sin\left(\frac{2\pi t}{3}\right)\right] \delta\left(t - \frac{3}{2}\right)\, dt$

TO SUBMIT

Solution. Because the argument of the delta function $t - \frac{3}{2}$ has its root at $t = \frac{3}{2}$, which is outside the integration limits of -3 to -2, the integral evaluates to zero:

$$\int_{-3}^{-2} \left[e^{(-t+1)} + \sin\left(\frac{2\pi t}{3}\right)\right] \delta\left(t - \frac{3}{2}\right)\, dt = \boxed{0}$$

d) $\int_{-3}^{2} \left[e^{(-t+1)} + \sin\left(\frac{2\pi t}{3}\right)\right] \delta\left(t - \frac{3}{2}\right)\, dt$

Solution. Using the sifting property of the delta function, we have:

$$\begin{aligned}\int_{-3}^{2} \left[e^{(-t+1)} + \sin\left(\frac{2\pi t}{3}\right)\right] \delta\left(t - \frac{3}{2}\right)\, dt &= \left[e^{(-\frac{3}{2}+1)} + \sin\left(\frac{2\pi(\frac{3}{2})}{3}\right)\right]\\ &= e^{-\frac{1}{2}} + \sin(\pi)\\ &= e^{-\frac{1}{2}} + 0\end{aligned}$$

$$\int_{-3}^{2} \left[e^{(-t+1)} + \sin\left(\frac{2\pi t}{3}\right) \right] \delta\left(t - \frac{3}{2}\right) dt = \boxed{e^{-\frac{1}{2}}}$$