CENG 384 - Signals and Systems for Computer Engineers
Spring 2023
Homework 1

Şanlı, Enes
e2375749@ceng.metu.edu.tr

Meyvecioğlu, Hasan Ege
e2449783@ceng.metu.edu.tr

April 2, 2023

---

1. (a) z = x +yj
   $\bar{z}$ = x -yj . Then we can write second equation
   2x + 5 + 2yj = -x +(y+1)j
   y = 1 and x = -5/3
   $|z| = \sqrt{1^2 + \frac{-5}{3}^2} = \frac{\sqrt{34}}{3}$
   $|z|^2 = |z| * |z| = \frac{34}{9}$

   (b) z = r$e^{j\theta} = r * (cos\theta + j * sin\theta)$
   $z^5 = r^5 * (cos5\theta + j * sin5\theta) = 32j$ so r = 2 and $\theta = \frac{\pi}{10}$
   $z = 2 * (cos\frac{\pi}{10} + j * sin\frac{\pi}{10})$

   (c) z = $\frac{(-1-j)*(1+j)*(\frac{1}{2}+\frac{\sqrt{3}j}{2})}{(-1-j)*(-1+j)}$
   z = $\frac{(-2j)*(\frac{1}{2}+\frac{\sqrt{3}j}{2})}{2}$
   z = $\frac{-1}{2}j - \frac{\sqrt{3}}{2}j^2$
   z = $\frac{\sqrt{3}}{2} - \frac{1}{2}j$
   $|z| = \sqrt{\frac{3}{4} + \frac{1}{4}} = 1$
   $\angle z = arctan(\frac{\frac{-1}{2}}{\frac{\sqrt{3}}{2}}) = arctan(\frac{-1}{\sqrt{3}}) = \frac{-\pi}{6}$

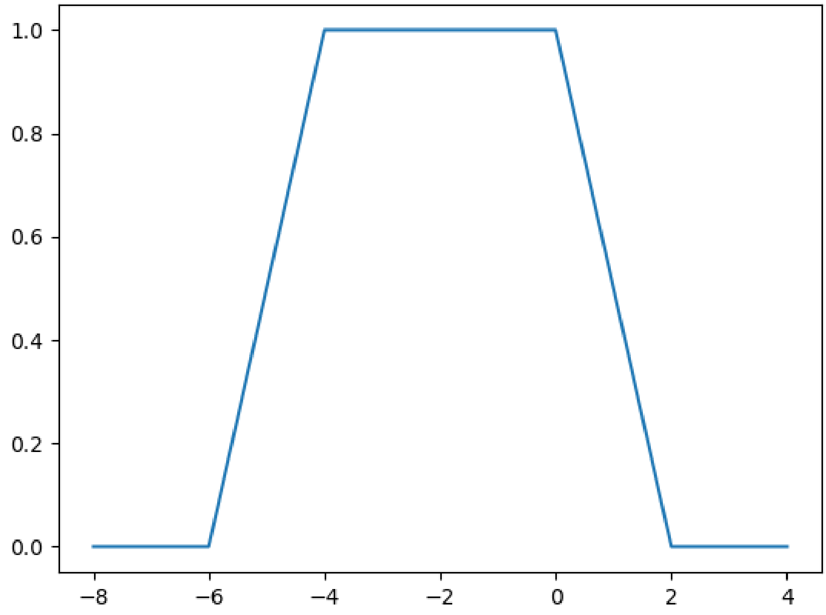
   (d) z = $j * (cos(\frac{\pi}{2}) - sin(\frac{\pi}{2}) * j)$
   z = $0 - j^2 * 1$
   z = 1

2.

$$x(t) = \left\{ \begin{array}{ll} 0, & -3 < t < -2 \\ t+2, & -2 < t < -1 \\ 1, & -1 < t < 1 \\ 2-t, & 1 < t < 2 \\ 0, & 2 < t < 3 \end{array} \right\} \quad (1)$$

We should put $\frac{t}{2} + 1$ instead of t, then

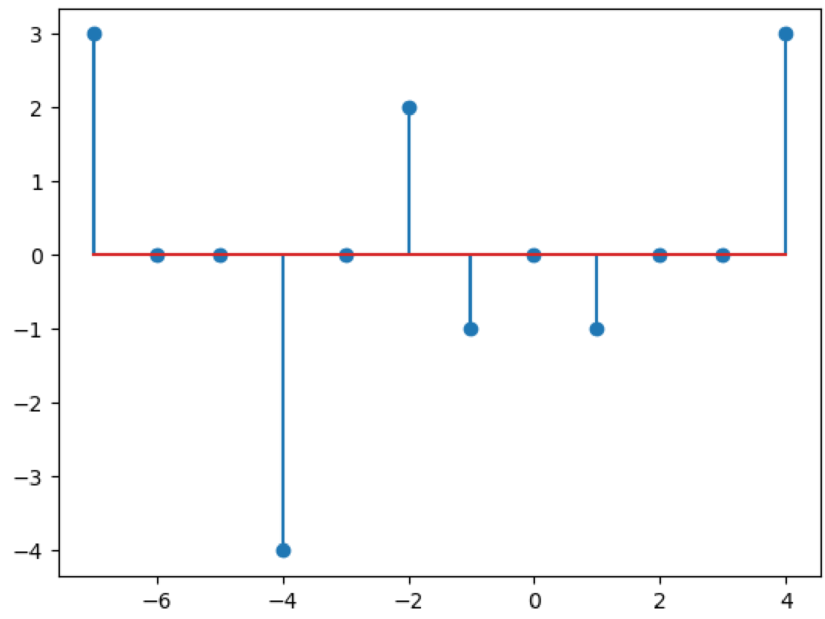
$$x(t) = \left\{ \begin{array}{ll} 0, & -8 < t < -6 \\ \frac{t}{2}+3, & -6 < t < -4 \\ 1, & -4 < t < 0 \\ 1-\frac{t}{2}, & 0 < t < 2 \\ 0, & 2 < t < 4 \end{array} \right\} \quad (2)$$



3. (a) In this question we are doing time compression and time reversal.
For x[-n] we get the symmetry about the y-axis, because it is time reversal. So
x[-1] = -1
x[-2] = 2
x[-4] = -4
x[-7] = 3

We do time compression for the x[2n-1] function. And the new interval of this function is
2n - 1 = -1
n = 0
2n - 1 = 8
n = 9/2 but this function discrete time so we take n = 4
So our new x[2n-1] function take some values in this ([0,1,2,3,4]) interval. But the other n's will be 0.

For n = 0 x[2n-1] equals to x[-1] = 0
For n = 1 x[2n-1] equals to x[1] = -1
For n = 2 x[2n-1] equals to x[3] = 0
For n = 3 x[2n-1] equals to x[5] = 0
For n = 4 x[2n-1] equals to x[7] = 3
So x[-n] + x[2n-1] is



(b) For x=-7 we can use $3\delta(x+7)$
For x=-4 we can use $-4\delta(x+4)$
For x=-2 we can use $2\delta(x+2)$
For x=-1 we can use $-1\delta(x+1)$
For x=1 we can use $-1\delta(x-1)$
For x=4 we can use $3\delta(x-4)$
So function = $3\delta(x+7) - 4\delta(x+4) + 2\delta(x+2) - 1\delta(x+1) - 1\delta(x-1) + 3\delta(x-4)$

4. (a) If this function is periodic then, x(t) = x(t + T_0)
so $5 * sin(3t + \frac{-\pi}{4}) = 5 * sin(3t + 3T_0 + \frac{-\pi}{4})$
we know that sin fundamental period is 2π. Then,
$3t + \frac{-\pi}{4} + k * 2\pi = 3t + 3T_0 + \frac{-\pi}{4}$ When k $\in Z$
k*$2\pi = 3T_0$
when k = 1 period = $T_0 = \frac{2\pi}{3}$ and it is periodic

(b) If this function is periodic then, x[n] = x[n + N_0]
so $cos(\frac{13\pi}{10}n) + sin(\frac{7\pi}{10}n) = cos(\frac{13\pi}{10}(n + N_0)) + sin(\frac{7\pi}{10}(n + N_0))$
we know that sin and cos fundamental period is 2π. Then,
For cos, $\frac{13\pi}{10}n + k * 2\pi = \frac{13\pi}{10}n + \frac{13\pi}{10}N_0$ When k $\in Z$
$\frac{20k}{13} = N_0$ so when k be a multiple of 13 N_0 must be a multiple of 20
For sin, $\frac{7\pi}{10}n + k * 2\pi = \frac{7\pi}{10}n + \frac{7\pi}{10}N_0$ When k $\in Z$
$\frac{20k}{7} = N_0$ so when k be a multiple of 7 N_0 must be a multiple of 20
So LCM(20,20) = 20. Because of that this function periodic
$N_0 = 20$

(c) If this function is periodic then, x[n] = x[n + N_0]
so $cos(7n - 5) = cos(7(n + N_0) - 5)$
we know that sin fundamental period is 2π. Then,
$7n - 5 + k * 2\pi = 7t + 7N_0 - 5$ When k $\in Z$
k*$2\pi = 7N_0$
Then, $N_0 = \frac{k*2\pi}{7}$,
But, there is a problem. In this equation N_0 must be a irrational number and our function is discrete time. So $N_0 \notin Z$, because of that this function is not periodic.

5. (a) The signal $x(t)$ can be expressed in terms of unit step functions. Its magnitude changes at times $t = 1$, $t = 3$ and $t = 4$ respectively. We can formulate it as:

$$x(t) = u(t-1) - 3 * u(t-3) + u(t-4)$$

(b) Recall that the derivative of the unit step function is the unit impulse function:

$$\delta(t) = \frac{\mathrm{d}u(t)}{\mathrm{d}t}$$

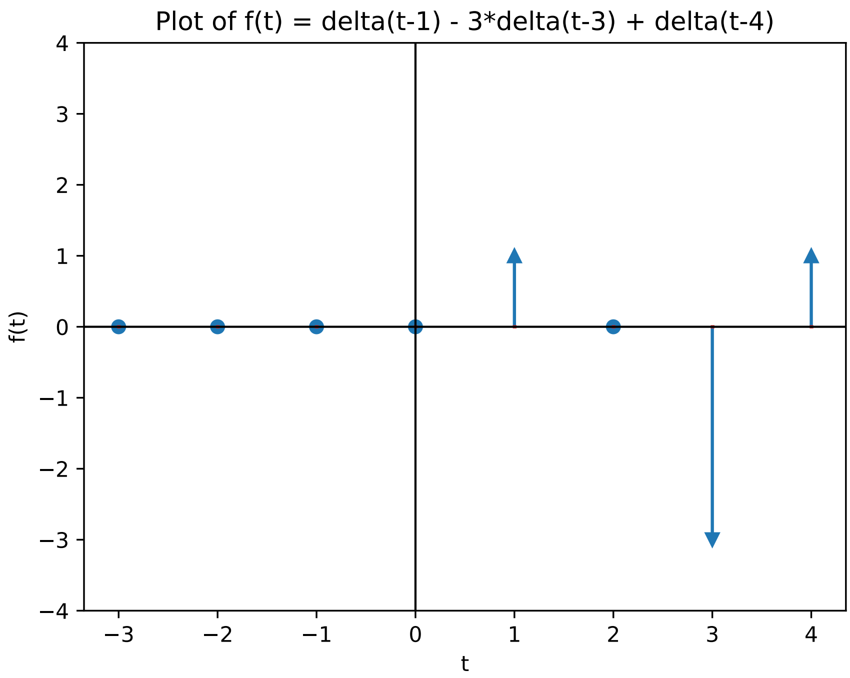
With this, we can find $\frac{\mathrm{d}x(t)}{\mathrm{d}t}$:

$$\frac{\mathrm{d}x(t)}{\mathrm{d}t} = \frac{\mathrm{d}}{\mathrm{d}t}(u(t-1) - 3 * u(t-3) + u(t-4))$$

$$= \frac{\mathrm{d}(u(t-1)}{\mathrm{d}t} - 3 * \frac{\mathrm{d}u(t-3)}{\mathrm{d}t} + \frac{\mathrm{d}u(t-4)}{\mathrm{d}t}$$

$$= \delta(t-1) - 3 * \delta(t-3) + \delta(t-4)$$

Hence, we can draw it as:



6. (a)

$$y(t) = t * x(2t+3)$$

- **With Memory and Not Causal:**
  It can be seen that the output value of y depends on the future value of input signal x. For instance, when $t = 1$, the output value depends on the future value of the input, $y(1) = 1*x(5)$. Hence it is neither memoriless nor causal.
- **Unstable:**
  Let input signal x be bounded by a real number B so that $x \leq B\ \forall x$. Then as time t goes to infinity, we can look a bound for the output:

$$\lim_{t\to\infty} y(t) = \lim_{t\to\infty} t * x(t) <= \lim_{t\to\infty} t * b = \infty$$

  As can be seen, the output is not bounded as its value grows continously with increasing t. Thus, it is not stable.
- **Linear:**
  We can check its linearity by checking whether it is obeying the superposition rule:

$$y_1 = t * x_1$$

$$y_2 = t * x_2$$

$$x = a_1 * x_1 + a_2 * x_2$$

$$y(t) = a_1 * y_1 + a_2 * y_2 = t * (a_1 * x_1 + a_2 * x_2) = a_1 * t * x_1 + a_2 * t * x_2$$

  It obeys the superposition law, therefore it is linear.

- **Non-Invertible:** If we try to take the inverse of the system, we would get:

$$y(t) = \frac{x(\frac{t-3}{2}}{\frac{t-3}{2}}$$

However, when $t = 3$, the equation is undefined hence we lose a point from the original signal. Therefore it is not invertible.
- **Time Varying:** If we shift the time t to be $t - t_0$ for the input signal, the output would be:

$$y(t) = t * (2(t - t_0) + 3) = t * (2t - 2t_0 + 3)$$

But the same shift on the output produces:

$$y(t - t_0) = (t - t_0) * (2(t - t_0) + 3) = (t - t_0) * (2t - 2t_0 + 3)$$

Since they are not equal, the system is time-varying.

(b)

$$y[n] = \sum_{k=1}^{\infty} x[n-k]$$

- **With Memory and Causal:** We can expand the summation as:

$$y[n] = \sum_{k=1}^{\infty} x[n-k] = \lim_{k\to\infty} (x[n-1] + x[n-2] + x[n-3] + ... + x[n-k])$$

It can be seen that the output of the system depends on only the past values of the input. Therefore, it has memory and it is causal.
- **Unstable:**
Let $x[n]$ be bounded by a real number B such that $x[n] \leq B$. Then,

$$y[n] = \sum_{k=1}^{\infty} x[n-k] \leq \sum_{k=1}^{\infty} b$$

Since the resulting summation does not converge, it is unbounded and unstable.
- **Linear:**
We can check its linearity by checking whether it is obeying the superposition rule:

$$y_1 = \sum_{k=1}^{\infty} x_1$$

$$y_2 = \sum_{k=1}^{\infty} x_2$$

$$x = a_1 * x_1 + a_2 * x_2$$

$$y = \sum_{k=1}^{\infty} x = \sum_{k=1}^{\infty} a_1 * x_1 + a_2 * x_2$$

$$y = a_1 * \sum_{k=1}^{\infty} x_1 + a_2 * \sum_{k=1}^{\infty} x_2 = a_1 * y_1 + a_2 * y_2$$

It holds the superposition law and it is linear.
- **Non-Invertible:** Suppose that, we feed two different inputs to the system:

$$x[n] = 0$$

and

$$x[n] = u[n]$$

Then the outputs would be:

$$y[n] = \sum_{k=1}^{\infty} 0 = \sum_{k=1}^{\infty} u[n-k] = 0$$

Both input gives the same value. Therefore, the system is non-invertible.
- **Time-Invariant:** The shift on the input produces the same shift on the output:

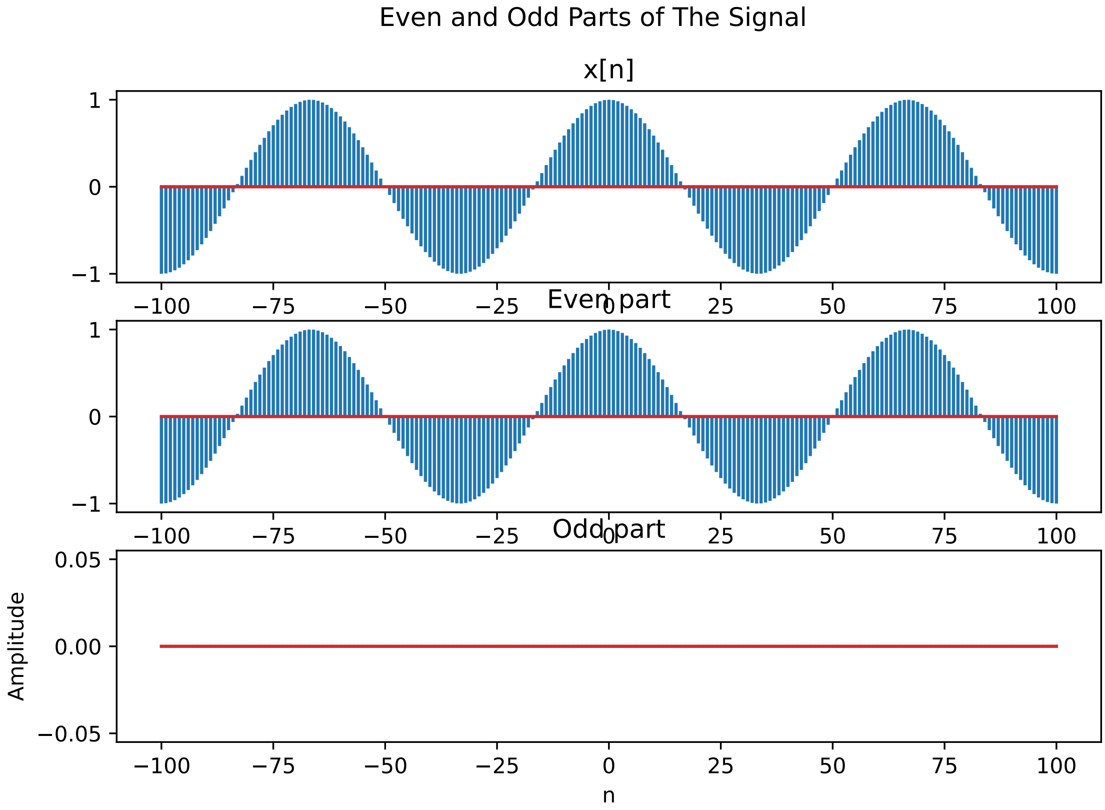
$$y[n] = \sum_{k=1}^{\infty} x[n-k] = \sum_{k=1}^{\infty} x[n-n_0-k] = y[n-n_0]$$

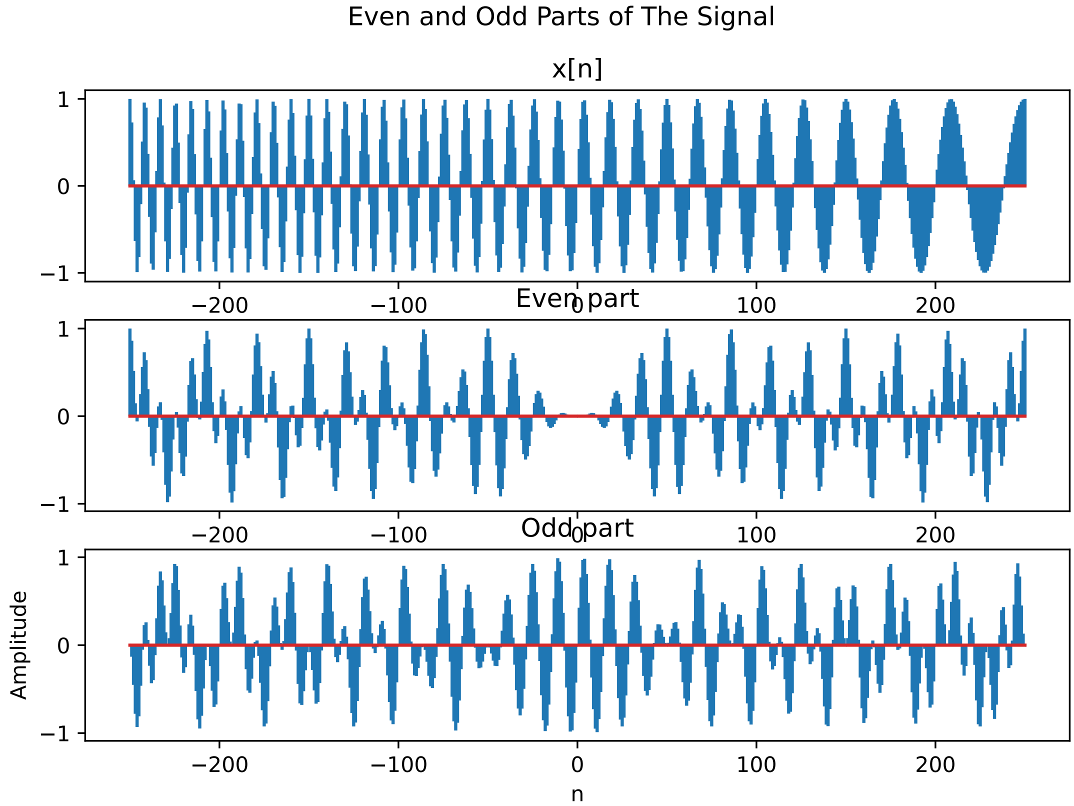
So it is time-invariant.

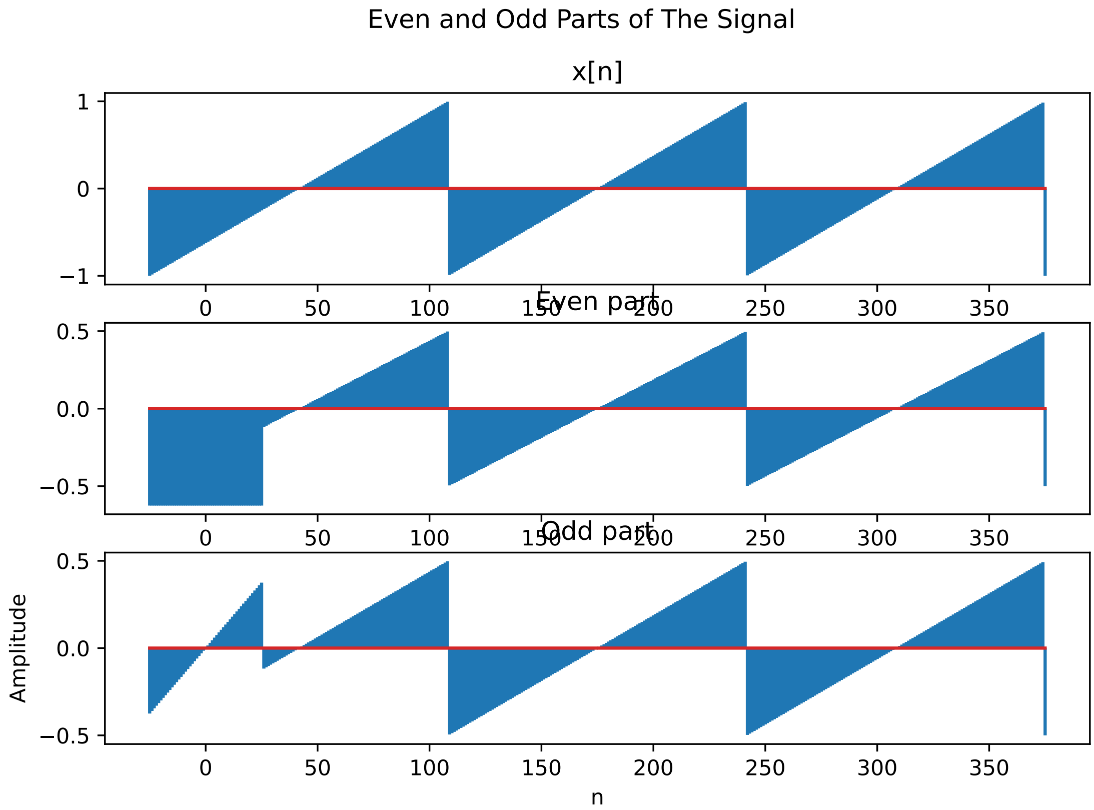
7. (a)

```python
import matplotlib.pyplot as plt

def part_a(filename):
    with open(filename, 'r') as f:

        data = f.read().split(",")
        starting_index = int(data[0])
        data = data[1:]
        data = [float(i) for i in data]

    ending_index = starting_index+len(data) -1
    x = list(range(starting_index, ending_index + 1))

    even_data = []
    odd_data = []

    k = abs(x[0])-abs(x[-1])
    if k<starting_index:
        for i in range(-k):
            x.insert(0,starting_index-i-1)
            data.insert(0,0)
    elif k>ending_index:
        for i in range(k):
            x.append(ending_index+i+1)
            data.insert(0,0)

    for i in range(len(data)):
        fx = data[i]
        fnegx = data[len(data)-i-1]

        even_data.append((fx + fnegx)/2)
        odd_data.append((fx - fnegx)/2)

    if k<starting_index:
        x = x[-k:]
        data = data[-k:]
        even_data = even_data[-k:]
        odd_data = odd_data[-k:]
    elif k>ending_index:
        x = x[:k]
        data = data[:k]
        even_data = even_data[:k]
        odd_data = odd_data[:k]


    fig, (ax1, ax2, ax3) = plt.subplots(3, 1, figsize=(16, 8))
    fig.set_size_inches(8.27, 5.5)

    ax1.stem(x, data,markerfmt='')
    ax1.set_title("x[n]")
    ax2.stem(x, even_data,markerfmt='')
    ax2.set_title("Even part")
    ax3.stem(x, odd_data,markerfmt='')
    ax3.set_title("Odd part")

    plt.xlabel('n')
    plt.ylabel('Amplitude')

    fig.suptitle('Even and Odd Parts of The Signal')
    plt.savefig(f"7a_{filename}.pdf", format="pdf", bbox_inches="tight"),
    plt.show()


part_a('sine_part_a.csv')
part_a('chirp_part_a.csv')
```

```
part_a('shifted_sawtooth_part_a.csv')
```

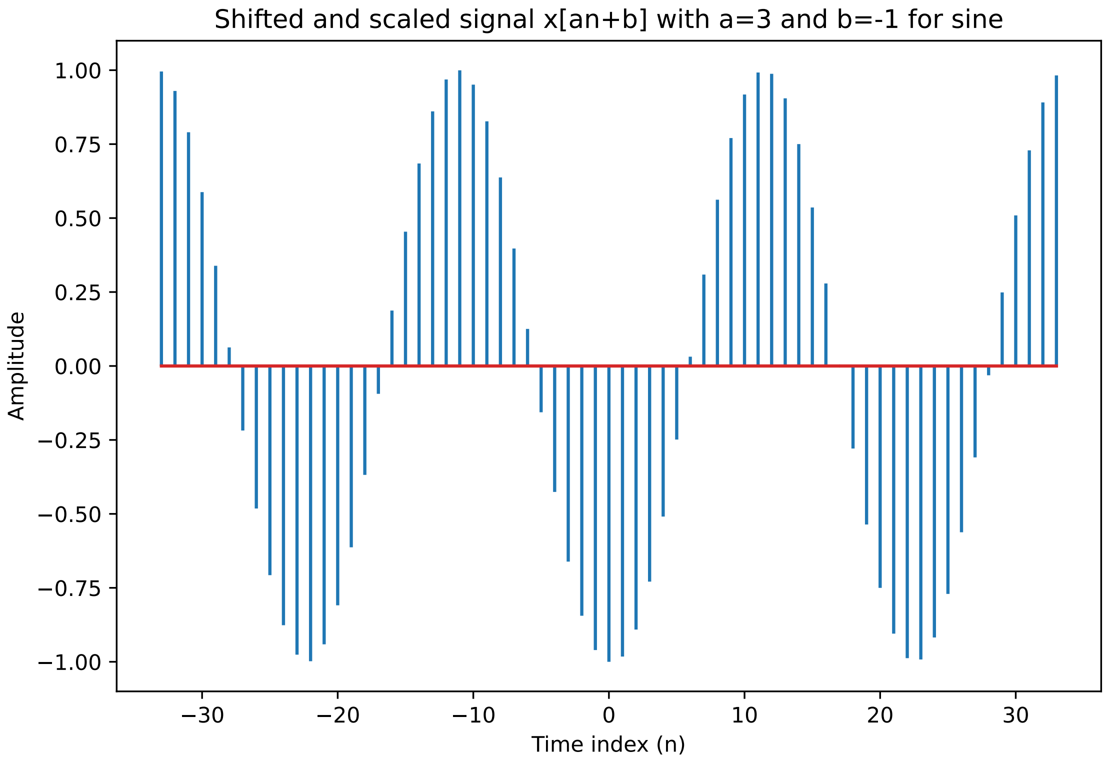


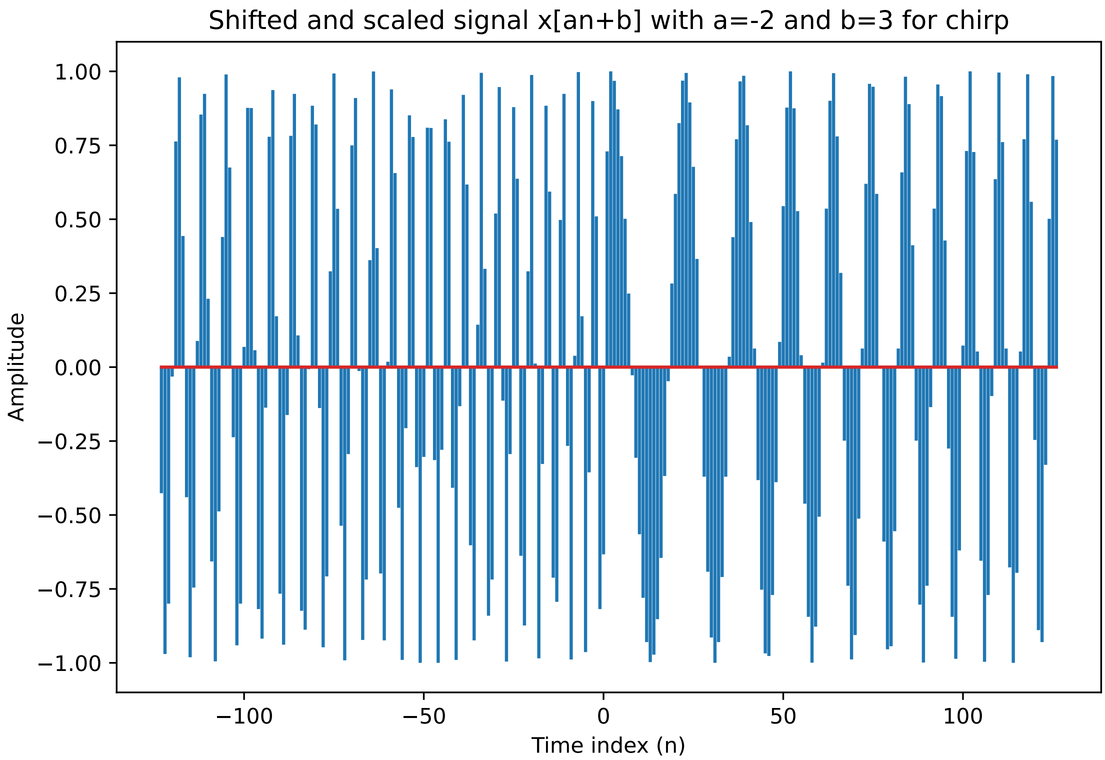


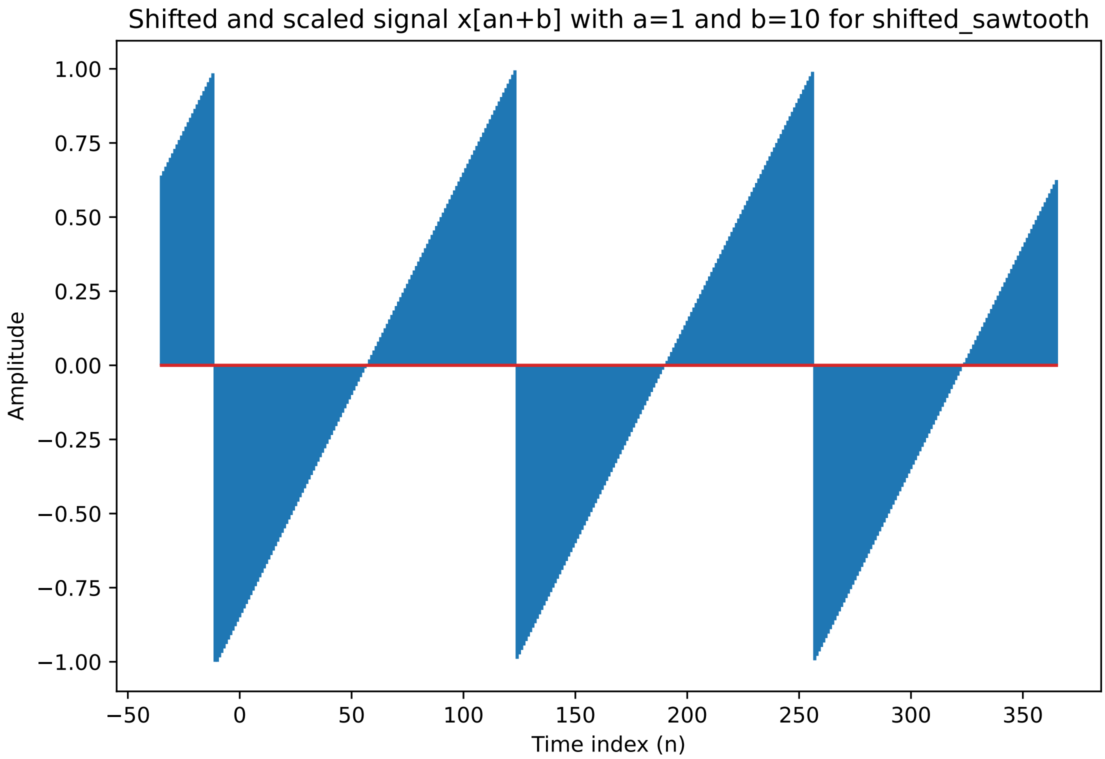


(b) -

```
import matplotlib.pyplot as plt

def part_b(filename):
    with open(filename, 'r') as f: #get and arrange the data

        data = f.read().split(",")
        starting_index = int(data[0])
        a =int(float(data[1]))
        b =int(float(data[2]))

        data = [float(i) for i in data[3:]]

    xx = [] #new time values
    shifted_signal = [] #new signal amplitude

    for n in range(starting_index, starting_index+len(data)):
        n_shifted = (n - b)/a    # compute the shifted index

        if(int(n_shifted)==n_shifted): #if the shifted index is not integer, it will di
            shifted_signal.append(data[n])
            xx.append(n_shifted)
    fig = plt.figure(figsize=(16, 8))
    fig.set_size_inches(8.27, 5.5)

    plt.stem(xx,shifted_signal,markerfmt="")  # plot the shifted and scaled signal
    plt.xlabel('Time index (n)')
    plt.ylabel('Amplitude')
    plt.title(f'Shifted and scaled signal x[an+b] with a={a} and b={b} for {filename[:-
    plt.savefig(f"7b_{filename}.pdf", format="pdf", bbox_inches="tight"),

    plt.show()

part_b('sine_part_b.csv')
part_b('chirp_part_b.csv')
part_b('shifted_sawtooth_part_b.csv')
```


Shifted and scaled signal x[an+b] with a=3 and b=-1 for sine
Amplitude
Time index (n)
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
−30
−20
−10
0
10
20
30


Shifted and scaled signal x[an+b] with a=-2 and b=3 for chirp
Amplitude
Time index (n)
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
−100
−50
0
50
100


Shifted and scaled signal x[an+b] with a=1 and b=10 for shifted_sawtooth
Amplitude
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
−50
0
50
100
150
200
250
300
350
Time index (n)