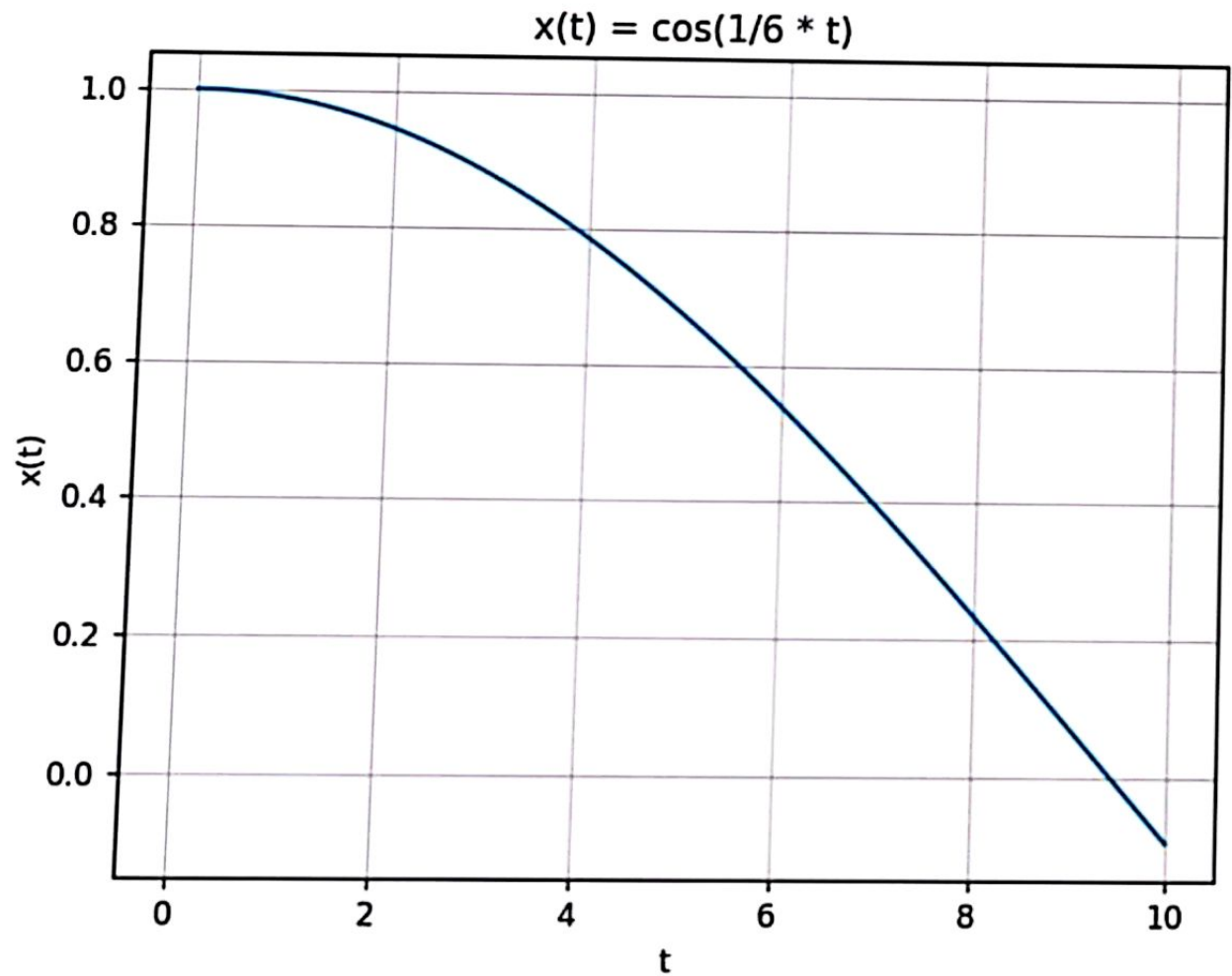


```

Welcome 1.py x
1 import numpy as np
2 import matplotlib.pyplot as plt
3
4 # محور زمان t تعریف دامنه
5 t = np.arange(0, 10, 0.01) # 0.01
6 x_t = np.cos((1/6) * t) # گنال
7
8 # رسم سیگنال
9 plt.figure()
10 plt.plot(t, x_t)
11 plt.xlabel('t')
12 plt.ylabel('x(t)')
13 plt.title('x(t) = cos(1/6 * t)')
14 plt.grid(True)
15 plt.show()
16

```

Figure 1



```

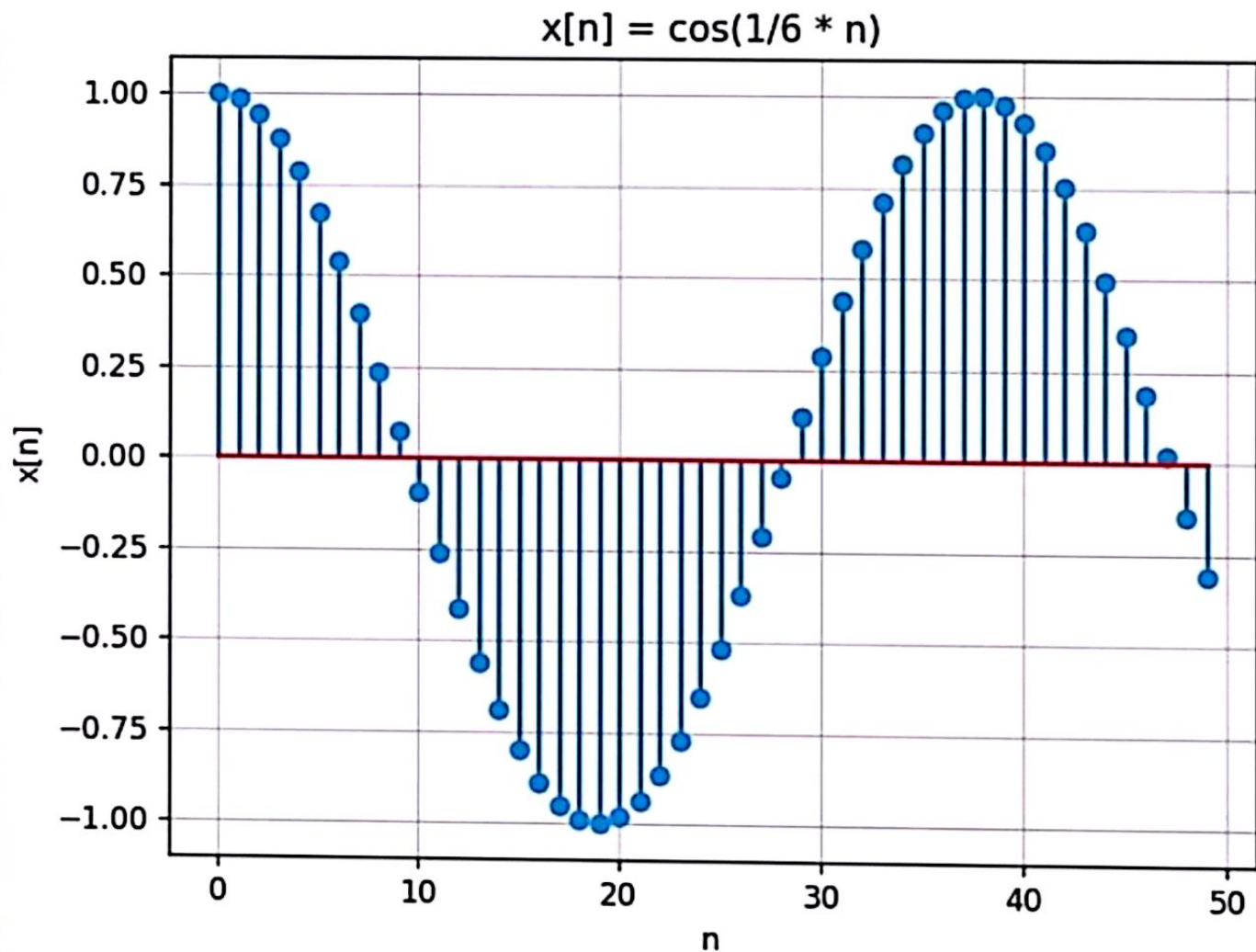
import numpy as np
import matplotlib.pyplot as plt

# محور گسسته n تعریف دامنه
n = np.arange(0, 50, 1) # (اد صحیح)
x_n = np.cos((1/6) * n) # سیگنال x[n]

# رسم سیگنال گسسته
plt.figure()
plt.stem(n, x_n)
plt.xlabel('n')
plt.ylabel('x[n]')
plt.title('x[n] = cos(1/6 * n)')
plt.grid(True)
plt.show()

```

Figure 1



```

1 import numpy as np
2 import matplotlib.pyplot as plt
3
4 # محور گسسته n تعریف دامنه
5 n = np.arange(0, 50, 1) # (اد صحیح)
6 x_n = np.cos((8 * np.pi / 31) * n)
7
8 # رسم سیگنال گسسته
9 plt.figure()
10 plt.stem(n, x_n)
11 plt.xlabel('n')
12 plt.ylabel('x[n]')
13 plt.title('x[n] = cos(8π/31 * n)')
14 plt.grid(True)
15 plt.show()
16

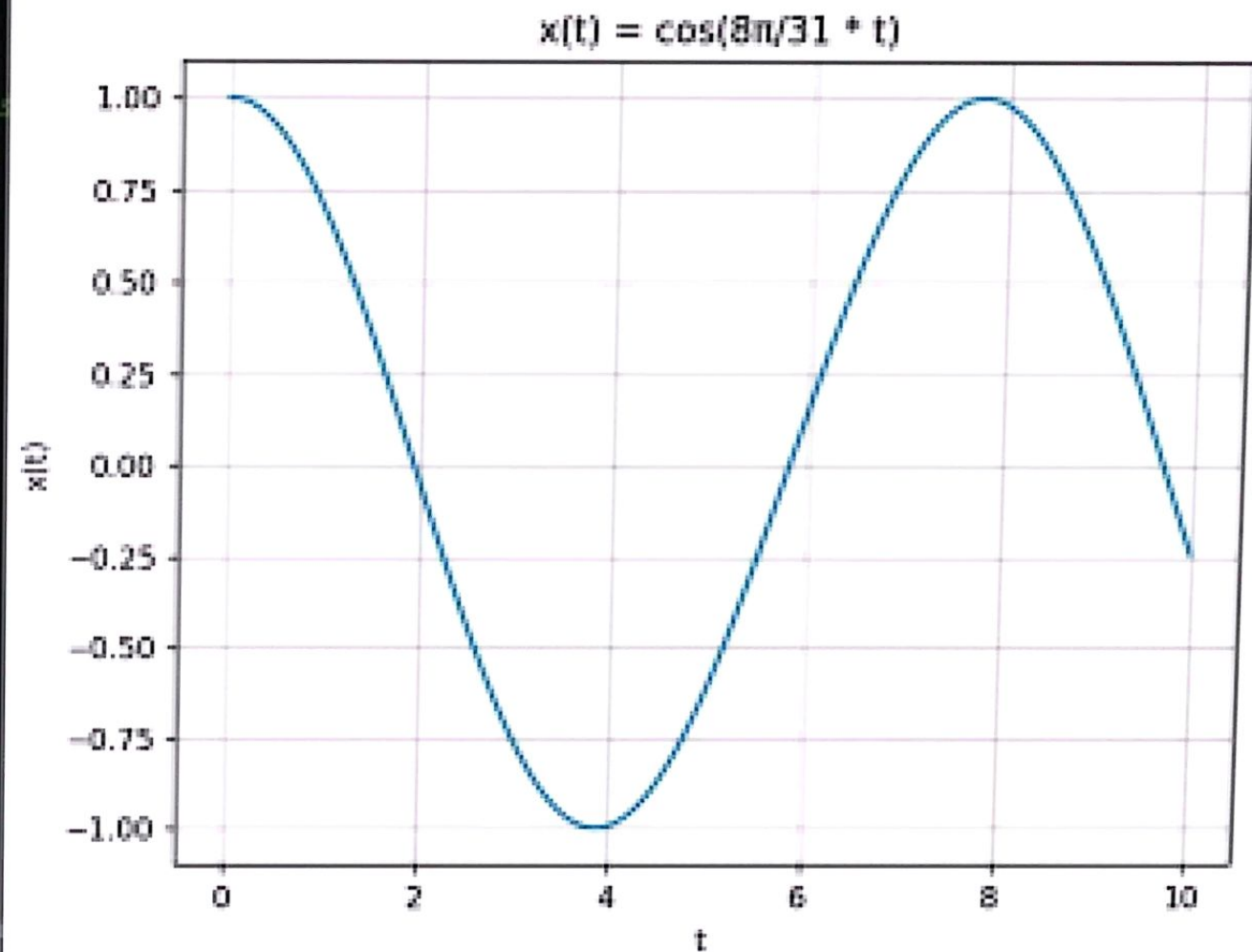
```



رسم سیگنال > 2.py > ...

```
1 import numpy as np
2 import matplotlib.pyplot as plt
3
4 # متغیر زمان t تعریف دامنه
5 t = np.arange(0, 10, 0.01) # 0.01 نام
6 x_t = np.cos((8 * np.pi / 31) * t) #
7
8 # رسم سیگنال
9 plt.figure()
10 plt.plot(t, x_t)
11 plt.xlabel('t')
12 plt.ylabel('x(t)')
13 plt.title('x(t) = cos(8π/31 * t)')
14 plt.grid(True)
15 plt.show()
16
```

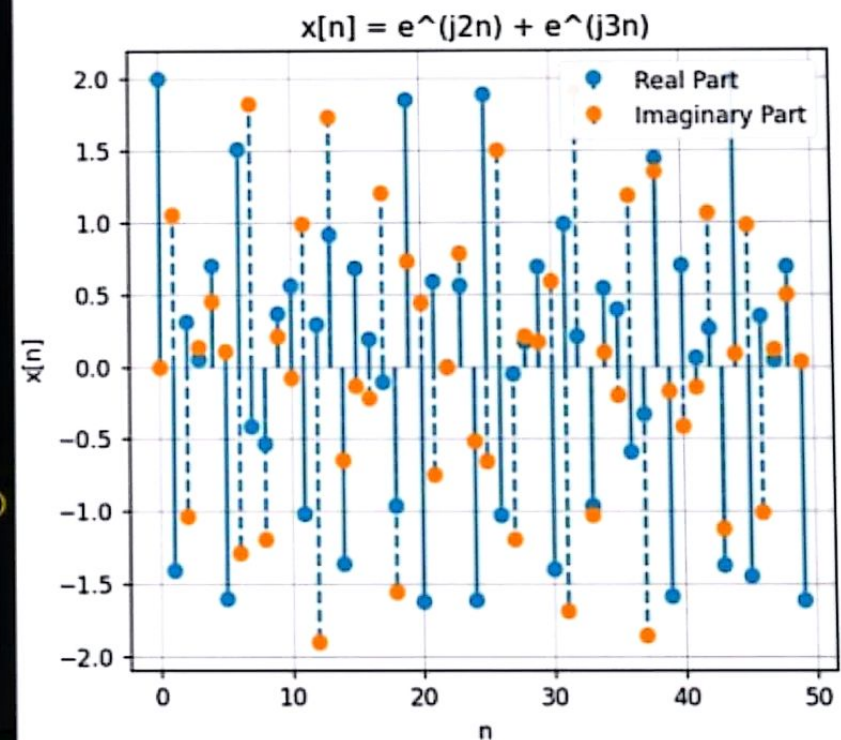
Figure 1



رسم سیگنال > 6.py > ...

```
1 import numpy as np
2 import matplotlib.pyplot as plt
3
4 # محور گسسته n تعریف دامنه
5 n = np.arange(0, 50, 1) # (اعداد صحیح) 1 با گام 50 تا 0 از
6
7 # تعریف سیگنال مختلط  $x[n] = e^{j2n} + e^{j3n}$ 
8 x_n = np.exp(1j * 2 * n) + np.exp(1j * 3 * n)
9
10 # استخراج قسمت حقیقی و موهومی
11 real_part = np.real(x_n)
12 imaginary_part = np.imag(x_n)
13
14 # رسم قسمت حقیقی سیگنال
15 plt.figure()
16 plt.stem(n, real_part, label='Real Part', basefmt=" ")
17 plt.stem(n, imaginary_part, label='Imaginary Part', basefmt=" ", linefmt='--', markerfmt='o')
18 plt.xlabel('n')
19 plt.ylabel('x[n]')
20 plt.title('x[n] = e^(j2n) + e^(j3n)')
21 plt.legend()
22 plt.grid(True)
23 plt.show()
24
```

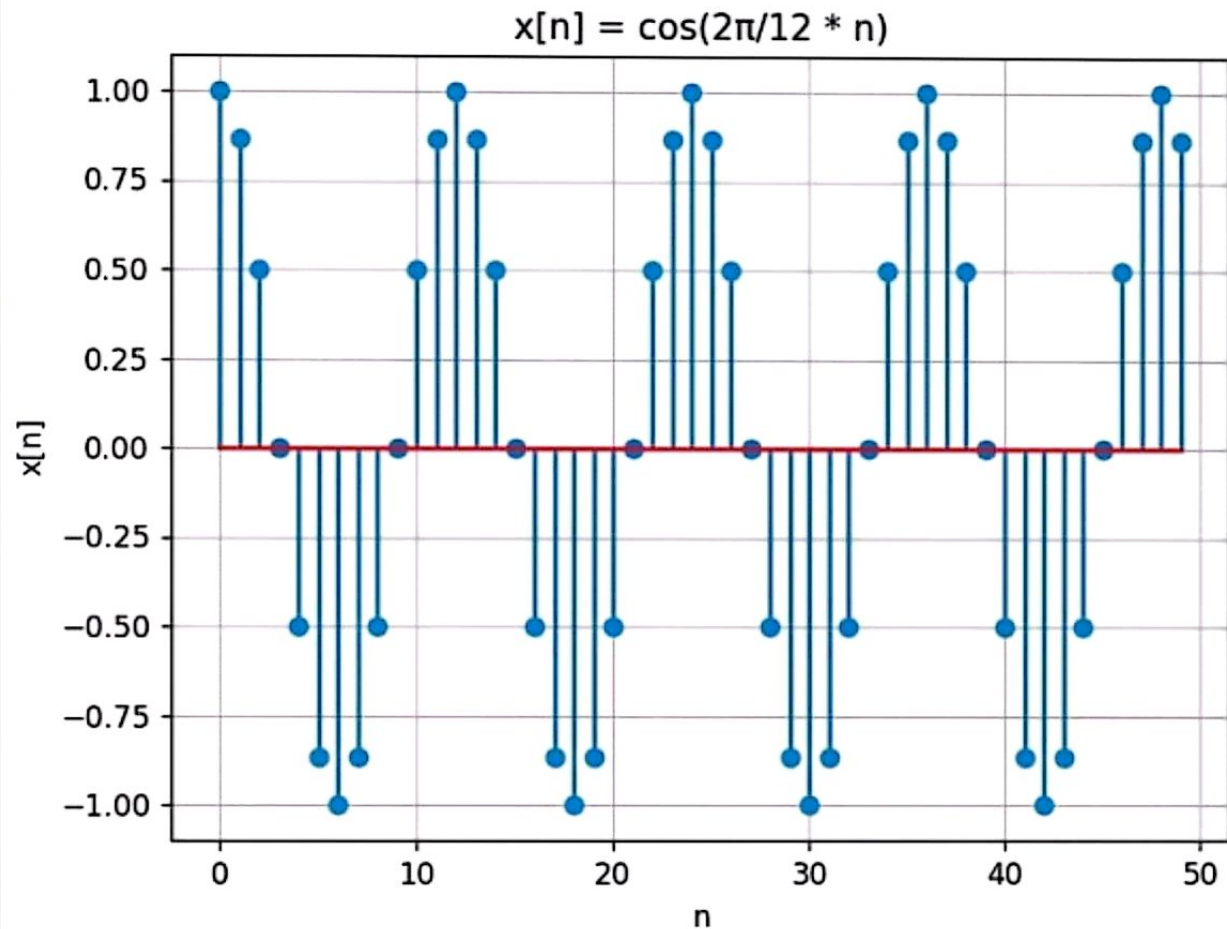
Figure 1



5.py > ...

```
1 import numpy as np
2 import matplotlib.pyplot as plt
3
4 # محور گسسته n تعریف دامنه
5 n = np.arange(0, 50, 1) # (تعداد صحیح)
6 x_n = np.cos((2 * np.pi / 12) * n) #
7
8 # رسم سیگنال گسسته
9 plt.figure()
10 plt.stem(n, x_n) # رسم سیگنال گسسته
11 plt.xlabel('n')
12 plt.ylabel('x[n]')
13 plt.title('x[n] = cos(2π/12 * n)')
14 plt.grid(True)
15 plt.show()
16
```

Figure 1



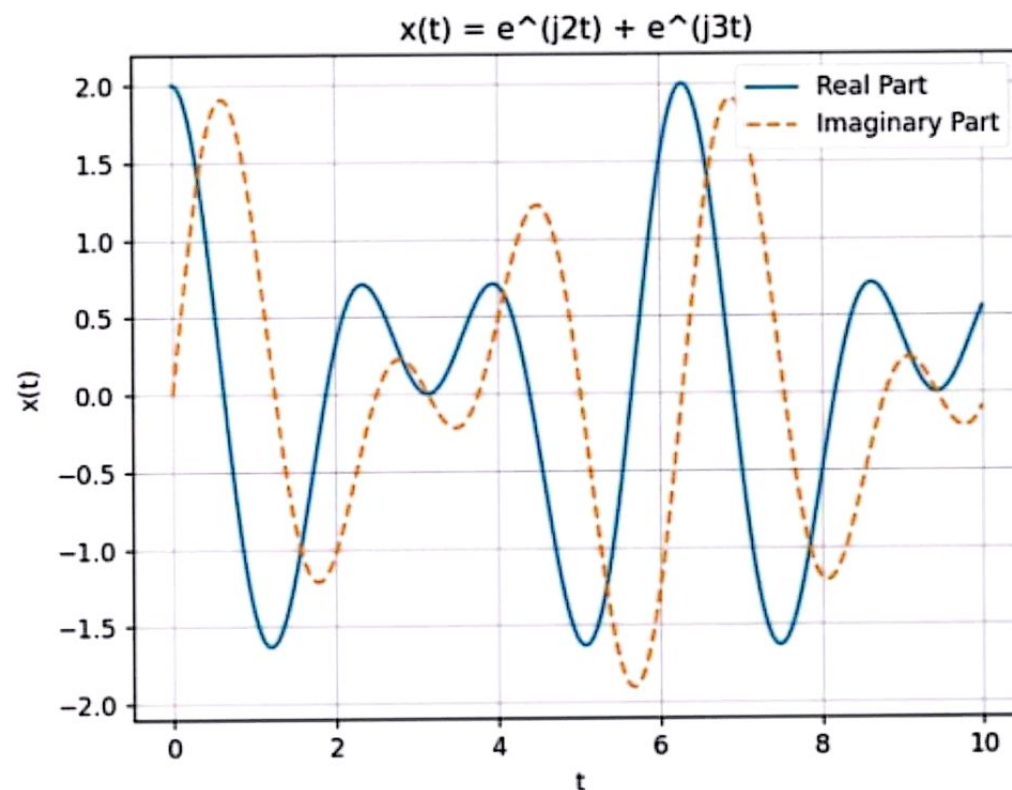
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```

5.py > ...
1  import numpy as np
2  import matplotlib.pyplot as plt
3
4  # محور زمان t تعریف دامنه
5  t = np.arange(0, 10, 0.01) # 0.01 با گام 10 تا 0
6
7  # تعریف سیگنال مختلط x(t) = e^(j2t) + e^(j3t)
8  x_t = np.exp(1j * 2 * t) + np.exp(1j * 3 * t)
9
10 # استخراج قسمت حقیقی و موهومی
11 real_part = np.real(x_t)
12 imaginary_part = np.imag(x_t)
13
14 # رسم قسمت حقیقی سیگنال
15 plt.figure()
16 plt.plot(t, real_part, label='Real Part')
17 plt.plot(t, imaginary_part, label='Imaginary Part', linestyle='--')
18 plt.xlabel('t')
19 plt.ylabel('x(t)')
20 plt.title('x(t) = e^(j2t) + e^(j3t)')
21 plt.legend()
22 plt.grid(True)
23 plt.show()

```

Figure 1

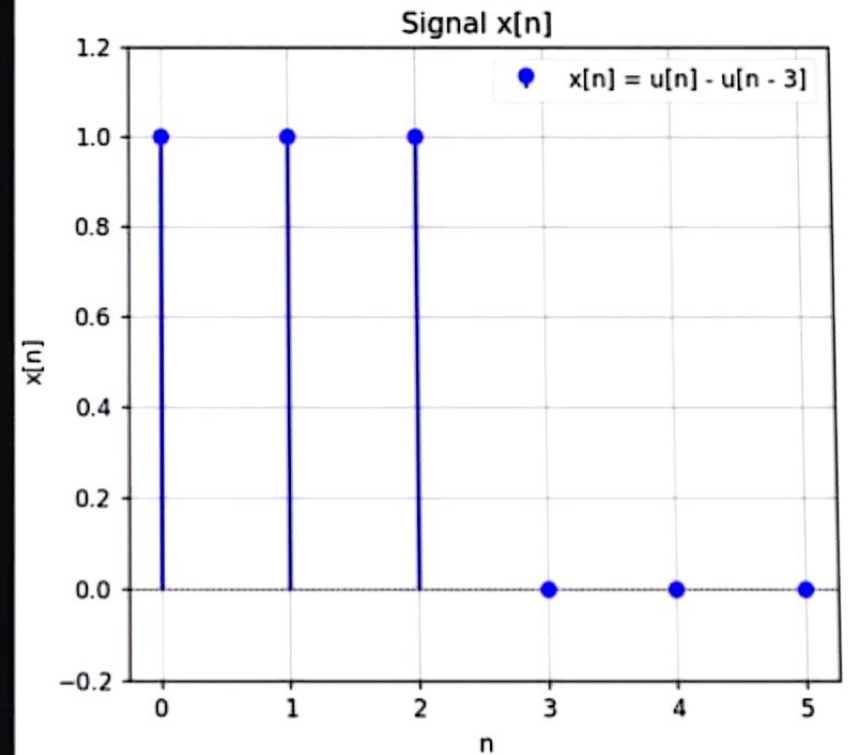



```

1 import numpy as np
2 import matplotlib.pyplot as plt
3
4 # محور گسسته n تعریف دامنه
5 n = np.arange(0, 6, 1) # 0 تا 5
6
7 # تعریف تابع واحد گسسته u[n]
8 def u(n):
9     return np.where(n >= 0, 1, 0)
10
11 # تعریف سیگنال x[n] = u[n] - u[n - 3]
12 x_n = u(n) - u(n - 3)
13
14 # رسم سیگنال
15 plt.figure()
16 plt.stem(n, x_n, basefmt=" ", linefmt='b-', markerfmt='bo', label='x[n] = u[n] - u[n - 3]')
17 plt.xlabel('n')
18 plt.ylabel('x[n]')
19 plt.title('Signal x[n]')
20 plt.axhline(0, color='black', lw=0.5, ls='--') # خط افقی صفر
21 plt.grid(True)
22 plt.ylim(-0.2, 1.2) # تنظیم محدوده محور عمودی
23 plt.legend()
24 plt.show()
25

```

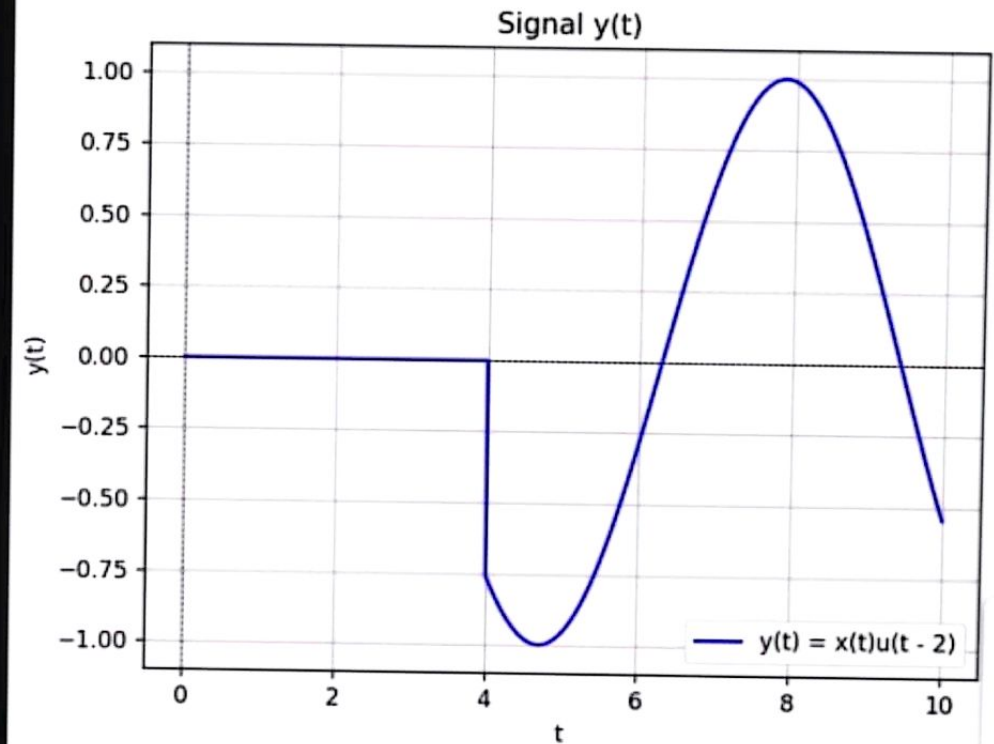
Figure 1



رسم سیگنال > 12.py > ...

```
1 import numpy as np
2 import matplotlib.pyplot as plt
3
4 # محور زمان t تعریف دامنه
5 t = np.linspace(0, 10, 1000) # از 0 تا 10 با 1000 نقطه
6
7 # تعریف تابع x(t)
8 x_t = np.sin(t) # را به عنوان سینوس تعریف کردیم
9
10 # تعریف تابع u(t-2)
11 def u(t):
12     return np.where(t >= 2, 1, 0)
13
14 # تعریف سیگنال y(t) = x(t) * u(t-2)
15 y_t = x_t * u(t - 2)
16
17 # رسم سیگنال
18 plt.figure()
19 plt.plot(t, y_t, label='y(t) = x(t)u(t - 2)', color='blue')
20 plt.xlabel('t')
21 plt.ylabel('y(t)')
22 plt.title('Signal y(t)')
23 plt.axhline(0, color='black', lw=0.5, ls='--') # خط افقی صفر
24 plt.axvline(2, color='black', lw=0.5, ls='--') # خط عمودی صفر
25 plt.grid(True)
26 plt.legend()
27 plt.show()
28
```

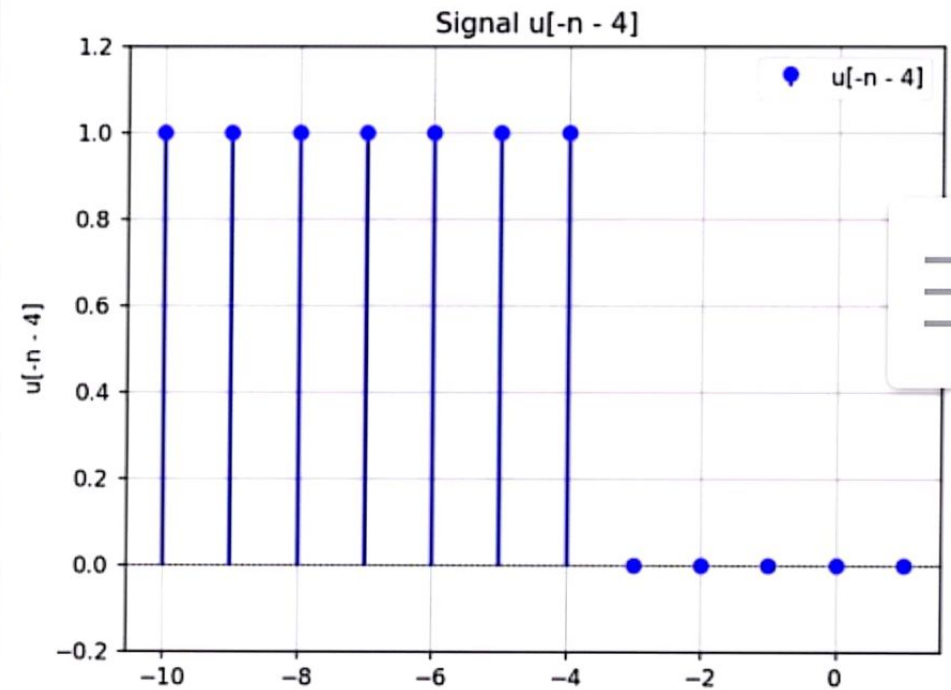
Figure 1



11.py > ...

```
1 import numpy as np
2 import matplotlib.pyplot as plt
3
4 # محور گسسته n تعریف دامنه
5 n = np.arange(-10, 2, 1) # از -10 تا 1
6
7 # تعریف تابع واحد گسسته u[n]
8 def u(n):
9     return np.where(n >= 0, 1, 0)
10
11 # تعریف سیگنال u[-n - 4]
12 x_n = u(-n - 4)
13
14 # رسم سیگنال
15 plt.figure()
16 plt.stem(n, x_n, basefmt=" ", linefmt='b-', markerfmt='bo', label='u[-n - 4]')
17 plt.xlabel('n')
18 plt.ylabel('u[-n - 4]')
19 plt.title('Signal u[-n - 4]')
20 plt.axhline(0, color='black', lw=0.5, ls='--') # خط افقی صفر
21 plt.grid(True)
22 plt.ylim(-0.2, 1.2) # تنظیم محدوده محور عمودی
23 plt.legend()
24 plt.show()
25
```

Figure 1



```

# محور زمان t تعریف دامنه
t = np.linspace(-1, 3, 1000) # از -1 تا 3 با 1000 نقطه

# تعریف تابع واحد u(t)
def u(t):
    return np.where(t >= 0, 1, 0)

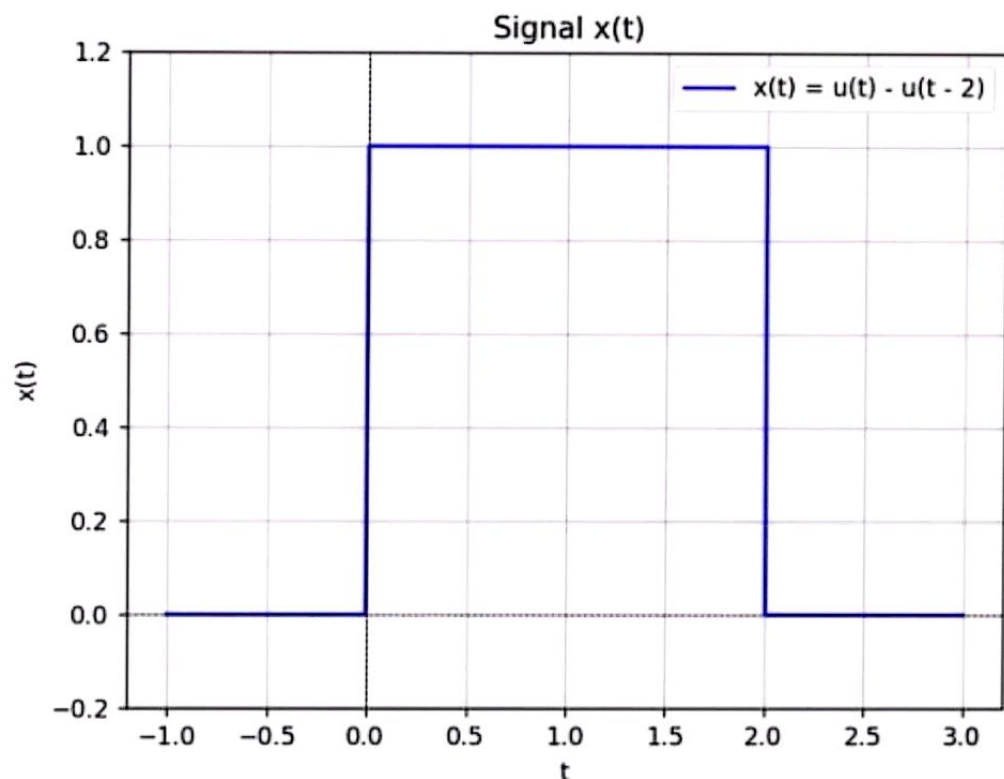
# تعریف سیگنال x(t) = u(t) - u(t - 2)
x_t = u(t) - u(t - 2)

# رسم سیگنال
plt.figure()
plt.plot(t, x_t, label='x(t) = u(t) - u(t - 2)', color='blue')
plt.xlabel('t')
plt.ylabel('x(t)')
plt.title('Signal x(t)')
plt.axhline(0, color='black', lw=0.5, ls='--') # خط افقی صفر
plt.axvline(0, color='black', lw=0.5, ls='--') # خط عمودی صفر
plt.grid(True)
plt.ylim(-0.2, 1.2) # تنظیم محدوده محور عمودی
plt.legend()
plt.show()

```

PROBLEMS OUTPUT DEBUG CONSOLE **TERMINAL** PORTS
 E:\university\موسسه\موسسه ل ان گیس & C:/Users/Mr.Sani/AppData/Local/Programs/P

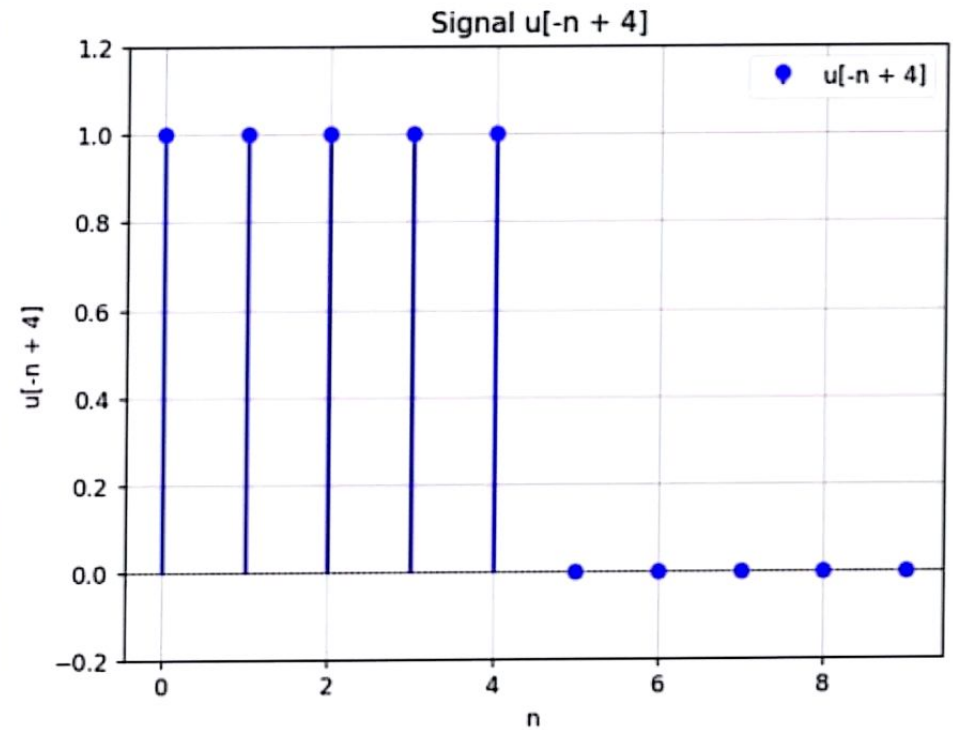
Figure 1



(x, y) = (-0.973, -0.00)

رسم سیگنال > 10.py > ...

```
1 import numpy as np
2 import matplotlib.pyplot as plt
3
4 # محور گسسته n تعریف دامنه
5 n = np.arange(0, 10, 1) # 0 تا 9
6
7 # تعریف تابع واحد گسسته u[n]
8 def u(n):
9     return np.where(n >= 0, 1, 0)
10
11 # تعریف سیگنال u[-n + 4]
12 x_n = u(-n + 4)
13
14 # رسم سیگنال
15 plt.figure()
16 plt.stem(n, x_n, basefmt=" ", linefmt='b-', markerfmt='bo', label='u[-n + 4]')
17 plt.xlabel('n')
18 plt.ylabel('u[-n + 4]')
19 plt.title('Signal u[-n + 4]')
20 plt.axhline(0, color='black', lw=0.5, ls='--') # خط افقی مقرر
21 plt.grid(True)
22 plt.ylim(-0.2, 1.2) # تنظیم محدوده محور عمودی
23 plt.legend()
24 plt.show()
25
```



```

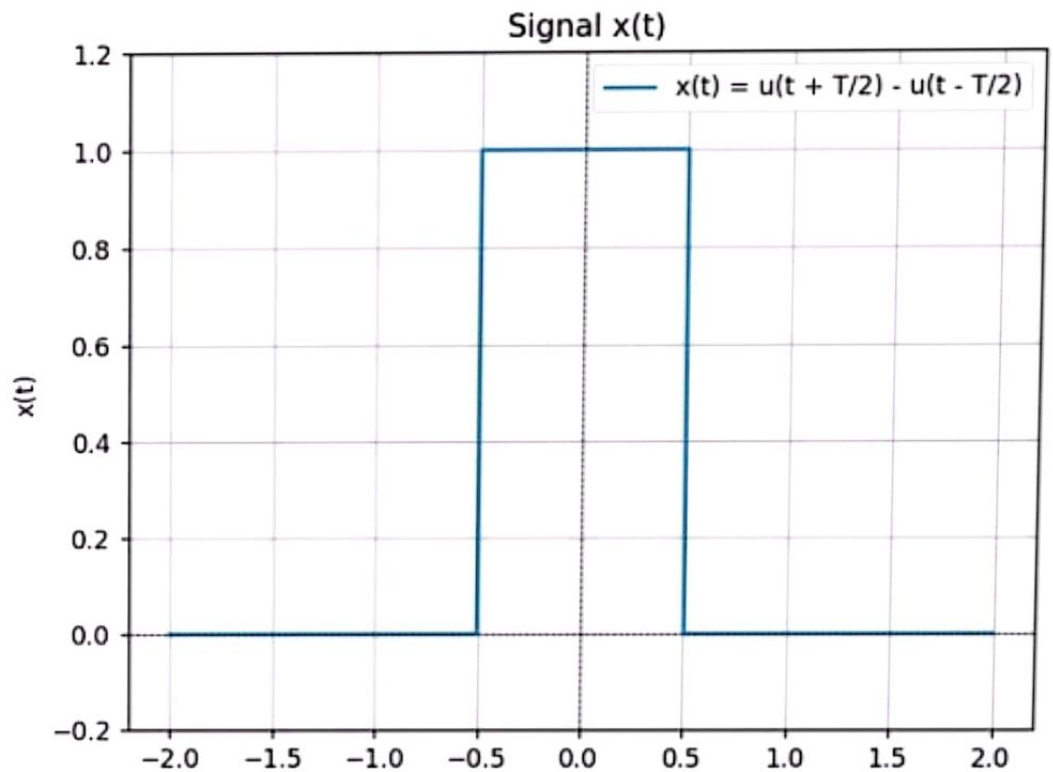
7  # تعريف T
8  T = 1
9
10 # تعريف تابع واحد u(t)
11 def u(t):
12     return np.where(t >= 0, 1, 0)
13
14 # تعريف سيگنال x(t) = u(t + T/2) - u(t - T/2)
15 x_t = u(t + T/2) - u(t - T/2)
16
17 # رسم سيگنال
18 plt.figure()
19 plt.plot(t, x_t, label='x(t) = u(t + T/2) - u(t - T/2)')
20 plt.xlabel('t')
21 plt.ylabel('x(t)')
22 plt.title('Signal x(t)')
23 plt.axhline(0, color='black', lw=0.5, ls='--') # خط افقی صفر
24 plt.axvline(0, color='black', lw=0.5, ls='--') # خط عمودی صفر
25 plt.grid(True)
26 plt.ylim(-0.2, 1.2) # تنظیم محدوده محور عمودی
27 plt.legend()
28 plt.show()
29

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

E:\university\انگیزش ل\مترجم ل\انگیزش ل & C:\Users\Mr.Sani\AppData\Local\Programs\Python

Figure 1



```

6.py 7.py 8.py 9.py
> 14.py > ...
3
1 # رسم مشتق f3(t)
2 plt.subplot(3, 1, 2)
3 plt.plot(t, f3_derivative, label="f3'(t)", color='orange')
4 plt.xlabel('t')
5 plt.ylabel("f3'(t)")
6 plt.title('Derivative of f3(t)')
7 plt.axhline(0, color='black', lw=0.5, ls='--') # فقی صفر
8 plt.axvline(0, color='black', lw=0.5, ls='--') # بودی صفر
9 plt.grid(True)
10 plt.legend()
11
12 # رسم انتگرال f3(t)
13 plt.subplot(3, 1, 3)
14 plt.plot(t, f3_integral, label='∫f3(t) dt', color='green')
15 plt.xlabel('t')
16 plt.ylabel('∫f3(t) dt')
17 plt.title('Integral of f3(t)')
18 plt.axhline(0, color='black', lw=0.5, ls='--') # فقی صفر
19 plt.axvline(0, color='black', lw=0.5, ls='--') # بودی صفر
20 plt.grid(True)
21 plt.legend()
22
23 plt.tight_layout()
24 plt.show()
25

```

PROBLEMS OUTPUT DEBUG CONSOLE **TERMINAL** PORTS
 E:\university\انگوش ل انگوین و اپام تروین ل انگوین & C:/Users/Mr.Sani/AppData/Local

