
 Marwadi University Marwadi Chandarana Group 	Marwadi University Faculty of Engineering & Technology Department of Information and Communication Technology	
Subject: Programming With Python (01CT1309)	Aim: Analysis of Discrete-Time Signals Using Z-Transform	
Experiment No: 17	Date:20/11/2025	Enrollment No:92510133049

Aim: Analysis of Discrete-Time Signals Using Z-Transform



IDE:

Install Library

pip install sympy

Example 1:



```
import sympy as sp
# Define symbols
n, z, a = sp.symbols('n z a')
# Define the signal  $x[n] = a^n * u[n]$ 
x_n = a**n
# Compute the Z-transform
X_z = sp.summation(x_n * z**(-n), (n, 0, sp.oo))
# Print the result
print("Z-transform of  $x[n] = a^n u[n]$ :")
sp.pprint(X_z, use_unicode=True)
```


 Marwadi University Marwadi Chandarana Group 	Marwadi University Faculty of Engineering & Technology Department of Information and Communication Technology	
Subject: Programming With Python (01CT1309)	Aim: Analysis of Discrete-Time Signals Using Z-Transform	
Experiment No: 17	Date:20/11/2025	Enrollment No:92510133049

```

x_n = 2**n
# Compute the Z-transform
X_z = sp.summation(x_n * z**(-n), (n, 0, sp.oo))
# Print the result
print("Z-transform of x[n] = a^n u[n]:")

```

 Marwadi University Marwadi Chandarana Group 	Marwadi University Faculty of Engineering & Technology Department of Information and Communication Technology	
Subject: Programming With Python (01CT1309)	Aim: Analysis of Discrete-Time Signals Using Z-Transform	
Experiment No: 17	Date:20/11/2025	Enrollment No:92510133049

```
sp.pprint(X_z, use_unicode=True)
```

Subject: Programming With Python (01CT1309)

Aim: Analysis of Discrete-Time Signals Using Z-Transform

Experiment No: 17

Date:20/11/2025

Enrollment No:92510133049

C: > Users > S Madesh > OneDrive > Documents > lab 20.py > ...

```
1 import sympy as sp
2 # Define symbols
3 n, z, a = sp.symbols('n z a')
4 # Define the signal x[n] = a^n * u[n]
5 x_n = a**n
6 # Compute the Z-transform
7 X_z = sp.summation(x_n * z**(-n), (n, 0, sp.oo))
8 # Simplify the result
9 X_z_simpl = sp.simplify(X_z)
10 # Print the result
11 print("Z-transform of x[n] = a^n u[n]:")
12 sp.pprint(X_z_simpl, use_unicode=True)
13 # Print region of convergence
14 print("Region of convergence: |z| > |a|")
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS



PS B:\popcket-hub> & "C:/Users/S Madesh/AppData/Local/Microsoft/WindowsApps/python3.13.6

Z-transform of x[n] = a^n u[n]:

$$\left\{ \begin{array}{ll} \frac{z}{-a + z} & \text{for } \left| \frac{a}{z} \right| < 1 \\ \sum_{n=0}^{\infty} a^n \cdot z^{-n} & \text{otherwise} \end{array} \right.$$



Region of convergence: |z| > |a|

PS B:\popcket-hub>


 Marwadi University Marwadi Chandarana Group 	Marwadi University Faculty of Engineering & Technology Department of Information and Communication Technology	
Subject: Programming With Python (01CT1309)	Aim: Analysis of Discrete-Time Signals Using Z-Transform	
Experiment No: 17	Date:20/11/2025	Enrollment No:92510133049

Example 3:

```
import sympy as sp
# Define symbols
n, z = sp.symbols('n z')
# Define the unit step signal u[n]
u_n = 1
# Compute the Z-transform
U_z = sp.summation(u_n * z**(-n), (n, 0, sp.oo))
# Print the result
print("Z-transform of the unit step signal u[n]:")
```

 Marwadi University Marwadi Chandarana Group 	Marwadi University Faculty of Engineering & Technology Department of Information and Communication Technology	
Subject: Programming With Python (01CT1309)	Aim: Analysis of Discrete-Time Signals Using Z-Transform	
Experiment No: 17	Date:20/11/2025	Enrollment No:92510133049

```
sp.pprint(U_z, use_unicode=True)
```

 Marwadi University Marwadi Chandarana Group	Marwadi University Faculty of Engineering & Technology Department of Information and Communication Technology
Subject: Programming With Python (01CT1309)	Aim: Analysis of Discrete-Time Signals Using Z-Transform
Experiment No: 17	Date: 20/11/2025 Enrollment No: 92510133049

C: > Users > S Madesh > OneDrive > Documents > lab 20.py > ...

```



1  import sympy as sp
2  # Define symbols
3  n, z = sp.symbols('n z')
4  # Define the unit step signal u[n]
5  u_n = 1
6  # Compute the Z-transform
7  U_z = sp.summation(u_n * z**(-n), (n, 0, sp.oo))
8  # Print the result
9  print("Z-transform of the unit step signal u[n]:")
10 sp.pprint(U_z, use_unicode=True)
11 # Define the unit impulse signal delta[n]

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS



PS B:\popkett-hub> & "C:/Users/S Madesh/AppData/Local/Microsoft/WindowsA

$$\begin{aligned}
 & \text{for } |z| < 1 \\
 & 1 - \frac{1}{z} \\
 & \sum_{n=0}^{\infty} z^{-n} \quad \text{otherwise}
 \end{aligned}$$


 Marwadi University Marwadi Chandarana Group 	Marwadi University Faculty of Engineering & Technology Department of Information and Communication Technology	
Subject: Programming With Python (01CT1309)	Aim: Analysis of Discrete-Time Signals Using Z-Transform	
Experiment No: 17	Date:20/11/2025	Enrollment No:92510133049

Example 4:

```
import sympy as sp
# Define symbols
n, z, alpha = sp.symbols('n z alpha')
# Define the signal x[n] = exp(alpha * n) * u[n]
x_n = sp.exp(alpha * n)
# Compute the Z-transform
X_z = sp.summation(x_n * z**(-n), (n, 0, sp.oo))
# Print the result
print("Z-transform of x[n] = exp(alpha * n) u[n]:")
```

 Marwadi University Marwadi Chandarana Group 	Marwadi University Faculty of Engineering & Technology Department of Information and Communication Technology	
Subject: Programming With Python (01CT1309)	Aim: Analysis of Discrete-Time Signals Using Z-Transform	
Experiment No: 17	Date:20/11/2025	Enrollment No:92510133049

```
sp.pprint(X_z, use_unicode=True)
```

 Marwadi University Marwadi Chandarana Group	Marwadi University Faculty of Engineering & Technology Department of Information and Communication Technology
Subject: Programming With Python (01CT1309)	Aim: Analysis of Discrete-Time Signals Using Z-Transform
Experiment No: 17	Date: 20/11/2025 Enrollment No: 92510133049

```

1  import sympy as sp
2  # Define symbols
3  n, z, alpha = sp.symbols('n z alpha')
4  # Define the signal x[n] = exp(alpha * n) * u[n]
5  x_n = sp.exp(alpha * n)
6  # Compute the Z-transform
7  X_z = sp.summation(x_n * z**(-n), (n, 0, sp.oo))
8  # Print the result
9  print("Z-transform of x[n] = exp(alpha * n) u[n]:")
10 sp.pprint(X_z, use_unicode=True)
11 # Define the region of convergence

```



PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

● PS B:\popcket-hub> & "C:/Users/S Madesh/AppData/Local/Microsoft/WindowsAp
Z-transform of x[n] = exp(alpha * n) u[n]:

$$\sum_{n=0}^{\infty} z^{-n} \cdot e^{\alpha \cdot n}$$


n = 0

○ PS B:\popcket-hub>

 Marwadi University Marwadi Chandarana Group 	Marwadi University Faculty of Engineering & Technology Department of Information and Communication Technology	
Subject: Programming With Python (01CT1309)	Aim: Analysis of Discrete-Time Signals Using Z-Transform	
Experiment No: 17	Date:20/11/2025	Enrollment No:92510133049

Example 5:

```
import sympy as sp
# Define symbols
n, z = sp.symbols('n z')
# Define the finite sequence x[n] = {1, 2, 3}
x_n = [1, 2, 3]
# Compute the Z-transform manually
X_z = sum(x_n[i] * z**(-i) for i in range(len(x_n)))
# Print the result
print("Z-transform of the finite sequence {1, 2, 3}:")
```



 Marwadi University Marwadi Chandarana Group	Marwadi University Faculty of Engineering & Technology Department of Information and Communication Technology	
Subject: Programming With Python (01CT1309)	Aim: Analysis of Discrete-Time Signals Using Z-Transform	
Experiment No: 17	Date: 20/11/2025	Enrollment No: 92510133049

```
sp.pprint(X_z, use_unicode=True)
```

```
C: > Users > S Madesh > OneDrive > Documents > lab 20.py > ...
1  import sympy as sp
2  # Define symbols
3  n, z = sp.symbols('n z')
4  # Define the finite sequence x[n] = {1, 2, 3}
5  x_n = [1, 2, 3]
6  # Compute the Z-transform manually
7  X_z = sum(x_n[i] * z**(-i) for i in range(len(x_n)))
8  # Print the result
9  print("Z-transform of the finite sequence {1, 2, 3}:")
10 sp.pprint(X_z, use_unicode=True)
11 |
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS B:\popcket-hub> & "C:/Users/S Madesh/AppData/Local/Microsoft/WindowsApp
Z-transform of the finite sequence {1, 2, 3}:
$$1 + \frac{2}{z} + \frac{3}{z^2}$$
PS B:\popcket-hub>



 Marwadi University Marwadi Chandarana Group 	Marwadi University Faculty of Engineering & Technology Department of Information and Communication Technology	
Subject: Programming With Python (01CT1309)	Aim: Analysis of Discrete-Time Signals Using Z-Transform	
Experiment No: 17	Date:20/11/2025	Enrollment No:92510133049

Example 6

```

import sympy as sp
# Define symbols
n, z, omega = sp.symbols('n z omega')
# Define the sinusoidal sequence  $x[n] = \sin(\omega * n) * u[n]$ 
x_n = sp.sin(omega * n)
# Compute the Z-transform
X_z = sp.summation(x_n * z**(-n), (n, 0, sp.oo))
# Print the result
print("Z-transform of  $x[n] = \sin(\omega * n) u[n]$ :")
sp.pprint(X_z, use_unicode=True)

```

 Marwadi University Marwadi Chandarana Group 	Marwadi University Faculty of Engineering & Technology Department of Information and Communication Technology	
Subject: Programming With Python (01CT1309)	Aim: Analysis of Discrete-Time Signals Using Z-Transform	
Experiment No: 17	Date: 20/11/2025	Enrollment No: 92510133049

```

1  import sympy as sp
2  # Define symbols
3  n, z, omega = sp.symbols('n z omega')
4  # Define the sinusoidal sequence x[n] = sin(omega * n)
5  x_n = sp.sin(omega * n)
6  # Compute the Z-transform
7  X_z = sp.summation(x_n * z**(-n), (n, 0, sp.oo))
8  # Print the result
9  print("Z-transform of x[n] = sin(omega * n) u[n]:")
10 sp.pprint(X_z, use_unicode=True)
11 import matplotlib.pyplot as plt

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS B:\popcket-hub> & "C:/Users/S Madesh/AppData/Local/Microso



● Z-transform of x[n] = sin(omega * n) u[n]:

∞

$$\sum_{n=0}^{\infty} z^{-n} \cdot \sin(n \cdot \omega)$$

n = 0

○ PS B:\popcket-hub>

 Marwadi University Marwadi Chandarana Group 	Marwadi University Faculty of Engineering & Technology Department of Information and Communication Technology	
Subject: Programming With Python (01CT1309)	Aim: Analysis of Discrete-Time Signals Using Z-Transform	
Experiment No: 17	Date: 20/11/2025	Enrollment No: 92510133049

Post Lab Exercise:

- Using Python, compute the Z-transform of the sequence $x[n] = 3^n u[n]$.

```

1  import sympy as sp
2
3  # Define symbols
4  n, z = sp.symbols('n z', real=True, positive=True)
5
6  # Define the sequence x[n] = 3^n * u[n]
7  # where u[n] is the unit step function (1 for n >= 0)
8  x_n = 3**n
9
10 # Compute the Z-transform
11 X_z = sp.summation(x_n * z**(-n), (n, 0, sp.oo))
12
13 # Simplify the result
14 X_z_simplified = sp.simplify(X_z)
15
16 print("Z-transform using SymPy:")
17 print(f"X(z) = {X_z_simplified}")
18 print(["Region of convergence: |z| > 3"])


```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```

PS B:\popcket-hub> & "C:/Users/S Madesh/AppData/Local/Microsoft/WindowsApps/python3.13.
● Z-transform using SymPy:
X(z) = Piecewise((z/(z - 3), 1/z < 1/3), (Sum(3**n/z**n, (n, 0, oo)), True))
○ PS B:\popcket-hub>

```

 Marwadi University Marwadi Chandarana Group	Marwadi University Faculty of Engineering & Technology Department of Information and Communication Technology	
Subject: Programming With Python (01CT1309)	Aim: Analysis of Discrete-Time Signals Using Z-Transform	
Experiment No: 17	Date: 20/11/2025	Enrollment No: 92510133049

- Using Python, compute the Z-transform of the sequence $x[n] = \cos(wn)u[n]$.

```

1  import sympy as sp
2
3  # Define symbols
4  n, z, w = sp.symbols('n z w', real=True, positive=True)
5
6  # Define the sequence x[n] = cos(wn) * u[n]
7  x_n = sp.cos(w * n)
8
9  # Compute the Z-transform
10 X_z = sp.summation(x_n * z**(-n), (n, 0, sp.oo))
11
12 print("Z-transform using SymPy summation:")
13 print(f"X(z) = {X_z}")

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```

PS B:\popcket-hub> & "C:/Users/S Madesh/AppData/Local/Microsoft/WindowsApps/python3.13.exe" "c:/Users/S Madesh/
Z-transform using SymPy summation:
X(z) = Sum(cos(n*w)/z**n, (n, 0, oo))
PS B:\popcket-hub>

```

--