

Date: 06/09/2025

Timing: 9:30 am to 4:15 pm

Long Hour Coding Exam

Task: Design and Implementation of a Python Package `signal_ICT_StudentName_EnrollmentNo` for Signal Generation and Operations.

Problem Statement:

You are required to design a custom Python package named `signal_ICT_StudentName_EnrollmentNo` that demonstrates fundamental concepts of Signals and Systems. The package must be modular, containing three separate modules:

1. unitary_signals.py

Implement the following functions:

- `unit_step(n)` – Generates a unit step signal.
- `unit_impulse(n)` – Generates a unit impulse signal.
- `ramp_signal(n)` – Generates a ramp signal.

Each function should return a NumPy array and plot the signal using matplotlib.

2. trigonometric_signals.py

Implement the following functions:

- `sine_wave(A, f, phi, t)` – Generates a sine wave with amplitude A, frequency f, phase phi, and time vector t.
- `cosine_wave(A, f, phi, t)` – Generates a cosine wave with similar parameters.
- `exponential_signal(A, a, t)` – Generates an exponential signal.

3. operations.py

Implement the following signal operations:

- `time_shift(signal, k)` – Shifts the signal by k units.
- `time_scale(signal, k)` – Scales the time axis of the signal by factor k.
- `signal_addition(signal1, signal2)` – Performs addition of two signals.
- `signal_multiplication(signal1, signal2)` – Performs point-wise multiplication of two signals.

Main Script (main.py)

- Import the above modules from the package.

- Demonstrate the following tasks:
 1. Generate and plot a unit step signal and a unit impulse signal of length 20.
 2. Generate a sine wave of amplitude 2, frequency 5 Hz, phase 0, over $t = 0$ to 1 sec.
 3. Perform time shifting on the sine wave by +5 units and plot both original and shifted signals.
 4. Perform addition of the unit step and ramp signal and plot the result.
 5. Multiply a sine and cosine wave of same frequency and plot the result.

Expected Deliverables:

1. Folder structure of the package:
2. signal_ICT_StudentName_EnrollmentNo /
3. __init__.py
4. unitary_signals.py
5. trigonometric_signals.py
6. operations.py
7. main.py
8. Well-documented Python code with function definitions and comments.
9. Proper use of NumPy (for signal arrays) and Matplotlib (for plotting).
10. At least 3 plots showing signals and operations as per requirements.

Student Submission Requirements:

- The Wheel file (.whl) and source distribution (.tar.gz) inside a dist/ folder.
- A README.md explaining package modules, installation, and usage.
- A screenshot/PDF report showing:
 1. Successful local installation from wheel.
 2. Successful upload to TestPyPI.
 3. Successful installation from TestPyPI.
- GitHub repo link.

Reference Link: <https://www.youtube.com/watch?v=9Ii34WheBOA>

Evaluation Criteria rubric (20 Marks):

- Package Design & Modularity (5 Marks) – Proper structure with three modules.
- Correct Implementation of Signals (5 Marks) – Unitary & trigonometric signals.
- Correct Implementation of Operations (5 Marks) – Scaling, shifting, addition, multiplication.
- Main Script Demonstration (3 Marks) – Calling and plotting from all modules.
- Code Quality & Documentation (2 Marks) – Comments, readability, efficiency.

```
adding 'signal_ICT_HeerMehta_92400133037/trigonometric_signals.py'
adding 'signal_ICT_HeerMehta_92400133037/unitary_signals.py'
adding 'signal_ict_heermehta-0.1.0.dist-info/licenses/LICENSE'
adding 'signal_ict_heermehta-0.1.0.dist-info/METADATA'
adding 'signal_ict_heermehta-0.1.0.dist-info/WHEEL'
adding 'signal_ict_heermehta-0.1.0.dist-info/top_level.txt'
adding 'signal_ict_heermehta-0.1.0.dist-info/RECORD'
removing build\bdist.win-amd64\wheel
Successfully built signal_ict_heermehta-0.1.0.tar.gz and signal_ict_heermehta-0.1.0-py3-none-any.whl
```

View at:

<https://test.pypi.org/project/signal-ICT-HeerMehta-92400133037/0.1.0/>

<https://test.pypi.org/project/signal-ICT-HeerMehta-92400133037/0.1.1/>

```
PS E:\College\Codes\Embedded C> pip install --index-url https://test.pypi.org/simple/ --no-deps signal-ICT-HeerMehta-92400133037
Looking in indexes: https://test.pypi.org/simple/
Collecting signal-ICT-HeerMehta-92400133037
  Downloading https://test-files.pythonhosted.org/packages/84/ad/35f330c16d198f1d5662dcb7a566fb2ce84d2ba9687f53e102566f3110d4/signal_ict_heermehta_92400133037-0.1.1-py3-none-any.whl.metadata (2.5 kB)
  Downloading https://test-files.pythonhosted.org/packages/84/ad/35f330c16d198f1d5662dcb7a566fb2ce84d2ba9687f53e102566f3110d4/signal_ict_heermehta_92400133037-0.1.1-py3-none-any.whl (5.2 kB)
Installing collected packages: signal-ICT-HeerMehta-92400133037
Successfully installed signal-ICT-HeerMehta-92400133037-0.1.1
PS E:\College\Codes\Embedded C>
```

```
✓ SIGNAL_ICT_HEERMEHTA_...
  ✓ dist
    ≡ signal_ict_heermeht...
    ≡ signal_ict_heermeht...
  ✓ signal_ICT_HeerMeht...
    > __pycache__
    • __init__.py
    • operations.py
    • trigonometric_signal...
    • unitary_signals.py
    • LICENSE
    • main.py
    • pyproject.toml
    • README.md
    • setup.py
```

GITHUB LINK-

https://github.com/Heer972005/signal_ICT_HeerMehta_92400133037