EXPERIMENT 1

Weather Modeling Using Quadratic Equation (4 Input Stages)

Name: T. Aruna Saroj

Regd. No.: 2023001153

Date: 12 July 2025

Aim:

To implement weather modeling using a quadratic equation and develop a system that predicts temperature using various input methods — hardcoded, keyboard input, file input, and multiple datasets from a CSV file.

Tools Used:

- Python
- NumPy
- Matplotlib
- Pandas
- IDLE (or any Python IDE)

Theory:

A quadratic equation is used to fit temperature vs. time data:

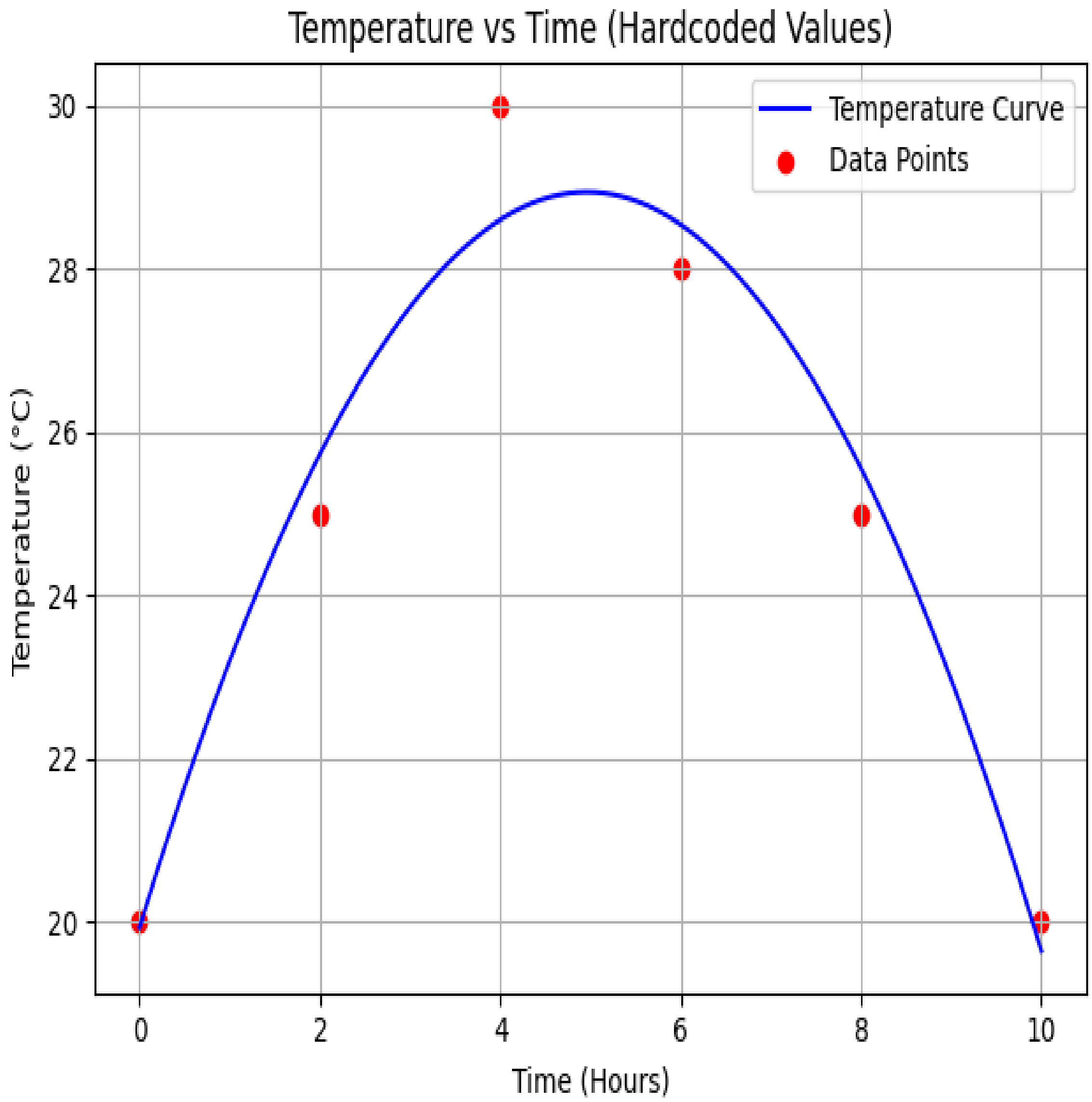
T(x)=Ax2+Bx+CT(x) = Ax^2 + Bx + C

Where T(x) is temperature, and x is time. The coefficients A, B, and C are determined using NumPy's polyfit () method.

Stages of Implementation:

Stage 1: Hardcoded Input

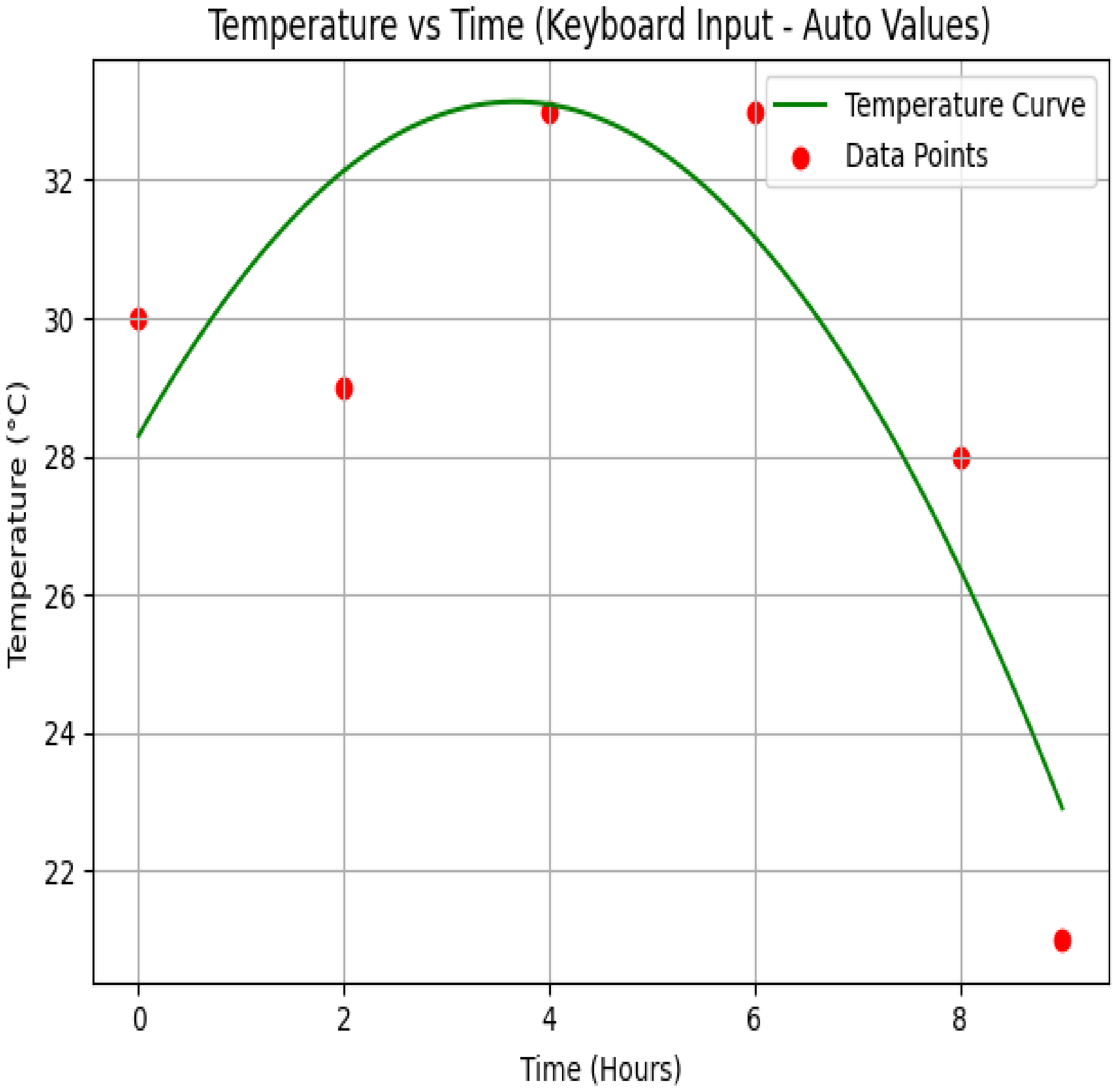
- Time and temperature values are fixed inside the code.
- Output: A graph is generated with data points and a fitted temperature curve.
- Graph saved as: exp1_stage1_graph.png



-

Stage 2: Keyboard Input

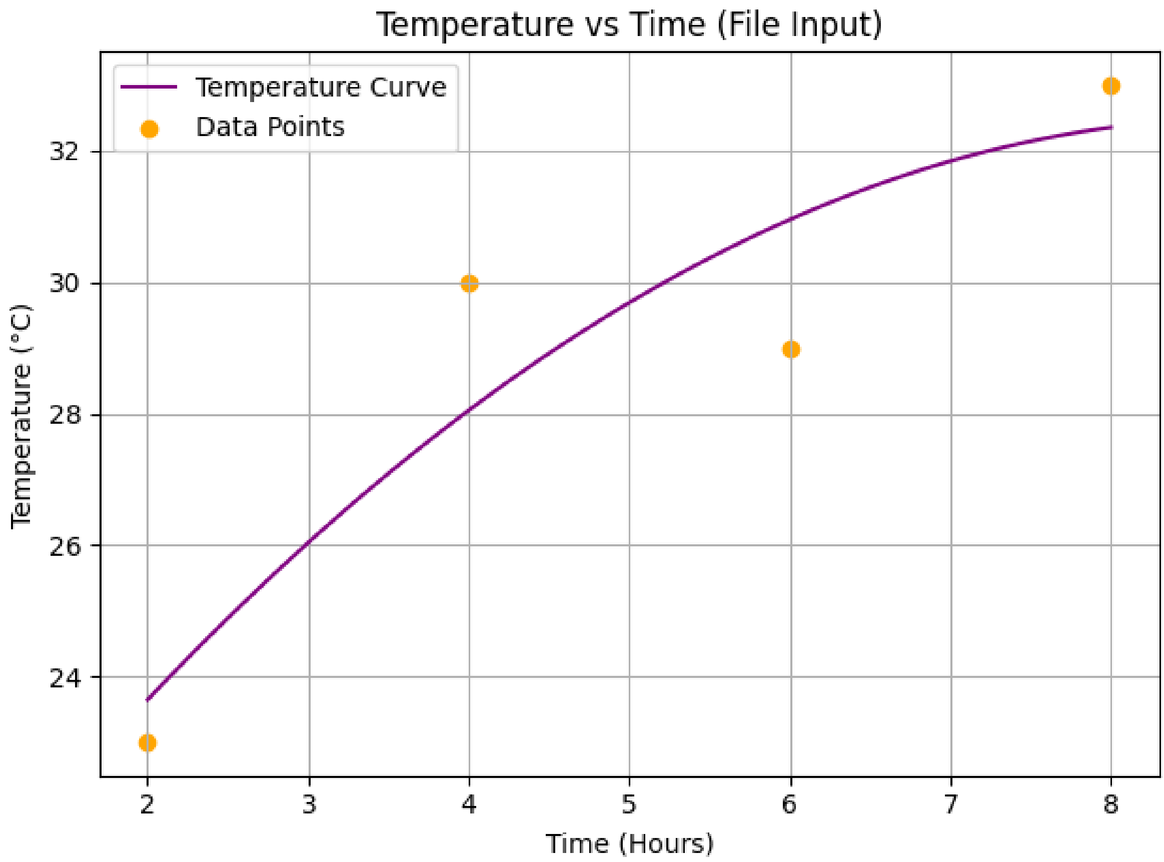
- The user enters time and temperature values via input ().
- The graph is generated based on this data.
- Graph saved as: exp1_stage2_graph.png



-

Stage 3: File Input

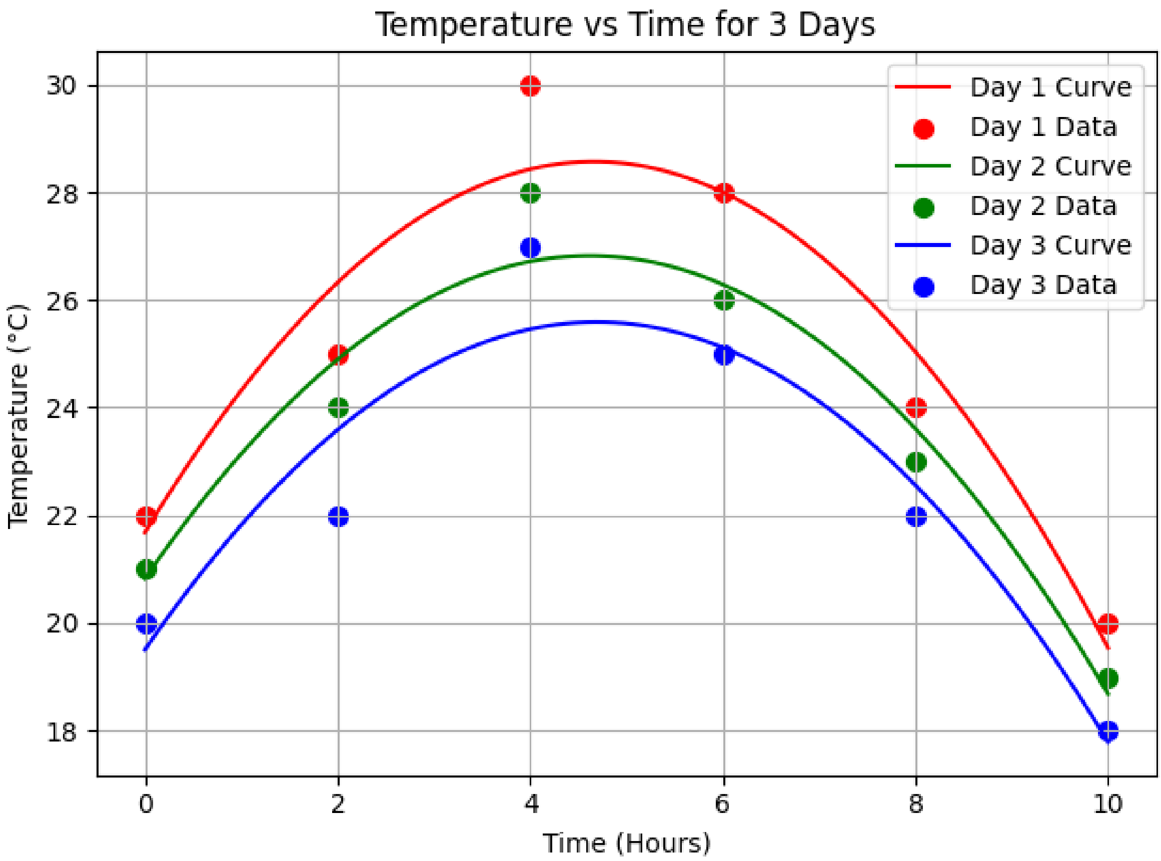
- Time and temperature values are read from a text file (data.txt).
- The model reads, processes, and plots the temperature curve.
- Graph saved as: exp1_stage3_graph.png



-

Stage 4: Multiple Input Sets from File

- Data for 3 different days is taken from a CSV file (data_multi.csv).
- Each day’s temperature is plotted with a separate curve and color.
- All curves and data points appear in a single graph.
- Graph saved as: exp1_stage4_graph.png



-

Results:

- All 4 stages executed successfully.
- Each method generated a smooth quadratic curve with real or simulated temperature values.
- Final graph (Stage 4) displayed all 3-day curves with distinct data points.

Conclusion:

This experiment shows how a quadratic equation can be used to fit and predict temperature data. It also demonstrates input flexibility — from manual entry to file-based input — and the ability to handle multiple datasets in one system. The graphs help visualize temperature trends effectively.

Output Files:

- stage1_hardcoded.py
- stage2_keyboard_input.py
- stage3_file_input.py
- stage4_multiple_sets.py
- data.txt
- data_multi.csv
- Graphs:
    - exp1_stage1_graph.png
    - exp1_stage2_graph.png
    - exp1_stage3_graph.png
    - exp1_stage4_graph.png