

## ICE 2: Temperature Sensor Program Testing (Individual)

---

### Objective:

To implement and test a Python program for a **Temperature Sensor** application using **Boundary Value Analysis** (BVA) and **Robustness Testing**. Students will explore different input scenarios, identify edge cases, and document results to ensure program robustness.

### Program Requirements:

- a) **Boundary Value Analysis (BVA):** Students will test input values within the valid range ( $-50^{\circ}\text{C}$  to  $150^{\circ}\text{C}$ ) and near boundaries to verify proper behavior.
- b) **Robustness Testing:** Students will explore invalid input values (outside the range, non-numeric inputs) to test how the program handles exceptions.
- c) **Expected Program Behavior:**
  - ✓ Compute **minimum**, **maximum**, and **average temperature** from a list of values.
  - ✓ Handle invalid or empty input gracefully (e.g., return error messages).
  - ✓ Validate inputs to ensure they are within the acceptable range.
- d) **Program Features:**
  - a. Input: A list of temperatures (integers or floats).
  - b. Output:
    - i. Minimum temperature
    - ii. Maximum temperature
    - iii. Average temperature (to 2 decimal places)
  - c. Error Handling:
    - i. Empty list: Show "No input provided."
    - ii. Invalid data: Show "Invalid input detected."
    - iii. Out-of-bound values: Show "Out-of-bound value detected."
- e) **Use the Provided Test Case Template:** Complete the table for each test case.

### Scenarios to Test:

- a) **Boundary Value Analysis (BVA)**
  - ✓ Test Case 1: Input at minimum boundary:  $[-50]$ .
  - ✓ Test Case 2: Input at maximum boundary:  $[150]$ .
  - ✓ Test Case 3: Inputs near the boundary:  $[-49, 149]$ .
- b) **Robustness Testing**
  - ✓ Test Case 4: Mixed valid and invalid inputs:  $[-60, 20, 160]$ .
  - ✓ Test Case 5: Alphabetic characters in input:  $[20, \text{"abc"}, 30]$ .
  - ✓ Test Case 6: Special characters in input:  $[10, \text{"@"}, -40]$ .
- c) **Special Scenarios**
  - ✓ Test Case 7: Very large input:  $[2^{**}31 - 1, -2^{**}31]$ .
  - ✓ Test Case 8: All inputs are the same:  $[50, 50, 50]$ .
  - ✓ Test Case 9: Empty list:  $[\ ]$ .

## ICE 2: Temperature Sensor Program Testing (Individual)

### Complete Test Case Table

Test Case No.	Sub-Case	Input Size	Input Values	Expected Output	Actual Output	Match (Yes/No)
1	A	1	[20]	Min: 20°C, Max: 20°C, Avg: 20°C		
	B	2	[15, 35]	Min: 15°C, Max: 35°C, Avg: 25°C		
2	A	0	[]	Error: "No input provided."		
3	A	3	[10, -10, 30]	Min: -10°C, Max: 30°C, Avg: 10°C		
4	A	4	[-50, 20, 150, 25]	Min: -50°C, Max: 150°C, Avg: 36.25°C		
	B	3	[10, "abc", 30]	Error: "Invalid input detected."		
5	A	2	[2**31 - 1, -2**31]			
6	A	3	[10, 10, 10]			

[Note: You may think of additional test cases beyond the given. Include in the table below if you find more.]

### General Submission Instructions:

- **Python File:** Save the file as ICE1\_<StudentID>.py
- **Report (Word/PDF):**
  - Completed **test case table** with input, expected output, actual output, and match
  - **Screenshots** of program execution