Homework 1   **100 Points Possible**

2/10/2025

Attempt 2 ⌄	◯ 2/4/2025

NEXT UP: Review Feedback

Attempt 2 Score:

N/A   🗨 Add Comment

Unlimited Attempts Allowed

2/11/2025

⌄ **Details**

Create a folder on your machine, for example, 'hw1,' and copy the following two files into that folder:

main.py (https://canvas.uh.edu/courses/19255/files/5863486?wrap=1) ↓ **(https://canvas.uh.edu/ courses/19255/files/5863486/download?download_frd=1) (https://canvas.uh.edu/courses/19255/ files/5863504?wrap=1)**

library.py (https://canvas.uh.edu/courses/19255/files/5863504?wrap=1) ↓ **(https://canvas.uh.edu/ courses/19255/files/5863504/download?download_frd=1)**

Add your code to main.py and upload this file to Canvas before the deadline.

Detailed Instructions

Objective

Python scripts find widespread application in processing sensor data. For this assignment, you will leverage the procedural Python concepts covered in class (Unit 1) to interact with a simulated (virtual) sensor, computing a range of fundamental yet valuable statistics. Additionally, your task involves crafting a text charts to visually represent the acquired data.

Requirements

You will need Spyder or a similar Python IDE to develop your program.

Procedure

1. Download the files `library.py` and `main.py` from Canvas.
2. Create a dedicated folder on your machine and move both files into that folder.
3. Open the folder, right-click on `main.py`, select "open with," and choose Spyder.
4. The file `main.py` contains a fully implemented function, `calculate_mean`, that shows how to retrieve data from a `Sensor` object. This function takes two parameters: a sensor object and the number of data samples `N` to calculate the mean value. Each data reading is always between 0 to 5 v.
5. Your task is to implement the following additional functions, all accepting the same parameters as `calculate_mean`. Make sure that you name the functions as described here:

- `calculate_std`: Returns the standard deviation of the data samples.

- `calculate_min(sr, N)`: Returns the smallest value of the data.

- `calculate_max(sr, N)`: Returns the largest value of the data.

- `calculate_range(sr, N)`: Returns the difference between the largest and smallest data.

- Notes:

  - `Sensor` returns different values each time you read it. So, the results accross function calls may not be consistent as they will operate on different data.
  - For all homework, you are restricted to using the material covered in the course this far. Since we have not covered lists, vectors, arrays, etc. you will not be allowed to use those.

6. Implement the `plot` function (defined with the same two parameters as the rest). This function generates a text-based chart of the data, ensuring the data is normalized so that the maximum line displayed is 40 characters long. That is, 0 v shows an empty line and 5 v shows a 40-char line.

7. Implement the `count` function (once again, defining the same two parameters). This function counts and visualizes the number of data samples falling into categories: below 0.25 V, between 0.25 V and 0.75 V, and above 0.75 V. Additionally, ensure the visual representation adheres to a maximum line length of 40 characters.

8. Upload your `main.py` file only.

- Do not compress it.
- Do not upload `library.py`.

Notes

- Add your name, email and PSID the top of your program:

```
# YOUR NAME <email> (PSID)
```

- To be eligible for points:
  - Develop your program exclusively using the concepts covered in class.
  - The program must run without major (runtime) errors.
  - Your submitted program **must be entirely your own original and independent work**.
- Include comments as needed to facilitate the grading of your program.
- While the use of generative AI tools like ChatGPT, Bard, and CoPilot is not prohibited for assignments (it is for exams) in this class, it should be employed as a study assistant and not as a means to complete your homework. You may be asked to provide explanations for any material submitted for grading at any point during the semester. Posted grades may be subject to revision or result in a failing grade if you are unable to adequately explain your submitted programs.