Test assignment

Introduction

You are given an ASCII file containing simulation results. Data is stored in a format of a GIDAS file, which has the following structure:

```
BEGIN
# All lines in BEGIN-END block are key-value pairs.
# Comments can appear on any line between BEGIN and END blocks.
SPEED = 6000
NodeID = '607-1abs'
START = 0
CHANNEL = ['Freq', 'Disp', 'Velo', &
    'Acce', 'Load', 'Forc']
UNIT = ['Hz', 'm', 'm/s', 'm/s^2', 'N', 'N']
END
        0         0         0         0         0         0
       10     2.3E-3    0.0015    16.5E-4        18       419
```

Lines between *BEGIN* and *END* contain parameters, key-value pairs. Immediately after a line containing *END*, there is a matrix containing the data. Number of columns in the matrix is equal to the number of elements in the *CHANNEL* and *UNIT* lists.

Assignment

This assignment is split into two connected parts. In the first one your task is to read a specific channel from GIDAS files. In the second part you plot the values read in the first part.

Reading GIDAS Files

Your task is to read a set of GIDAS files coming from a CFD simulation performed using AVL FIRE™. Simulation results are given for several speed cases, each containing a single GIDAS file. We are interested in reading data for channel 'flow:total_mass'.

Plotting GIDAS Files

Once the channels for all provided GIDAS files have been read, you are expected to plot the given channel, for each file in a separate figure, versus crank angle. After results for all files have been plotted, create a summary plot.

The summary should display a curve of average 'flow:total_mass' values, across all speeds, for each single crank angle processed during the simulation. In addition to the summary diagram, please also create a file containing the values shown in the summary diagram.

The values on each diagram must be displayed in grams, not kilograms. Please take this into account.

When developing a solution, please take care how you define which channel is plotted. If possible, do not hardcode it in just one place. Make it possible for a user to define a channel they need.

The script doesn't have to be located in the same location in which the results are stored. Make it possible for users to define where the results are stored and which speed cases have to be processed.

Expected Diagrams

In this example, all diagrams are presented in the same figure. This is not required. It's perfectly acceptable to put and save each diagram in its own figure.
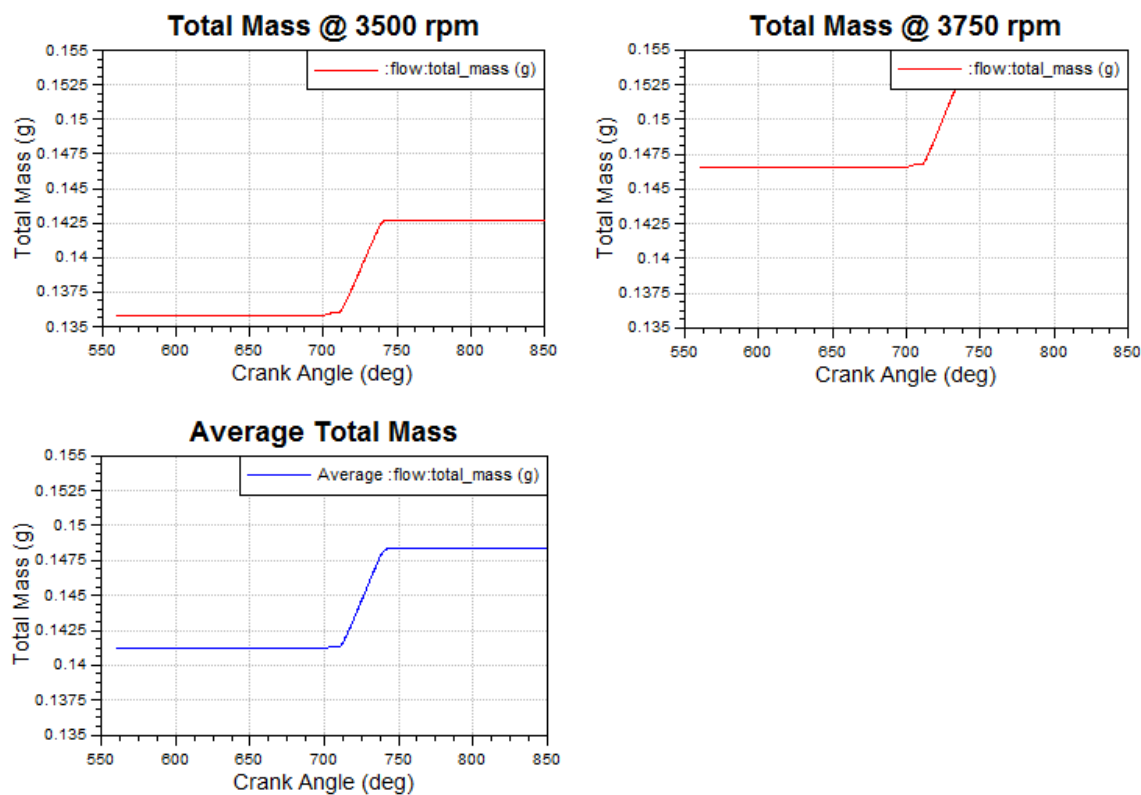
Remarks

Please make sure to use the common programming best practices when developing the task. Be careful to use a single Python style guide. Don't overcomplicate it, though. If you feel you can improve the assignment in any way, feel free to do so. You are also free to define any visual style and graphics you like.